

УТВЕРЖДЕН

**Серверная операционная система с
интегрированными серверными службами
МСВСфера 9 Сервер
Руководство администратора**

версия 1.0

Индв. № подл.	Подпись и дата	Взам. инв №	Индв. № дубл.	Подпись и дата

Оглавление

Аннотация	5
1 Общие сведения	6
1.1 Назначение и область применения	6
1.2 Обеспечение безопасности и требования к администратору	6
2 Установка и начальная настройка системы	9
2.1 Системные требования	9
2.2 Создание загрузочного USB-носителя и запись iso-образа дистрибутива	9
2.3 Установка системы с USB-носителя	14
3 Управление пакетами	23
3.1 Введение и основные понятия	23
3.2 Пакетный менеджер DNF	24
3.3 Безопасность	32
4 Идентификация и аутентификация	34
4.1 Введение	34
4.2 Добавление нового пользователя	34
4.3 Изменение уже имеющихся пользовательских записей	36
4.4 Удаление пользователей	38
4.5 Добавление группы пользователей	39
4.6 Изменение существующей группы пользователей	40
4.7 Удаление существующей группы пользователей	40
4.8 Создание и изменение пароля пользователя	41
4.9 Изменение срока действия учётной записи и пароля пользователя	43
4.10 Получение сведений о пользователе	44
4.11 Конфигурационный файл /etc/login.defs	45
4.12 Конфигурационный файл /etc/pam.d/system-auth	47
4.13 Конфигурационный файл /etc/issue	51
4.14 Конфигурационный файл /etc/shadow	51
5 Управление доступом	53
5.1 Введение	53
5.2 Установка и изменение прав доступа к файлам и директориям	53
5.3 Назначение и изменение владельца файла и директории	54
5.4 Изменение группы-владельца файла или директории	55
5.5 Просмотр и изменение списков правил контроля доступа для файлов и директорий	55
5.6 Просмотр списков контроля доступа	56
5.7 Редактирование пользовательских квот для файловой системы	59
5.8 Конфигурационный файл /etc/profile	59
5.9 Конфигурационный файл /etc/security/limits.conf	62
5.10 Конфигурационный файл /etc/fstab	65

6	Регистрация событий безопасности	67
6.1	Введение	67
6.2	Создание и удаление правил регистрации событий безопасности	67
6.3	Добавление правила регистрации событий безопасности	70
6.4	Поиск данных регистрации событий безопасности	71
6.5	Генерация отчетов по данным регистрации событий безопасности	72
6.6	Конфигурационный файл /etc/audit/auditd.conf	74
7	Ограничение программной среды	76
7.1	Введение	76
7.2	Включение программ в автозагрузку	76
7.3	Управление системными службами	77
7.4	Настройка запуска программ по расписанию	79
7.5	Управление программными пакетами	80
7.6	Установка последней версии пакета/группы пакетов	81
8	Стирание данных	83
8.1	Введение	83
8.2	Заполнение случайными числами места, занятого файлами	83
8.3	Стирание данных в свободном пространстве раздела, в котором находится директория	84
8.4	Стирание данных в разделах подкачки	85
8.5	Стирание данных в оперативной памяти	85
9	Контроль целостности	87
9.1	Введение	87
9.2	Вычисление и сверка контрольной суммы файла	87
9.3	Проверка целостности данных	88
10	Обеспечение надёжного функционирования	91
10.1	Введение	91
10.2	Архивация файлов и директорий	91
10.3	Создание архивов и извлечение файлов из них	92
10.4	Резервное копирование данных	93
10.5	Создание дисковых RAID-массивов	94
11	Фильтрация сетевого потока	96
11.1	Введение	96
11.2	Настройка файрвола (брандмауэра)	96
11.3	Конфигурационный файл /etc/firewalld/firewalld.conf	98
12	Мониторинг функционирования	100
12.1	Введение	100
12.2	Анализ системных журналов	100
12.3	Получение информации о выполняемых процессах	101
12.4	Получение информации о состоянии текущих процессов	102

12.5 Мониторинг и анализ сетевого трафика	102
12.6 Получение информации о сеансах пользователей	103
12.7 Получение информации о последних выполненных командах	104

Аннотация

Настоящее руководство предназначено для администраторов серверной операционной системы с интегрированными серверными службами МСВСфера 9 Сервер. Руководство ориентировано на специалистов, знакомых с операционными системами типа Linux и имеющих минимальный практический опыт работы с ними. Руководство снабжено примерами, сделанными в операционной системе МСВСфера 9 Сервер, установленной в базовой конфигурации.

1 Общие сведения

1.1 Назначение и область применения

МСВСфера 9 Сервер — серверная операционная система на основе ядра Linux с набором интегрированных служб и приложений, включающим веб-сервер, почтовый сервер, сервер служб сетевой инфраструктуры, серверы файлов и печати, средства резервного копирования и восстановления данных, множество других служб и приложений, а также средства администрирования и защиты информации. Развёрнутую ОС МСВСфера 9 Сервер применяют в качестве программной платформы для использования в корпоративных сетях и серверных окружениях.

МСВСфера 9 Сервер включена в Реестр отечественного ПО, запись №16243 от 30.12.2022

1.2 Обеспечение безопасности и требования к администратору

Внедрению и использованию операционной системы должны предшествовать подготовительные процедуры, направленные на обеспечение безопасности при приемке установочного дистрибутива операционной системы от поставщика, на обеспечение безопасной установки, настройки и запуска операционной системы и на создание безопасной среды её функционирования. Реализация подготовительных процедур должна обеспечиваться необходимыми ресурсами и сопровождаться назначением ответственных за их выполнение должностных лиц.

Процедуры безопасной приемки должны предусматривать меры подтверждения подлинности установочного дистрибутива операционной системы, исключающие возможности преднамеренного или непреднамеренного внесения изменений в поставляемую версию, т.е. замены её фальсифицированной или неработоспособной версией.

К таким мерам в общем случае относятся:

- проверка подлинности источника поставки путем визуального контроля наличия и целостности специальных защитных стикеров (наклеек, знаков) на упаковке комплекта поставки, а также целостности самой упаковки;
- проверка комплектности поставки в соответствии с заявкой, договорными материалами и спецификацией, сверка маркировки и номера версии;
- проверка целостности установочного дистрибутива с помощью программного средства контроля целостности путем сравнения с эталонным значением контрольной суммы или с помощью средств электронной подписи.

Процедуры безопасной установки, настройки, запуска операционной системы и создания безопасной среды её функционирования в общем случае должны предусматривать меры, обеспечивающие:

- совместимость операционной системы со средствами вычислительной техники, на которых планируется её установка и использование;
- установку, конфигурирование, настройку, запуск и управление операционной системой в соответствии с эксплуатационной документацией и принятой политикой безопасности;
- защиту от действий, направленных на нарушение физической целостности средств вычислительной техники, на которых она функционирует;
- доверенную загрузку операционной системы, контроль доступа к процессу загрузки, блокирование попыток несанкционированной загрузки, контроль целостности компонентов загружаемой операционной среды;
- наличие ресурсов для выполнения функциональных возможностей безопасности операционной системы, хранения создаваемых резервных копий, а также защищенное хранение данных операционной системы и защищаемой информации;
- ограничение на установку программного обеспечения и его компонентов, не задействованных в технологическом процессе обработки информации;
- доверенный маршрут между операционной системой и пользователями;
- доверенный канал передачи данных между операционной системой и средствами вычислительной техники, на которых происходит обработка информации, а также с которых происходит их администрирование;
- невозможность отключения или обхода компонентов операционной системы и средств защиты информации;
- препятствие несанкционированному копированию информации, содержащейся в операционной системе, на съемные носители информации, в том числе контроль вноса (выноса) в (из) контролируемую зону съемных носителей информации;
- проверку целостности получаемых от поставщика внешних модулей уровня ядра перед их установкой в операционную систему;
- выделение вычислительных ресурсов для процессов в соответствии с их приоритетами;
- профессиональную компетентность и надежность персонала, ответственного за администрирование системы, его способность выполнять свои обязанности в точном соответствии с принятой политикой безопасности и эксплуатационной документацией;
- возможность генерации аутентификационной информации, соответствующей заданной метрике качества;
- недоступность аутентификационной информации для лиц, не уполномоченных на её использование;

- разделение полномочий пользователей и администраторов с назначением им минимально необходимых прав и привилегий;
- исключение в процессе использования системы доступа пользователей к приложениям, выполняющимся с более высокими правами доступа, чем права, предоставленные им согласно матрице доступа;
- завершение администраторами приложений, запущенных ими с административными правами, после окончания работы с ними;
- запрет пользователям на передачу посторонним лицам своей личной идентификационной и аутентификационной информации, а также на регистрацию кого-либо в системе под своим именем и паролем.

Для управления МСВСфера 9 Сервер используются командные интерпретаторы (shell). Поэтому администратор должен иметь:

- базовые навыки администрирования ОС семейства Linux;
- навыки конфигурирования и настройки программных продуктов и ОС;
- опыт работы со стандартными элементами графического интерфейса приложений;
- навыки поддержания в работоспособном состоянии технических средств ПК.

2 Установка и начальная настройка системы

2.1 Системные требования

Для использования операционной системы требуется компьютер со следующими минимальными характеристиками:

- Процессор Intel или AMD версии не ниже x86-64-v2 (Intel Nehalem и более поздние, AMD Bulldozer и более поздние).
- 2048 Мбайт оперативной памяти.
- 20 Гбайт свободного пространства памяти на жестком диске в зависимости от используемой конфигурации.

Установка МСВСфера 9 Сервер может осуществляться различными способами: с оптического диска, с жесткого диска, по сети. В данном документе описывается стандартная установка МСВСфера 9 Сервер с загрузочного USB-носителя. См. [2.3 Установка системы с USB-носителя](#).

2.2 Создание загрузочного USB-носителя и запись iso-образа дистрибутива

В настоящее время наиболее удобным способом установки операционной системы МСВСфера 9 Сервер является использование USB-носителя с записанным на него дистрибутивом. Ниже мы рассмотрим, как создать загрузочный USB-носитель и записать на него iso-образ дистрибутива.

Программное обеспечение, рекомендуемое для создания загрузочного USB-носителя и записи iso-образа дистрибутива МСВСфера 9 Сервер:

- [Fedora Media Writer](#) — для операционных систем семейства Windows, Linux и macOS;
- [balenaEtcher](#) — для операционных систем семейства Windows, Linux и macOS;
- [Win32 Disk Imager](#) — для операционных систем семейства Windows;
- Утилита командной строки `dd` — для операционных систем семейства Linux.

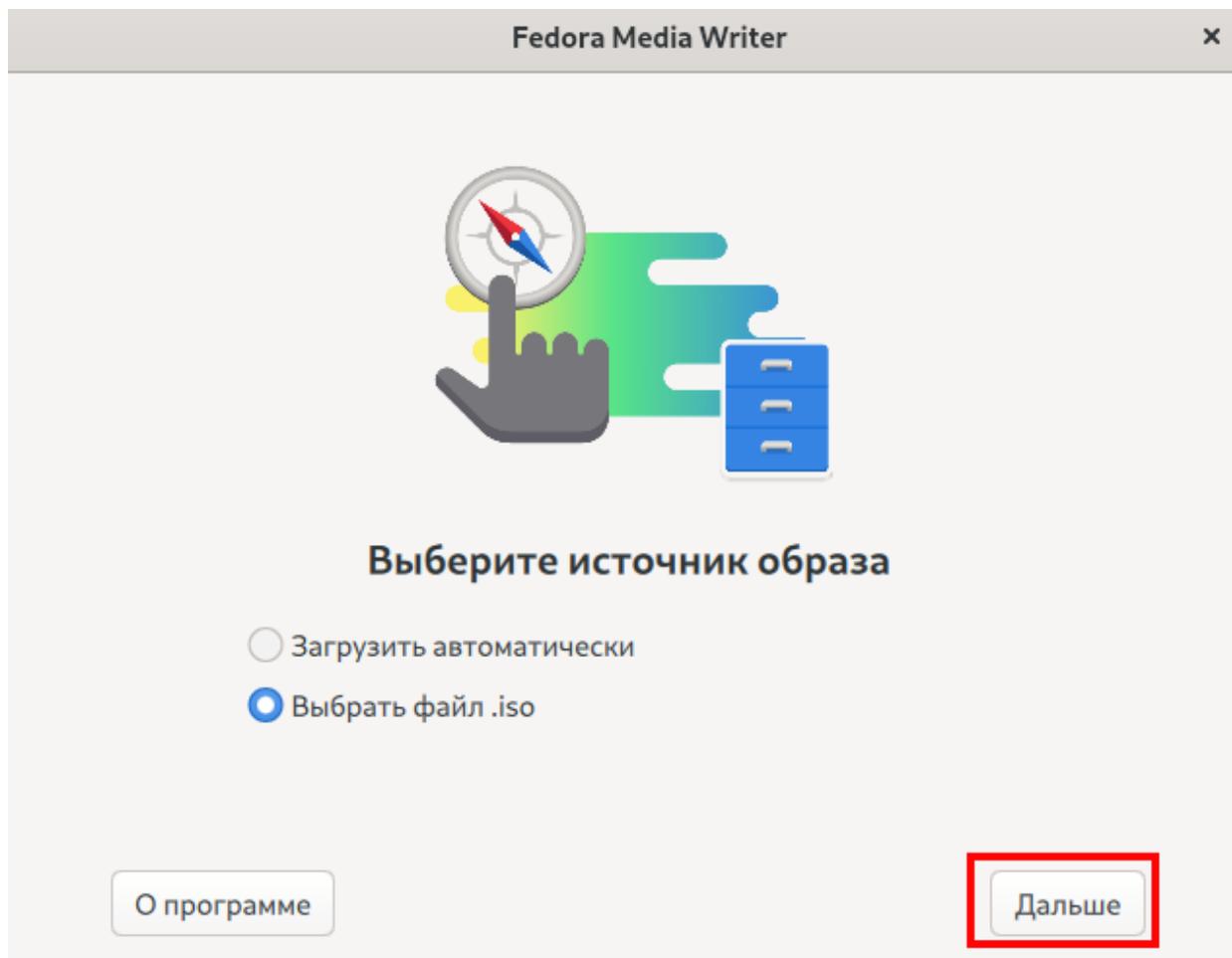
Интерфейс указанного программного обеспечения интуитивно понятный, дополнительные инструкции вы можете найти в документации соответствующего ПО.

В качестве примера рассмотрим процесс создания загрузочного USB-носителя и записи iso-образа дистрибутива МСВСфера 9 Сервер в программе Fedora Media Writer

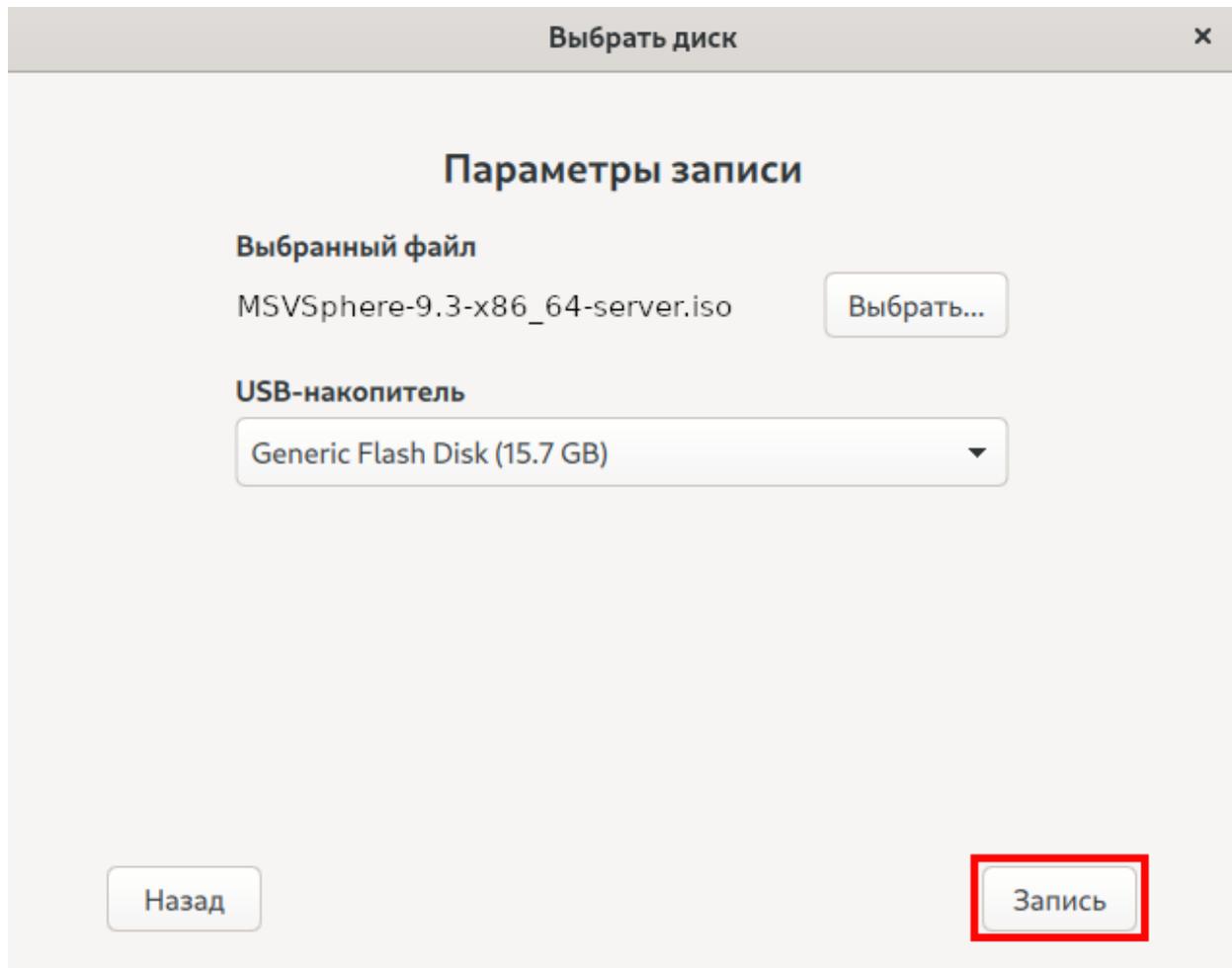
в операционной системе семейства Windows и с использованием утилиты командной строки dd в операционной системе семейства Linux.

2.2.1 Пример создания загрузочного USB-носителя и записи iso-образа дистрибутива МСВСфера 9 Сервер в программе Fedora Media Writer (Windows)

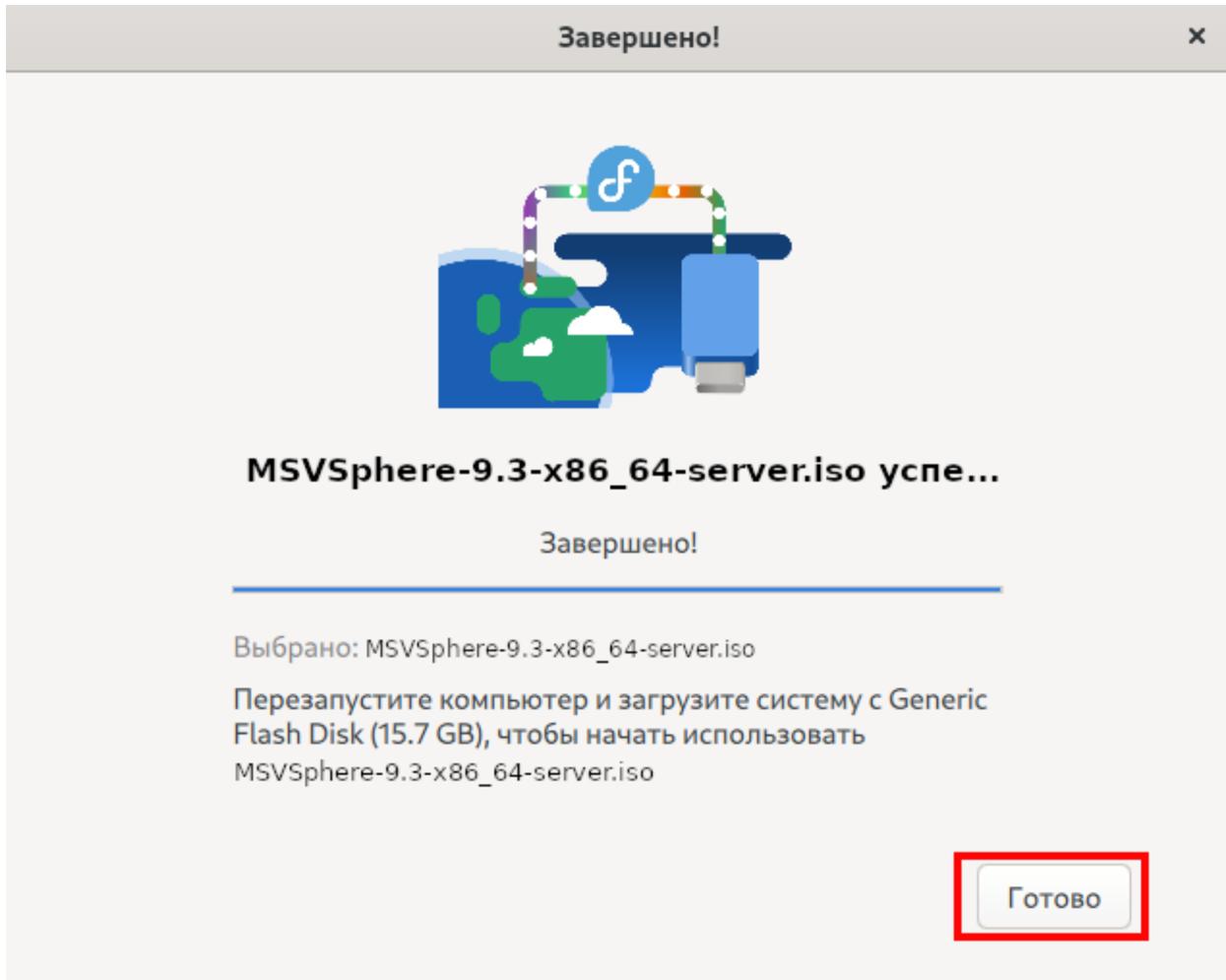
1. Скачайте последнюю версию Fedora Media Writer для Windows на ваше устройство.
2. Запустите установочный файл и выполните установку Fedora Media Writer на ваше устройство.
3. Вставьте USB-носитель, на который вы планируете записывать iso-образ дистрибутива. Убедитесь, что на нём достаточно места.
4. Скачайте актуальный iso-образ МСВСфера 9 Сервер: https://repo1.msvsphere-os.ru/msvsphere/9/isos/x86_64/.
5. Запустите Fedora Media Writer.
6. Выберите источник образа — «Выбрать файл iso» и нажмите «Далее».



7. В окне «Выбрать диск» → «Параметры записи» → «Выбранный файл» нажмите на кнопку «Выбрать» для выбора iso-образа МСВСфера 9 Сервер, загруженного ранее.
8. USB-накопители определяются автоматически. Если у вас подключено несколько USB-носителей, выберите необходимый из списка.
9. После выбора iso-образа МСВСфера 9 Сервер нажмите «Запись».



10. При необходимости укажите пароль администратора для подтверждения записи.
11. Начнётся запись iso-образа МСВСфера 9 Сервер на USB-носитель. Это может занять некоторое время.
12. После завершения записи нажмите «Готово».
13. Вы успешно создали загрузочный USB-носитель МСВСфера 9 Сервер! Теперь можно приступить к установке системы (см. «*2.3 Установка системы с USB-носителя*»).



2.2.2 Пример создания загрузочного USB-носителя и записи iso-образа дистрибутива МСВСфера 9 Сервер с помощью утилиты командной строки dd (Linux)

1. Вставьте USB-носитель, на который вы планируете записывать iso-образ дистрибутива. Убедитесь, что на нём достаточно места.
2. Скачайте актуальный iso-образ МСВСфера 9 Сервер: https://repo1.msvsphere-os.ru/msvsphere/9/isos/x86_64/.
3. Откройте «Терминал».
4. Введите команду для записи iso-образа:

```
dd oflag=dsync if=MSVSphe-9.3-x86_64-server.iso of=/dev/sdc bs=1M
↪status=progress;sync
```

При необходимости измените 9.3 на ту версию, которую вы устанавливаете.

Или

```
pv MSVSphere-9.3-x86_64-server.iso | dd oflag=dsync of=/dev/sdc bs=1M;  
↪sync
```

где `/dev/sdc` — это USB-носитель.

При необходимости измените **9.3** на ту версию, которую вы устанавливаете.

2.3 Установка системы с USB-носителя

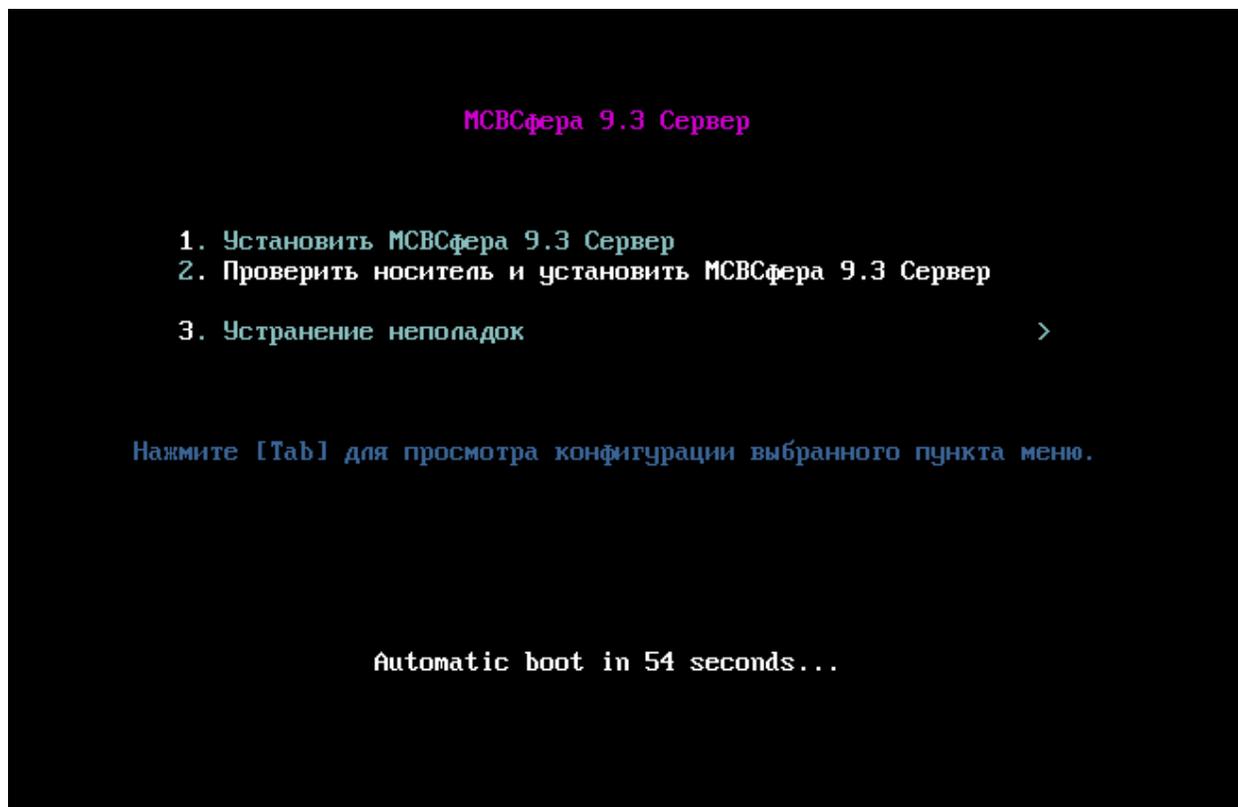
Для установки МСВСфера 9 Сервер с USB-носителя необходимо перед началом установки выбрать приоритетную загрузку с USB-носителя в BIOS устройства, либо выбрать загрузку с USB-носителя однократно в процессе инициализации компьютера.

Для установки и загрузки МСВСфера 9 Сервер может потребоваться отключить параметр Secure Boot в BIOS устройства, на которое производится установка.

Для начала установки подключите USB-носитель с установочным дистрибутивом к компьютеру.

Рассмотрим пример установки МСВСфера 9 Сервер.

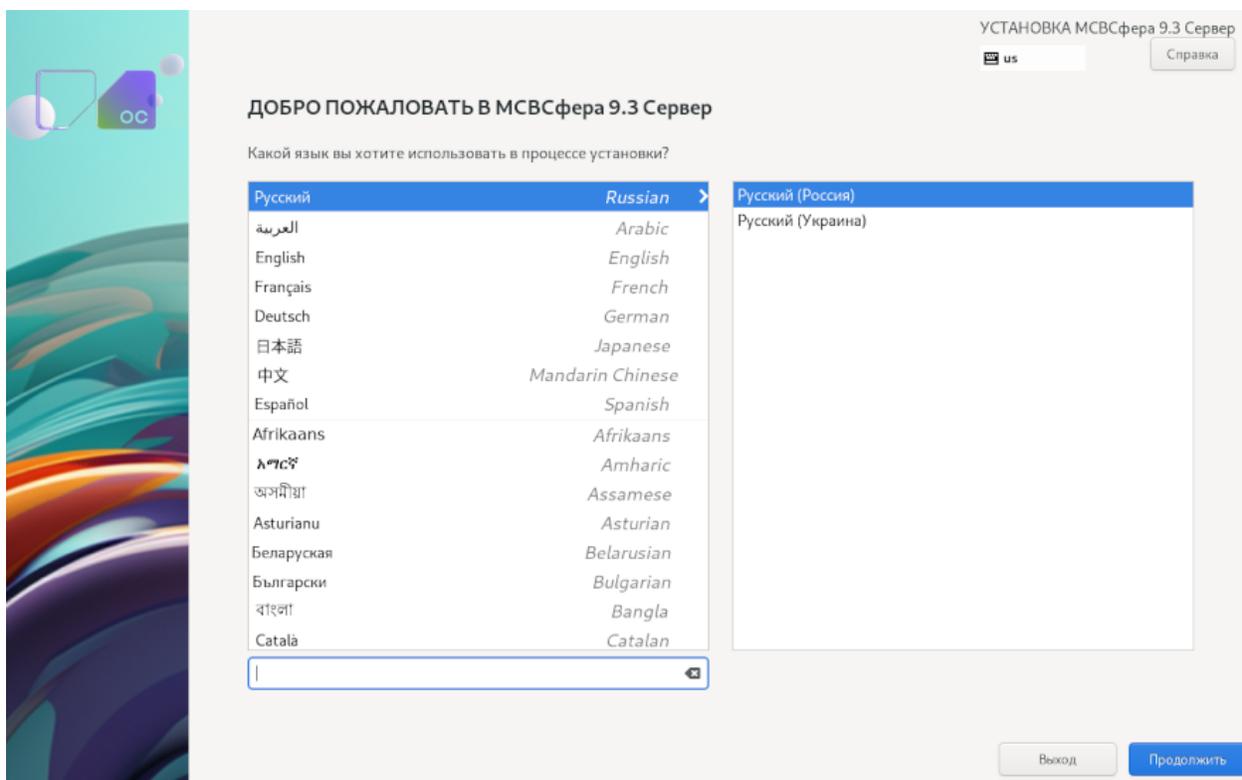
Сначала установка будет проходить в текстовом режиме.



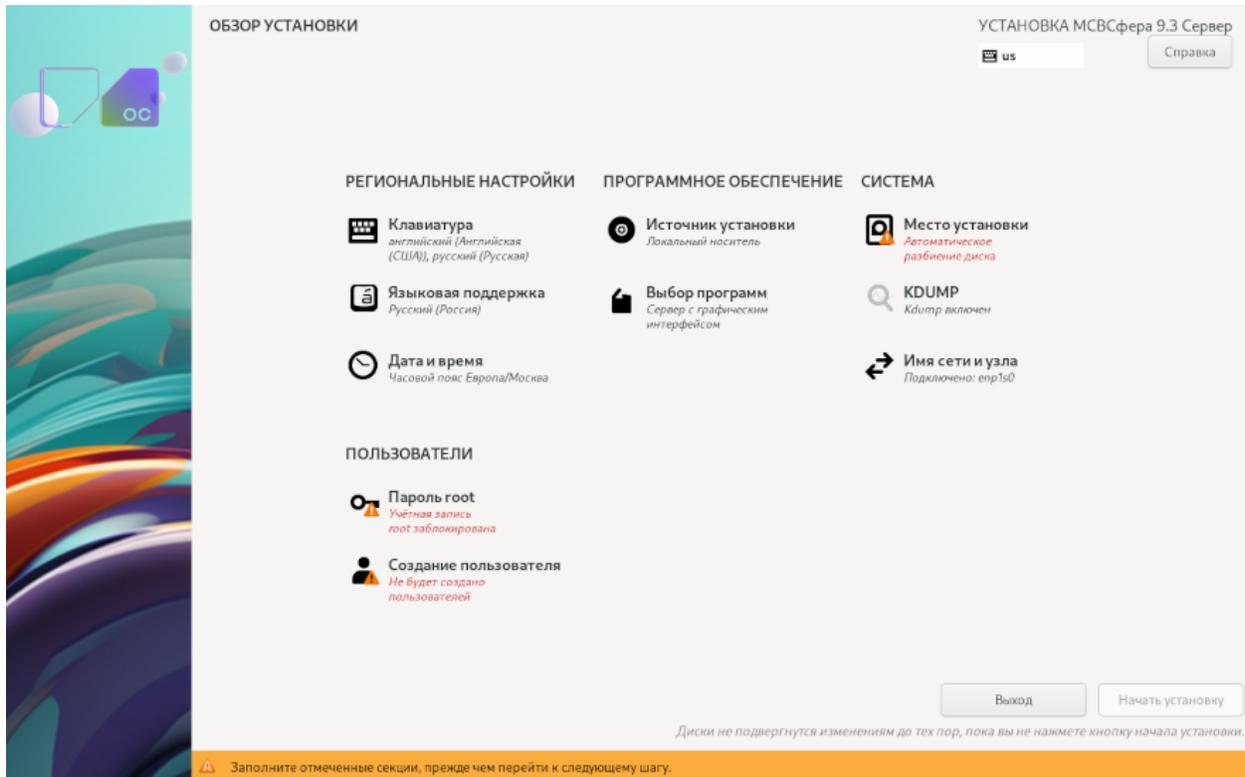
Доступны следующие варианты:

- Установить МСВСфера 9 Сервер — начнётся установка МСВСфера 9 Сервер на ваше устройство.
- Проверить носитель и установить МСВСфера 9 Сервер — программа установки проверит контрольные суммы образа диска, подтверждая что скачивание образа и запись на загрузочный носитель прошли без ошибок.
- Устранение неполадок — вы сможете перейти в режим восстановления, который представляет собой минимальную среду МСВСфера 9 Сервер, загружаемую с загрузочного носителя. В этом режиме используются утилиты командной строки, с помощью которых вы можете монтировать или не монтировать файловые системы, заносить в чёрный список и добавлять драйверы, устанавливая и обновлять системные пакеты, а также управлять разделами.

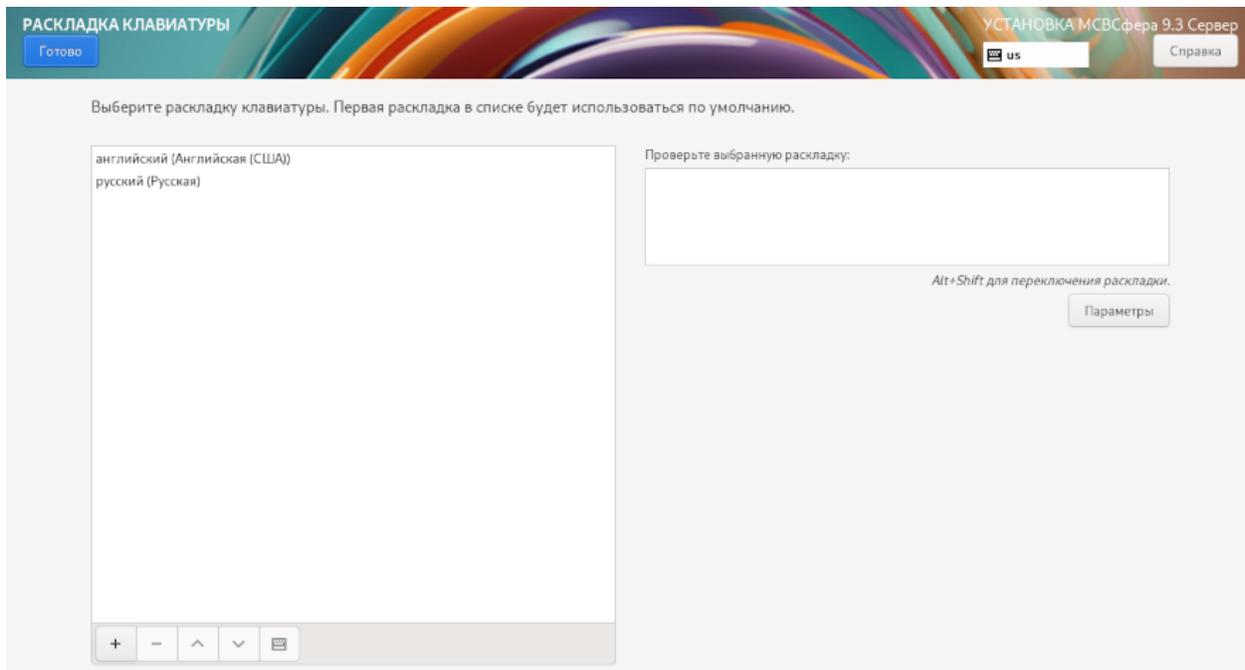
При нажатии на «Установить МСВСфера 9 Сервер» установка продолжится в графическом режиме и на экране монитора компьютера появится окно с предложением выбрать язык установки.



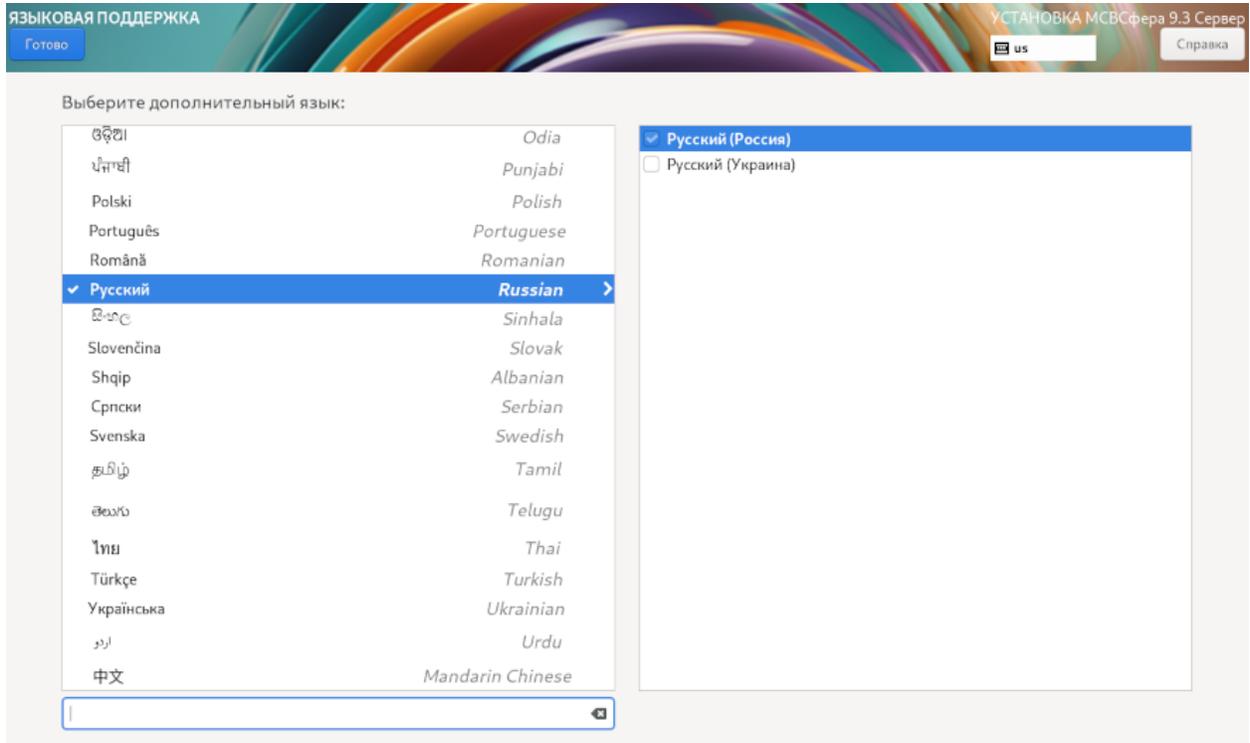
Затем появится окно «Обзор установки», с помощью которого, последовательно нажимая кнопку «Готово», можно будет произвести все необходимые настройки.



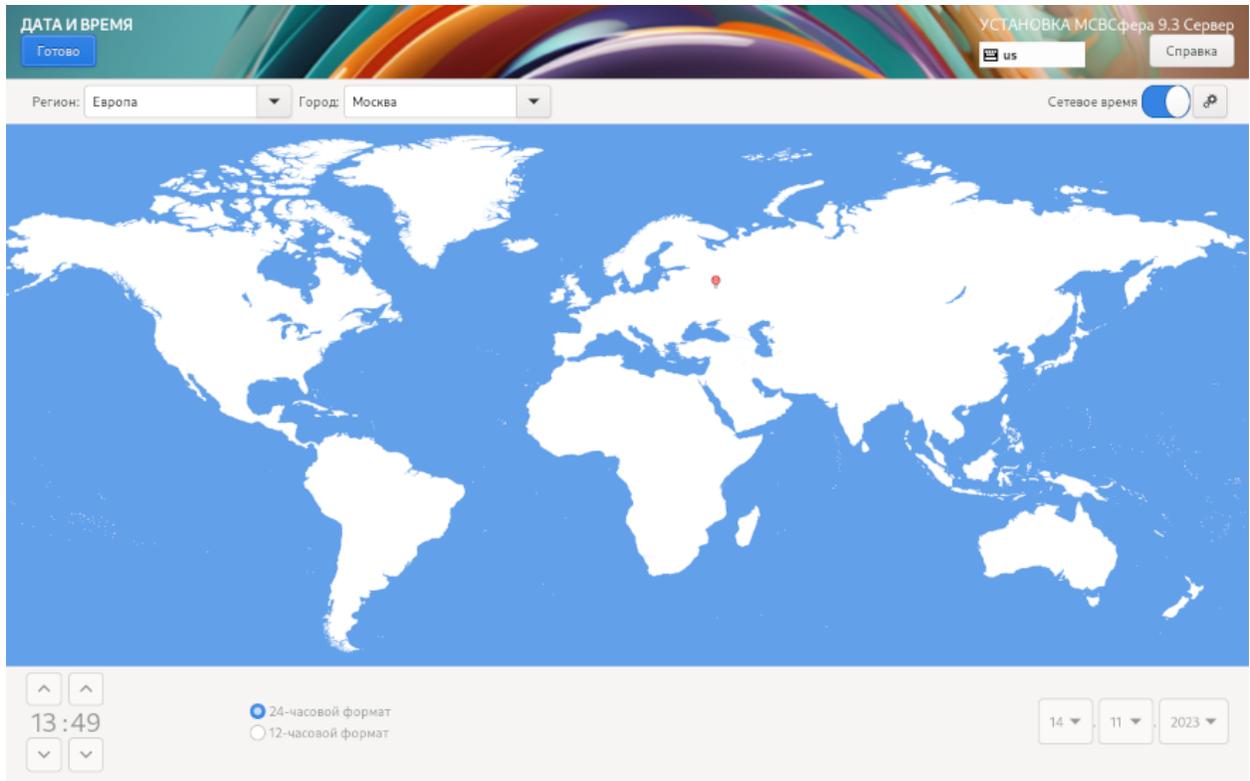
Раскладка клавиатуры.



Языковая поддержка.



Дата и время.



Источник установки.

Выбор программ.

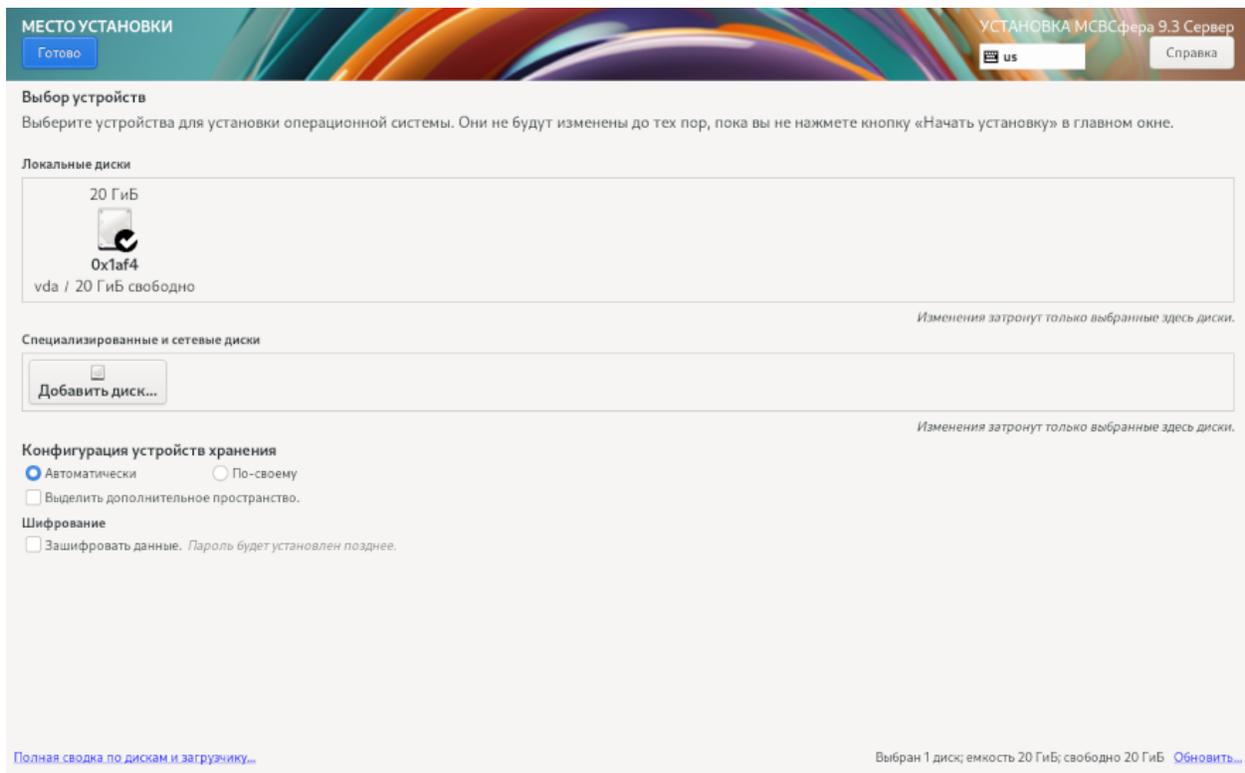
Доступны следующие варианты базового окружения:

- **Сервер с графическим интерфейсом** — включает все приложения, содержащиеся в окружении **Сервер**, а также графический интерфейс.

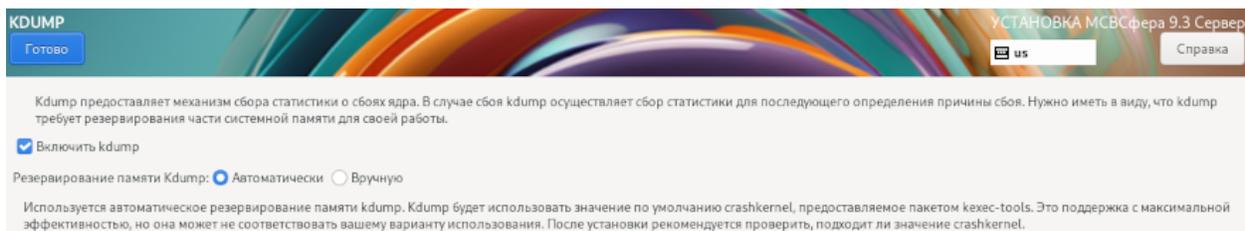
- **Сервер** — включает минимальный набор серверных приложений и текстовый интерфейс.
- **Сервер виртуализации** — включает все приложения, содержащиеся в окружении **Сервер**, а также пакеты для обеспечения виртуализации QEMU/KVM (без графического интерфейса).
- **Минимальная установка** — включает минимальный набор пакетов, необходимых для работы операционной системы, без приложений и без графического интерфейса.

Вы также можете добавить дополнительное программное обеспечение для выбранного базового окружения, отметив его в окне справа. Пакеты будут загружены и установлены автоматически.

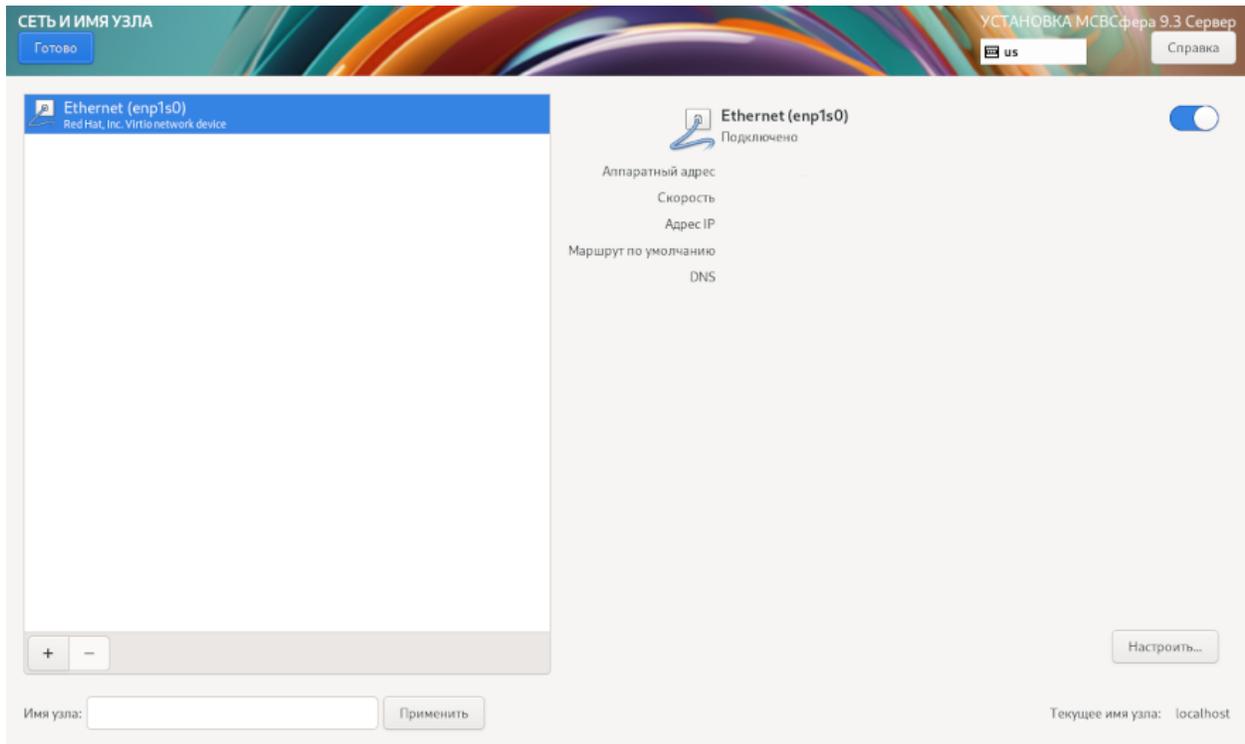
Место установки.



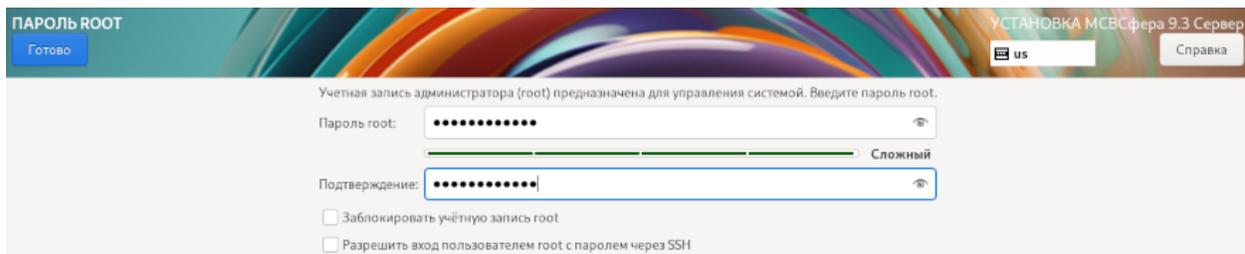
Диагностика сбоев ядра.



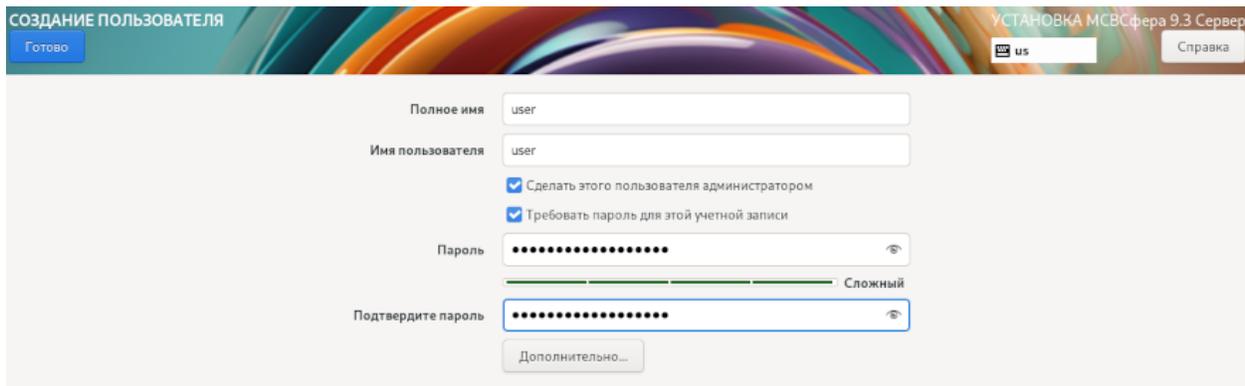
Имя сети и узла.



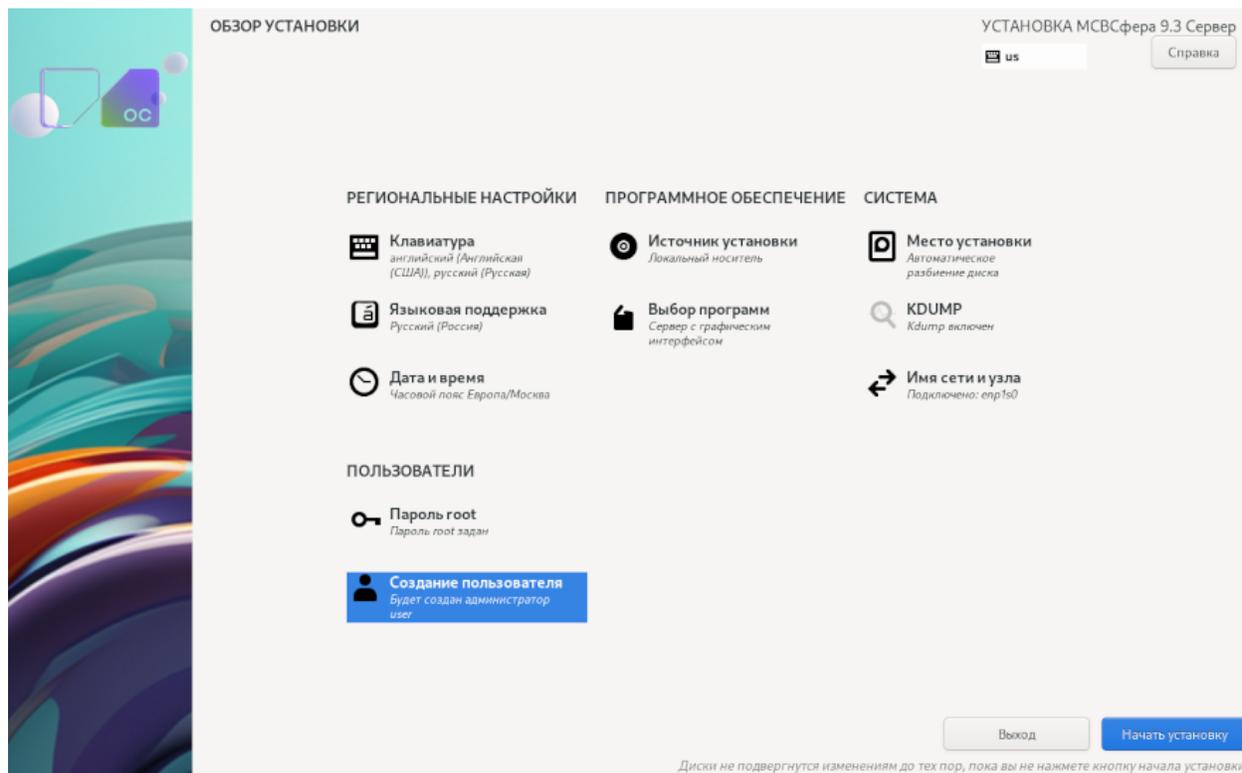
Задать пароль суперпользователя root.



И создать нового пользователя.



После того как все необходимые настройки произведены, нажмите нижнюю правую кнопку «Начать установку», и процесс установки начнётся.



Продолжительность установки может составить примерно 20-30 минут, в зависимости от быстродействия оборудования и выбранной конфигурации программного обеспечения.

По завершении установки на экране монитора появится соответствующее уведомление с предложением произвести перезагрузку.



После извлечения USB-носителя с установочным дистрибутивом и перезагрузки системы появится приглашение войти в систему, пройдя идентификацию и аутентификацию.

3 Управление пакетами

3.1 Введение и основные понятия

МСВСфера 9 Сервер представляет собой комплексную систему, которая обеспечивает стабильную и безопасную работу для пользователей.

Так как МСВСфера 9 Сервер собрана на базе ядра Linux, то в ней несколько приложений могут использовать одни и те же библиотеки или, например, одно приложение может использовать другое. С одной стороны это даёт возможность освободить место, занимаемое приложением, и снизить потребление ресурсов, а с другой стороны возникает необходимость обеспечения целостности системы.

Информация о всех необходимых приложению бинарных и конфигурационных файлах, о том, как их следует разместить в файловой системе, а также данные о зависимостях хранятся в архиве специального формата, называемом **пакетом**.

В МСВСфера 9 Сервер форматом пакета является RPM (рекурсивный акроним RPM Package Manager, ранее Red Hat Package Manager), а сами файлы, содержащие пакеты, имеют расширение `.rpm`.

Как было упомянуто выше, приложения могут совместно использовать одни и те же библиотеки или даже целые программы, и здесь возникает понятие **зависимости**: в приложении может не хватать чего-то для работы, и ему для этого нужно другое приложение или библиотека. То есть один пакет начинает зависеть от другого. И удалив, например, одну библиотеку можно нарушить работу сразу нескольких приложений.

Для работы с пакетами и обеспечения целостности системы используются программы, называемые **пакетными менеджерами**. Они управляют пакетами: устанавливают, удаляют, обновляют, ведут учёт, выводят информацию, отслеживают версии и зависимости и пр.

В МСВСфера 9 Сервер пакетным менеджером является **DNF**.

Так как пакеты зависят друг от друга, то зачастую недостаточно установить только один пакет — нужно устанавливать сразу несколько, поэтому разработчики создают и поддерживают специальные централизованные серверы, называемые **репозиториями**, где хранятся различные пакеты. Пакетный менеджер видит зависимости каждого пакета, сам находит подходящие пакеты в репозитории и предлагает их установить.

Дистрибутив МСВСфера 9 Сервер имеет набор собственных репозиториях для всех поддерживаемых выпусков и архитектур, в которых содержится огромное количество приложений и программ.

Обычно некоторые пакеты, которые часто используют вместе, объединены в **группы**. Посмотреть список доступных групп поможет пакетный менеджер DNF.

Кроме групп также есть **модули**, которые тоже содержат сразу несколько пакетов, но при этом пакеты в модуле связаны версиями.

3.2 Пакетный менеджер DNF

Рассмотрим основные операции с пакетами, которые может выполнить пакетный менеджер DNF.

3.2.1 Найти нужный пакет

Для поиска пакета (даже не зная его точного имени) выполните следующую команду:

```
dnf search имя_пакета
```

В имени пакета вы можете использовать шаблоны, а также указывать только те буквы из названия, которые помните.

Пример: найдём пакет по первым буквам (запустим от имени администратора — `sudo`):

```
sudo dnf search *fox
```

Результат работы команды:

```
[user@msvsphere test]$ sudo dnf search *fox
==== Имя совпадение: *fox =====
firefox.x86_64 : Mozilla Firefox Web browser
```

3.2.2 Установить нужный пакет

Для установки пакета выполните следующую команду:

```
dnf install имя_пакета
```

DNF проверит все зависимости и при обнаружении нужных, но ещё не установленных пакетов, установит их, пользуясь всеми доступными репозиториями.

Пример: установим пакет `firefox.x86_64` (запустим от имени администратора — `sudo`):

```
sudo dnf install firefox.x86_64
```

Результат работы команды:

```
[user@msvsphere test]$ sudo dnf install firefox.x86_64
Зависимости разрешены.
```

```
=====
Пакет                Архитектура Версия                Репозиторий
↳Размер
=====
Установка:
firefox              x86_64      102.14.0-2.el9_2.inferit appstream 107
↳М
Установка зависимостей:
sphere-indexhtml    noarch      9-3.el9                appstream 33 k
Результат транзакции
=====
```

```
Установка 2 Пакета
```

```
Объем загрузки: 107 М
```

```
Объем изменений: 276 М
```

```
Продолжить? [д/Н]: д
```

```
Загрузка пакетов:
```

```
(1/2): sphere-indexhtml-9-3.el9.noarch.rpm      170 kB/s | 33 kB
```

```
↳ 00:00
```

```
(2/2): firefox-102.14.0-2.el9_2.inferit.x86_64.rpm 7.7 MB/s | 107 MB
```

```
↳ 00:13
```

```
-----
```

```
Общий размер                                7.6 MB/s | 107 MB
```

```
↳ 00:14
```

```
Проверка транзакции
```

```
Проверка транзакции успешно завершена.
```

```
Идет проверка транзакции
```

```
Тест транзакции проведен успешно.
```

```
Выполнение транзакции
```

```
Подготовка      :                               1/1
```

```
Установка       : sphere-indexhtml-9-3.el9.noarch 1/2
```

```
Установка       : firefox-102.14.0-2.el9_2.inferit.x86_64 2/2
```

```
Запуск скриптлета: firefox-102.14.0-2.el9_2.inferit.x86_64 2/2
```

```
Проверка        : firefox-102.14.0-2.el9_2.inferit.x86_64 1/2
```

```
Проверка        : sphere-indexhtml-9-3.el9.noarch 2/2
```

```
Установлен:
```

```
firefox-102.14.0-2.el9_2.inferit.x86_64 sphere-indexhtml-9-3.el9.
```

```
↳noarch
```

```
Выполнено!
```

3.2.3 Обновить установленные пакеты

Для проверки наличия обновлений выполните следующую команду:

```
dnf check-upgrade
```

Пример работы команды:

```
[user@msvsphere test]$ dnf check-upgrade
MSVSphere 9 - AppStream          7.2 MB/s | 9.6 MB      00:01
MSVSphere 9 - BaseOS             4.2 MB/s | 3.6 MB      00:00
MSVSphere 9 - CRB                3.0 MB/s | 2.7 MB      00:00
MSVSphere 9 - Extras             1.8 MB/s | 989 kB       00:00

NetworkManager.x86_64           1:1.42.2-6.el9_2.inferit baseos
NetworkManager-ads1.x86_64      1:1.42.2-6.el9_2.inferit baseos
NetworkManager-bluetooth.x86_64 1:1.42.2-6.el9_2.inferit baseos
NetworkManager-libnm.x86_64     1:1.42.2-6.el9_2.inferit baseos
NetworkManager-team.x86_64      1:1.42.2-6.el9_2.inferit baseos
NetworkManager-tui.x86_64       1:1.42.2-6.el9_2.inferit baseos
NetworkManager-wifi.x86_64      1:1.42.2-6.el9_2.inferit baseos
NetworkManager-wwan.x86_64      1:1.42.2-6.el9_2.inferit baseos
```

Для обновления всей системы выполните следующую команду от имени администратора — **sudo**:

```
sudo dnf upgrade
```

Для обновления определённого пакета (и его зависимостей) выполните следующую команду от имени администратора — **sudo**:

```
sudo dnf upgrade имя_пакета
```

3.2.4 Удалить установленный пакет

Для удаления пакета выполните следующую команду:

```
dnf remove имя_пакета
```

Пример: удалим пакет `firefox.x86_64` (запустим от имени администратора — **sudo**):

```
sudo dnf remove firefox.x86_64
```

Результат работы команды:

```
[user@msvsphere test]$ sudo dnf remove firefox.x86_64
Зависимости разрешены.
```

```
=====
Пакет                Архитектура  Версия
↳Репозиторий        Размер
=====
Удаление:
firefox              x86_64      102.9.0-3.el9_1.inferit.3
↳@appstream         276 M
Удаление
неиспользуемых
зависимостей:
sphere-indexhtml    noarch      9-3.el9
↳@appstream         35 k
=====
```

Результат транзакции

Удаление 2 Пакета

Освобожденное место: 276 M

Продолжить? [д/Н]: д

Проверка транзакции

Проверка транзакции успешно завершена.

Идет проверка транзакции

Тест транзакции проведен успешно.

Выполнение транзакции

```
Подготовка          :                               1/1
Запуск скрипглета:  firefox-102.9.0-3.el9_1.inferit.3.x86_64  1/2
Удаление             :  firefox-102.9.0-3.el9_1.inferit.3.x86_64  1/2
Запуск скрипглета:  firefox-102.9.0-3.el9_1.inferit.3.x86_64  1/2
Удаление             :  sphere-indexhtml-9-3.el9.noarch          2/2
Запуск скрипглета:  sphere-indexhtml-9-3.el9.noarch            2/2
Проверка             :  firefox-102.9.0-3.el9_1.inferit.3.x86_64  1/2
Проверка             :  sphere-indexhtml-9-3.el9.noarch          2/2
```

Удален:

```
firefox-102.9.0-3.el9_1.inferit.3.x86_64  sphere-indexhtml-9-3.el9.
↳noarch
```

Вы можете увидеть, что также были удалены все пакеты, которые зависят от удаляемого.

3.2.5 Проверить целостность пакета

Для проверки целостности rpm-пакета выполните следующую команду:

```
rpm -V имя_rpm_пакета
```

В результате работы команды будет указана следующая информация:

- размер пакета
- полномочия
- тип
- владелец
- группа
- MD5-сумма
- дата последнего изменения пакета

3.2.6. Получить информацию об установленном пакете

Для получения подробной информации об установленном пакете выполните следующую команду:

```
dnf info имя_пакета
```

Пример работы команды для пакета `firefox.x86_64`:

```
[user@msvsphere test]$ dnf info firefox.x86_64
Установленные пакеты
Имя           : firefox
Версия        : 102.14.0
Выпуск        : 2.el9_2.inferit
Архитектура   : x86_64
Размер        : 276 M
Источник      : firefox-102.14.0-2.el9_2.inferit.src.rpm
Репозиторий   : @System
Из репозитор  : appstream
Краткое опис  : Mozilla Firefox Web browser
URL           : https://www.mozilla.org/firefox/
Лицензия      : MPLv1.1 or GPLv2+ or LGPLv2+
Описание      : Mozilla Firefox is an open-source web browser, designed
↳ for standards
               : compliance, performance and portability.
```

Рассмотрим основные операции с модулями.

3.2.7 Посмотреть список доступных модулей

Для просмотра списка доступных модулей выполните следующую команду:

```
dnf module list
```

3.2.8 Установить выбранный модуль

Для установки выбранного модуля выполните следующую команду (от имени администратора — `sudo`):

```
sudo dnf module install имя_модуля:версия
```

Например, для установки модуля `ruby:3.1` используйте следующую команду:

```
sudo dnf module install ruby:3.1
```

3.2.9 Удалить указанный модуль

Для удаления указанного модуля выполните следующую команду:

```
sudo dnf module remove имя_модуля:версия
```

Например, для удаления пакета `ruby:3.1` используйте следующую команду:

```
sudo dnf module remove ruby:3.1
```

3.2.10 Описание репозитория MSVSphere 9 Сервер

Рассмотрим репозитории MSVSphere 9 Сервер.

- **MSVSphere 9 - AppStream** — приложения общего назначения.
- **MSVSphere 9 - BaseOS** — базовый набор пакетов операционной системы.
- **MSVSphere 9 - CRB** — дополнительные пакеты для разработчиков.
- **MSVSphere 9 - Extras** — набор дополнительных приложений.
- **MSVSphere 9 - HighAvailability** — пакеты для создания кластеров высокой доступности.
- **MSVSphere 9 - NFV** — компоненты для виртуализации сетевых служб.

- **MSVSphere 9 - ResilientStorage** — пакеты для создания кластерных хранилищ.
- **MSVSphere 9 - RT** — набор пакетов для системы реального времени.

3.2.11 Посмотреть список включенных и доступных репозиториев

Для просмотра списка включенных репозиториев выполните следующую команду:

```
dnf repolist
```

Для просмотра списка включенных и отключенных репозиториев выполните следующую команду:

```
dnf repolist all
```

Для вывода подробного описания для каждого включенного репозитория выполните следующую команду:

```
dnf repolist -v
```

Для вывода списка отключенных репозиториев выполните следующую команду:

```
dnf repolist disabled
```

Для получения подробной информации о конкретном репозитории выполните следующую команду:

```
dnf repolist название репозитория -v
```

Пример: вывести подробную информацию о репозитории BaseOS:

```
dnf repolist BaseOS -v
```

```
ИД репозитория           : baseos
Имя репозитория          : MSVSphere 9 - BaseOS
Статус репозитория       : включено
Версия репозитория       : 9.3
Метки дистрибутива       : [cpe:/o:ncsd:msvsphere:9]: , 9, M, S, S,
→V, e, e, h, p, r
Репозиторий обновлен    : Пт 25 авг 2023 15:33:16
Пакеты репозитория      : 1 164
Пакеты-в-репозитории    : 1 164
Размер-репозитория      : 1.2 G
```

(продолжение на следующей странице)

(продолжение с предыдущей страницы)

```

Зеркала-репозитория      : https://mirrors.inferitos.ru/mirrorlist/9/
↳baseos
Базовый-URL-репозитория  : https://repo1.msvsphere-os.ru/msvsphere/9/
↳isos/x86_64/ (0 more)
Истечение срока репозитория: 86 400 секунд(а) (осталось: Пт 25 авг
↳2023 16:05:43)
Имя файла репозитория    : /etc/yum.repos.d/msvsphere-baseos.repo
Всего пакетов            : 1 164

```

Здесь мы видим, что репозиторий включен, количество пакетов в репозитории и его размер, а также другие важные параметры.

Зеркала репозитория — это серверы, дублирующие содержимое этого репозитория. Они позволяют снизить нагрузку с основных серверов.

3.2.12 Добавить в систему сторонний репозиторий

Иногда возникает необходимость установить приложение, которого нет в имеющихся репозиториях. В этом случае есть возможность добавить в систему сторонний репозиторий.

Важно: Рекомендуем быть предельно осторожными при подключении сторонних репозиториях и тщательно соблюдать меры безопасности.

Вы можете подключить сторонний репозиторий, если есть `.repo`-файл, с помощью следующей команды:

```
dnf config-manager --add-repo путь_к_.repo_файлу
```

Пример подключения `.repo`-файла `docker.io`:

```
dnf config-manager --add-repo https://download.docker.com/linux/rhel/
↳docker-ce.repo
```

3.2.13 Включить или отключить репозиторий

Вы можете по необходимости временно включать и отключать репозитории, чтобы установить приложение из конкретного репозитория. При этом репозиторий не будет удалён.

Команда включения репозитория:

```
sudo dnf config-manager --set-enabled имя_репозитория
```

Команда отключения репозитория:

```
sudo dnf config-manager --set-disabled имя_репозитория
```

При необходимости вы можете вывести справку по команде `config-manager`:

```
dnf config-manager --help-cmd
```

3.3 Безопасность

3.3.1 Использование сторонних репозиториев/пакетов

Так как сторонние репозитории и пакеты загружаются из Интернета, то при их скачивании и установке необходимо быть уверенными в безопасности устанавливаемых приложений. Важно быть уверенным, что никакая третья сторона не изменяла содержимое пакета при передаче его от автора к пользователю. Подписание пакета является способом защиты пакета для конечного пользователя. Поэтому репозитории и все пакеты в них подписываются специальным цифровым ключом.

Приватный ключ есть только у разработчиков. Публичный ключ может располагаться на сайте репозитория, либо распространяться вместе с операционной системой.

Разработчики подписывают пакеты приватным ключом, а с помощью публичного ключа конечный пользователь может убедиться, что это тот самый пакет и никакая третья сторона не изменяла его.

Ниже мы рассмотрим, как проверить цифровую подпись пакета.

3.3.2 Цифровые подписи пакетов и их проверка

Для проверки цифровой подписи пакета выполните следующую команду (находясь в папке с пакетом):

```
rpm --checksig имя_пакета.rpm
```

Пример: проверим цифровую подпись пакета `VirtualBox-7.0-7.0.10_158379_el9-1.x86_64.rpm`:

```
rpm --checksig VirtualBox-7.0-7.0.10_158379_el9-1.x86_64.rpm
```

Результат работы команды:

```
[user@msvsphere Загрузки]$ rpm --checksig VirtualBox-7.0-7.0.10_
↳158379_el9-1.x86_64.rpm
VirtualBox-7.0-7.0.10_158379_el9-1.x86_64.rpm: rsa sha1 (md5) pgp md5_
↳OK
```

Вы можете также использовать опцию `-v` для вывода более полной информации о проверке.

4 Идентификация и аутентификация

4.1 Введение

Средства идентификации и аутентификации предоставляют возможности идентификации объектов доступа, идентификации и проверки подлинности субъектов доступа при входе в систему и при доступе к защищаемым объектам, управления идентификаторами, в том числе их создания, присвоения и уничтожения, управления аутентификационными данными, в том числе их инициализации, защищенного хранения, блокирования и разблокирования, проверки соответствия аутентификационной информации заданной метрике качества, защиты обратной связи при вводе аутентификационной информации, а также другие возможности.

4.2 Добавление нового пользователя

Для добавления нового пользователя используется утилита `useradd`. Она позволяет добавить учетную запись нового пользователя. Режимы ее работы и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 1: Опции утилиты `useradd` и их значения

Опция	Значение
<code>-c, --comment</code>	Любая текстовая строка. Используется как поле для имени и фамилии пользователя, длина этого поля не должна превосходить 128 символов.
<code>-b, --base-dir</code>	Базовый системный каталог по умолчанию, если не указан другой каталог. Базовый каталог объединяется с именем учётной записи для определения домашнего каталога.
<code>-d, --home</code>	Для создаваемого пользователя в качестве начального каталога будет использован базовый каталог. По умолчанию это значение получается объединением имени пользователя с базовым каталогом и используется как имя домашнего каталога.
<code>-d, --home-dir</code>	Задать домашний каталог нового пользователя. Если данная опция не используется, то в качестве домашнего каталога выбирается каталог типа <code>/базовый_системный_каталог/имя_пользователя</code> .
<code>-D, --defaults</code>	Вывести значения стандартных опций.
<code>-e, --expiredate</code>	Дата окончания срока действия учётной записи пользователя. Задаётся в формате <code>ГГГГ-ММ-ДД</code> .

продолжение на следующей странице

Таблица 1 – продолжение с предыдущей страницы

Опция	Значение
--f, --inactive	Число дней, которые должны пройти после окончания срока действия пароля, чтобы учётная запись заблокировалась. Если указано значение 0 , то учётная запись блокируется сразу после окончания срока действия пароля, а при значении -1 данная возможность не используется. По умолчанию используется значение -1 .
-g, --gid	Название группы нового пользователя или её идентификационный номер. Указываемое название группы или её номер должны существовать в системе.
-G, --groups	Список дополнительных групп, в которых числится пользователь. Перечисление групп осуществляется через запятую без пробелов. На указанные группы действуют те же ограничения, что и для группы, указанной в опции -g .
-m, --create-home	Создает начальный домашний каталог нового пользователя, если он ещё не существует. Если каталог уже существует, добавляемый пользователь должен иметь права на доступ к указанному каталогу.
-M, --no-create-home	Позволяет не создавать домашний каталог нового пользователя.
-K, --key	Используется для изменения значений по умолчанию для параметров, хранимых в конфигурационном файле /etc/login.def .
-N, --no-user-group	Позволяет добавить нового пользователя в группу, указанную в опции -g или заданную по умолчанию в конфигурационном файле /etc/default/useradd , не создавая группу, название которой совпадает с именем нового пользователя. Если опции -g , -N , -U не указаны, то настройки групп по умолчанию определяются в конфигурационном файле /etc/login.defs .
-o, --non-unique	Позволяет создать учётную запись с уже имеющимся, не уникальным идентификатором.
-p, --password	Позволяет задать новый пароль для учетной записи.

продолжение на следующей странице

Таблица 1 – продолжение с предыдущей страницы

Опция	Значение
-r, --system	Позволяет создать системную учётную запись. По умолчанию для данной категории учетных записей домашний каталог не создаётся вне зависимости от значения соответствующего параметра конфигурационного файла <code>/etc/login.defs</code> . Для создания домашнего каталога системного пользователя необходимо вместе с опцией <code>-r</code> задать опцию <code>-m</code> .
-s, --shell	Полный путь к программе, используемой в качестве начального командного интерпретатора для пользователя сразу после регистрации. Длина этого поля не должна превосходить 256 символов. Если задать пустое значение, то будет использоваться оболочка по умолчанию.
-u, --uid	Позволяет задать идентификационный номер (численное неотрицательное значение идентификатора) пользователя. Это значение должно быть уникальным, если не задействована опция <code>-O</code> .
U, --user-group	Позволяет создать группу, название которой совпадает с именем пользователя, присоединив данного пользователя к этой группе.
-h, --help	Показать краткую справку об утилите.

Пример: создадим пользователя с именем `user` и зададим для него основную группу `users` и две дополнительные группы `ftp` и `developers`, к которым он будет приписан.

Для этого выполним следующую команду:

```
useradd -g users -G ftp,developers user
```

4.3 Изменение уже имеющихся пользовательских записей

Для изменения уже имеющихся пользовательских записей используется утилита `usermod`. Она позволяет изменить данные существующей учетной записи пользователя. Режимы ее работы и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 2: Опции утилиты **usermod** и их значения

Опция	Значение
-a, --append	Добавить пользователя в дополнительную группу. Следует использовать только вместе с параметром -G .
-c, --comment	Новое значение поля комментария.
-d, --home	Новый домашний каталог учетной записи. Если указан параметр -m , то содержимое текущего домашнего каталога будет перемещено в новый домашний каталог, который будет создан, если он ещё не существует.
-e, --expiredate	Установить дату окончания срока действия учетной записи в формате ГГГГ-ММ-ДД .
-f, --inactive	Установить пароль после окончания срока действия учетной записи в INACTIVE . Если указано значение 0 , то учётная запись блокируется сразу после окончания срока действия пароля, а при значении -1 данная возможность не используется. По умолчанию используется значение -1 .
-g, --gid	Принудительно назначить первичную группу.
-G, --groups	Список дополнительных групп.
-l, --login	Новое значение учетной записи.
-L, --lock	Заблокировать пароль пользователя. Это делается помещением символа ! в начало шифрованного пароля, что приводит к его блокировке. Не следует использовать этот параметр вместе с -p или -U .
-m, --move-home	Переместить содержимое домашнего каталога пользователя в новое место. Если новый домашний каталог не существует, то он создаётся автоматически. Данная опция используется только вместе с опцией -d .
-o, --non-unique.	При использовании с параметром -u этот параметр позволяет указывать не уникальный числовой идентификатор пользователя.
-p, --password	Задать новый пароль для учетной записи.
-s, --shell	Задать новую оболочку для учетной записи.
-u, --uid	Новый идентификационный номер для учетной записи.
-U, --unlock	Разблокировать учетную запись.

Пример: изменим срок действия учетной записи пользователя с идентификатором **user6**.

Для этого выполним следующую команду:

```
usermod -e 2020-05-01 user6
```

где 2020-05-01 — дата истечения срока действия учетной записи в формате ГГГГ-ММ-ДД.

Пример: изменим идентификатор (значение учётной записи) пользователя с `user6` на `user7`.

Для этого выполним следующую команду:

```
usermod -l user7 user6
```

4.4 Удаление пользователей

Для удаления пользователей используется утилита `userdel`. Она позволяет удалить существующую учетную запись пользователя. Режимы ее работы и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 3: Опции утилиты `userdel` и их значения

Опция	Значение
<code>-f, --force</code>	С этой опцией учётная запись будет удалена, даже если пользователь в этот момент работает в системе. Она также заставляет утилиту удалить домашний каталог пользователя и почтовый ящик, даже если другой пользователь использует тот же домашний каталог или если почтовый ящик не принадлежит данному пользователю. Внимание! Перед использованием этого параметра убедитесь в необходимости этого действия! Этот параметр может привести систему в нерабочее состояние!
<code>-r, --remove</code>	Файлы в домашнем каталоге пользователя будут удалены вместе с самим домашним каталогом и почтовым ящиком. Пользовательские файлы, расположенные в других файловых системах, нужно искать и удалять вручную.
<code>-n</code>	Задаёт, сколько месяцев идентификатор пользователя должен устаревать перед повторным использованием. Задайте <code>-1</code> , чтобы указать, что идентификатор пользователя никогда не должен использоваться повторно. Задайте <code>0</code> , чтобы указать, что идентификатор пользователя можно немедленно использовать повторно. Если опция <code>-n</code> не задана, то идентификатор будет устаревать стандартное количество месяцев перед повторным использованием.

продолжение на следующей странице

Таблица 3 – продолжение с предыдущей страницы

Опция	Значение
<code>-h, --help</code>	Показать краткую справку.

Пример: удалим пользователя с идентификатором `user7`.

Для этого выполним следующую команду:

```
userdel -r user7
```

4.5 Добавление группы пользователей

Для добавления группы пользователей используется утилита `groupadd`. Режимы ее работы и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 4: Опции утилиты `groupadd` и их значения

Опция	Значение
<code>-f</code>	Вернуть статус успешного выполнения, если группа уже существует. Если используется вместе с параметром <code>-g</code> и указанный идентификатор группы уже существует, то выбирается другой уникальный идентификатор группы, то есть параметр <code>-g</code> игнорируется.
<code>-g</code>	Числовое значение идентификатора группы. Значение должно быть уникальным, если не задан параметр <code>-o</code> . Значение должно быть не отрицательным. По умолчанию берётся значение больше 999 и больше идентификатора любой другой группы. Значения от 0 и до 999 обычно зарезервированы под системные группы.
<code>-K</code>	Изменить значения по умолчанию для параметров, которые хранятся в конфигурационном файле <code>/etc/login.defs</code> .
<code>-o</code>	Разрешить добавление группы с не уникальным идентификатором.
<code>-r, --system</code>	Создать системную группу.
<code>-h, --help</code>	Показать краткую справку.

Пример: создадим группу `group2` с числовым значением идентификатора `8285`.

Для этого выполним следующую команду:

```
groupadd group2 -g 8285
```

4.6 Изменение существующей группы пользователей

Для изменения существующей группы пользователей используется утилита **groupmod**. Режимы ее работы и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 5: Опции утилиты **groupmod** и их значения

Опция	Значение
-g, --gid	Изменить идентификатор группы.
-n, --new-name	Изменить имя группы.
-o, --non-unique	Позволяет использовать не уникальный идентификатор группы.
-p, --password	Изменить пароль.
-h, --help	Показать краткую справку.

Пример: изменим идентификатор группы пользователей **users** на **ftp**.

Для этого выполним следующую команду:

```
groupmod -g ftp users
```

4.7 Удаление существующей группы пользователей

Для удаления существующей группы пользователей используется утилита **groupdel**. Утилита позволяет удалить определение группы из системы путем удаления записи о соответствующей группе из файла **/etc/group**. Однако она не удаляет идентификатор группы из файла паролей. Удаленный идентификатор действует для всех файлов и каталогов, которые его имели.

Пример: удалим группу с именем **group3**.

Для этого выполним следующую команду:

```
groupdel group3
```

4.8 Создание и изменение пароля пользователя

Для создания и изменения пароля пользователя (в том числе для блокировки учетной записи пользователя) используется утилита **passwd**. Обычный пользователь может изменить пароль только своей учётной записи, суперпользователь **root** может изменить пароль любой учётной записи.

При изменении пароля проверяется информация об устаревании пароля, чтобы убедиться, что пользователю разрешено изменять пароль в настоящий момент. Если выяснится, что не разрешено, то утилита не производит изменение пароля и завершает работу.

При изменении пароля пользователь должен будет сначала ввести старый пароль, если он был. Введенное пользователем значение старого пароля зашифровывается и сравнивается со значением зашифрованного текущего пароля. Затем пользователю необходимо будет дважды ввести новый пароль. Значение второго ввода сравнивается с первым, и они должны совпасть. После этого пароль тестируется на сложность подбора, т.е. его значение не должно быть легко угадываемым.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 6: Опции утилиты **passwd** и их значения

Опция	Значение
-a, --all	Эту опцию можно использовать только вместе с -S для вывода статуса всех пользователей.
-d, --delete	Удалить пароль пользователя (сделать его пустым). Это быстрый способ заблокировать пароль учётной записи.
-e, --expire	Немедленно сделать пароль устаревшим. Это заставит пользователя изменить пароль при следующем входе в систему.
-i, --inactive	Эта опция используется для блокировки учётной записи по прошествии заданного числа дней после устаревания пароля. То есть если пароль устарел и прошло больше дней, чем указано, то пользователь больше не сможет использовать свою учётную запись.
-l, --lock	Заблокировать указанную учётную запись. Эта опция блокирует учётную запись путем изменения значения пароля на такое, которое не может быть ранее указанным зашифрованным паролем.

продолжение на следующей странице

Таблица 6 – продолжение с предыдущей страницы

Опция	Значение
<code>-m, --mindays</code>	Задать минимальное количество дней между сменой пароля. Нулевое значение этого поля указывает на то, что пользователь может менять свой пароль тогда, когда захочет.
<code>-S, --status</code>	Показать состояние учётной записи. Информация о состоянии содержит семь полей. Первое поле содержит имя учётной записи. Второе поле указывает, заблокирована ли учётная запись, она без пароля или у неё есть рабочий пароль. Третье поле хранит дату последнего изменения пароля. В следующих четырёх полях хранятся минимальный срок, максимальный срок, период выдачи предупреждения и период неактивности пароля. Все эти сроки измеряются в днях.
<code>-u, --unlock</code>	Разблокировать указанную учётную запись. Этот параметр активирует учётную запись путем изменения пароля на прежнее значение, которое было перед использованием параметра <code>-l</code> .
<code>-w, --warndays</code>	Установить число дней выдачи предупреждения, перед тем как потребуются смена пароля.
<code>-x, --maxdays</code>	Установить максимальное количество дней, в течение которых пароль остаётся рабочим, после чего его надо будет изменить.
<code>-h, --help</code>	Показать краткую справку.

Пример: зададим пароль пользователю `user4`. Работа команды `passwd`:

```
[root@msvsphere ~]# passwd user4
Изменяется пароль пользователя user4.
Новый пароль :
Повторите ввод нового пароля :
passwd: все данные аутентификации успешно обновлены.
[root@msvsphere ~]#
```

Пример: посмотрим состояние учётной записи `user4`. Работа команды `passwd`:

```
[root@msvsphere /]# passwd -S user4
user4 PS 2023-07-04 0 99999 7 -1 (Пароль задан, шифр SHA512.)
[root@msvsphere /]#
```

Где:

- `user4` — имя пользователя.

- **PS** — статус пароля.
- **2023-07-04** — отображает время последнего изменения пароля.
- **0** и **99999** — минимальный и максимальный срок действия пароля.
- **7** — срок вывода предупреждения.
- **-1** — срок деактивации пароля.

4.9 Изменение срока действия учётной записи и пароля пользователя

Утилита **chage** позволяет установить дату завершения срока действия учетной записи пользователя, минимальный и максимальный срок действия пароля, дату завершения срока действия пароля, а также количество дней, в течение которых пользователю будут выводиться предупреждения о приближении завершения срока действия пароля.

Командой **chage** может пользоваться только суперпользователь, за исключением использования её с параметром **-l**, который позволяет непривилегированным пользователям определить время, когда истекает их личный пароль или учетная запись.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 7: Опции утилиты **chage** и их значения

Опция	Значение
-m	Меняет значение mindays на минимальное число дней между сменой пароля. Значение 0 в этом поле обозначает, что пользователь может изменять свой пароль когда угодно.
-M	Меняет значение maxdays на максимальное число дней, в течение которых пароль будет действителен. Когда сумма maxdays и lastday меньше, чем текущий день, у пользователя будет запрошен новый пароль до начала работы в системе.
-d	Меняет значение lastday на день, когда пароль был изменен последний раз (число дней с 1 января 1970). Дата также может быть указана в формате ГГГГ-ММ-ДД.

продолжение на следующей странице

Таблица 7 – продолжение с предыдущей страницы

Опция	Значение
-E	Используется для задания даты, с которой учетная запись пользователя станет недоступной. Дата также может быть указана в формате ГГГГ-ММ-ДД.
-I	Используется для задания количества дней «неактивности», то есть дней, когда пользователь вообще не входил в систему, после которых его учетная запись будет заблокирована. Значение 0 отключает этот режим.
-W	Используется для задания числа дней, когда пользователю начнет выводиться предупреждение об истечении срока действия его пароля и необходимости его изменения.
-l	Просмотреть текущую информацию о дате истечения срока действия пароля для пользователя.

Пример: посмотрим текущую информацию о дате истечения срока действия пароля для пользователя `user4`. Работа команды `chage`:

```
[root@msvsphere ~]# chage -l user4
Последний раз пароль был изменён: мар 12, 2023
Срок действия пароля истекает: никогда
Пароль будет деактивирован через: никогда
Срок действия учётной записи истекает: никогда
Минимальное количество дней между сменой пароля: 0
Максимальное количество дней между сменой пароля: 99999
Количество дней с предупреждением перед деактивацией пароля: 7
[root@msvsphere ~]#
```

4.10 Получение сведений о пользователе

Утилита `id` позволяет получить сведения об указанном пользователе или о текущем пользователе, запустившем данную утилиту, если он не указал явно имя пользователя.

По умолчанию выводятся числовые идентификаторы пользователя и группы, действующие идентификаторы пользователя и группы, а также идентификаторы других групп, в которых состоит пользователь.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 8: Опции утилиты `id` и их значения

Опция	Значение
<code>-g, --group</code>	Выводит только подлинный числовой идентификатор групп.
<code>-G, --groups</code>	Выводит все подлинные числовые идентификаторы групп, в которых состоит пользователь.
<code>-n, --name</code>	Выводит действующие имена пользователей или групп.
<code>-r, --real</code>	Выводит подлинные числовые идентификаторы пользователей или групп.
<code>-u, --user</code>	Выводит только подлинный числовой идентификатор пользователя.
<code>--version</code>	Выводит информацию о версии утилиты и завершает работу.
<code>--help</code>	Выводит справку по этой утилите и завершает работу.

Пример: выведем сведения о текущем пользователе `user`. Работа команды `id`:

```
[user@msvsphere ~]$ id
uid=1000(user) gid=1000(user) группы=1000(user),100(users)
→КОНТЕКСТ=user_u:user_r:user_t:s0
```

4.11 Конфигурационный файл `/etc/login.defs`

Конфигурационный файл `/etc/login.defs` позволяет задавать параметры, определяющие использование пользователями своих паролей.

```
# Password aging controls:
#
#     PASS_MAX_DAYS   Maximum number of days a password may be used.
#     PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#     PASS_MIN_LEN    Minimum acceptable password length.
#     PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7
```

Рис. 1: Фрагмент конфигурационного файла `/etc/login.defs`

Параметры перечислены в таблице:

Таблица 9: Параметры конфигурационного файла /
etc/login.defs и их описание

Параметр	Описание
PASS_MAX_DAYS	Определяет максимальный срок действия пароля, т.е. максимальное число дней, в течение которых действие пароля сохраняется. По истечении этого срока запускается процесс принудительной смены пароля. Если значение параметра не задано, то есть параметр закомментирован символом # или ему присвоено значение -1, то данное ограничение не установлено (отменяется).
PASS_MIN_DAYS	определяет минимальный срок между изменениями пароля, т.е. минимальное число дней между двумя последовательными изменениями пароля. Если значение параметра не задано, то есть параметр закомментирован символом # или ему присвоено значение -1, то данное ограничение не установлено (отменяется).
PASS_MIN_LEN	Определяет минимальную допустимую длину задаваемого пароля.
PASS_WARN_AGE	Определяет, за сколько дней до истечения срока действия пароля начнётся вывод предупреждения о необходимости его смены. Если значение параметра не задано, то есть параметр закомментирован символом # или ему присвоено значение -1, то данное ограничение не установлено (отменяется). Если значение параметра 0, то предупреждение о необходимости смены пароля будет выведено в день его устаревания.

Пример: зададим максимальное количество дней действия пароля, равное 30 суткам:

```
[root@msvsphere ~]# cat /etc/login.defs | grep PASS_MAX_DAYS
# PASS_MAX_DAYS Maximum number of days a password may be used.
PASS_MAX_DAYS 30
[root@msvsphere ~]#
```

4.12 Конфигурационный файл /etc/pam.d/system-auth

Конфигурационный файл /etc/pam.d/system-auth позволяет задавать настройки подключаемых модулей аутентификации.

```
# Generated by authselect on Fri Jul 14 14:08:55 2023
# Do not modify this file manually.

auth      required      pam_env.so
auth      required      pam_faildelay.so delay=20000
00
auth      sufficient    pam_fprintd.so
auth      [default=1 ignore=ignore success=ok] pam_usertype.so isregular
auth      [default=1 ignore=ignore success=ok] pam_localuser.so
auth      sufficient    pam_unix.so nullok
auth      [default=1 ignore=ignore success=ok] pam_usertype.so isregular
auth      sufficient    pam_sss.so forward_pass
auth      required      pam_deny.so

account   required      pam_unix.so
account   sufficient    pam_localuser.so
account   sufficient    pam_usertype.so issystem
account   [default=bad success=ok user_unknown=ignore] pam_sss.so
account   required      pam_permit.so
```

Рис. 2: Фрагмент конфигурационного файла /etc/pam.d/system-auth

Каждая строка в нем представляет собой правило, состоящее из трёх обязательных полей и одного опционального. Поля разделены символом пробела. Порядок, в котором указаны правила, определяет очередность их проверки.

Синтаксис правила:

```
type control module-path [module-arguments]
```

Поле **type** задаёт тип вызываемого модуля и может принимать одно из четырех допустимых значений:

- **auth** — предназначен для аутентификации пользователя путём запроса и проверки его пароля;
- **account** — используется для контроля доступа к сервису/приложению. Например, может быть произведён запрос о том, не истёк ли срок действия аккаунта пользователя, разрешено ли пользователю работать с определённым сервисом в определённое время, хватает ли системных ресурсов для работы;
- **password** — применяется для установки/изменения паролей;
- **session** — управляет действиями пользователя в рамках активной сессии после его успешной аутентификации в системе.

Поле `control` задаёт действие, которое нужно выполнить после вызова модуля. Доступно несколько действий:

- **required** — модуль должен вернуть положительный ответ. Если он возвращает отрицательный ответ, то пользователь будет уведомлен об этом только после того, как все остальные модули данного типа будут проверены;
- **requisite** — требует от модуля положительный ответ. В случае получения отрицательного ответа последовательная проверка выполнения остальных правил моментально прекращается и пользователь получает сообщение об ошибке аутентификации;
- **sufficient** — в случае, если ни один из модулей с действием **required** или **sufficient**, расположенных перед текущим, не вернул отрицательного ответа, текущий модуль вернёт положительный ответ и все последующие модули будут проигнорированы;
- **optional** — результат проверки модуля важен только в том случае, если действие является единственным для данного модуля;
- **include** — предназначается для добавления строк заданного типа из других файлов конфигурации из каталога `/etc/pam.d/` в файл конфигурации `/etc/pam.d/system-auth`. Название файла указывается в качестве аргумента действия.

Поле `module-path` задаёт путь к вызываемому модулю.

Поле `module-arguments` — дополнительные необязательные параметры модуля, необходимые для определения действий некоторых отдельных модулей в случае успешной авторизации. Так, если в конфигурационном файле найти строку, содержащую `pam_pwquality.so`, и добавить в нее `minlen=8`, то будет установлена минимальная длина пароля, равная 8-ми символам.

Пример: В качестве примера сделаем блокировку учетной записи пользователя, который совершит определенное количество неудачных попыток входа в систему.

Для этого внесем в файл `/etc/pam.d/system-auth` следующие изменения:

1. Сначала допишем в секцию `auth` строку `auth required pam_tally2.so deny=2 onerr=fail`, т.е. подключим модуль `pam_tally2` и установим блокировку пользователя после двух (значение параметра `deny`) неудачных попыток входа.
2. Затем в секции `account` добавим строку `account required pam_tally2.so` и закомментируем строки вида `auth requisite pam_succeed_if.so uid >= 1000 quiet` и `auth required pam_deny.so`.
3. Потом строку `auth sufficient pam_unix.so nullok try_first_pass` заменим на `auth required pam_unix.so nullok try_first_pass`.

После этого пользователь, допустивший подряд две неверных попытки входа, на третьей получит сообщение о том, что его учетная запись заблокирована. И даже если четвертой попыткой он введет верный пароль, то все равно не получит доступ к системе.

```
[user@msvsphere ~]$ su user2
Пароль:
su Сбой при проверке подлинности
[user@msvsphere ~]$ su user2
Пароль:
su Сбой при проверке подлинности
[user@msvsphere ~]$ su user2
Пароль:
Учетная запись заблокирована как следствие неудачных попыток входа
→(всего 3)
su Сбой при проверке подлинности
[user@msvsphere ~]$ su user2
Пароль:
Учетная запись заблокирована как следствие неудачных попыток входа
→(всего 4)
su Сбой при проверке подлинности
[user@msvsphere ~]$
```

Пример: В качестве другого примера настроим проверку паролей на сложность подбора через `pam_cracklib`.

1. Для этого добавим или изменим следующую строку:

```
password requisite pam_cracklib.so try_first_pass retry=3 type=
→minlen=6 dcredit=-2 ucredit=-3 lcredit=-2 ocredit=-1
```

Это значит следующее:

- после трех неуспешных попыток (`retry=3`) модуль вернет ошибку;
- минимальная длина для пароля — 6 символов (`minlen=6`);
- минимальное количество цифр — 2 (`dcredit=-2`);
- минимальное количество символов верхнего регистра — 3 (`ucredit=-3`);
- минимальное количество символов нижнего регистра — 2 (`lcredit=-2`);
- минимальное количество других символов — 1 (`ocredit=-1`).

2. Удалим или закомментируем следующую строку:

```
password requisite pam_pwquality.so try_first_pass local_users_only
→retry=3 authtok_type=
```

Результат

- Выполним команду **passwd** для смены пароля пользователя **user2**.
- Зададим пароль из трех символов и увидим сообщение «НЕУДАЧНЫЙ ПАРОЛЬ: слишком короткий».
- Зададим пароль из четырех символов, система выдаст сообщение «НЕУДАЧНЫЙ ПАРОЛЬ: короткий».
- Зададим пароль из шести символов (букв и цифр), в результате чего получим сообщение «НЕУДАЧНЫЙ ПАРОЛЬ: слишком простой».
- После трех неуспешных попыток модуль вернет ошибку.

```
[user2@msvsphere user]$ passwd
Изменяется пароль пользователя user2.
Смена пароля для user2.
(текущий) пароль Unix:
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: слишком короткий
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: короткий
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: слишком простой
passwd: Использовано максимальное число попыток, заданное для службы
```

- Зададим пароль достаточной длины из одних цифр и получим сообщение «НЕУДАЧНЫЙ ПАРОЛЬ: не содержит достаточное число РАЗЛИЧНЫХ символов».
- Зададим пароль достаточной длины, содержащий все указанные требования, кроме включения в него отличных от алфавита и цифр символов. Например, **2QyfM0b4**. Получим сообщение «НЕУДАЧНЫЙ ПАРОЛЬ: слишком простой».

```
[user2@msvsphere user]$ passwd
Изменяется пароль пользователя user2.
Смена пароля для user2.
(текущий) пароль Unix:
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: не содержит достаточное число РАЗЛИЧНЫХ символов
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: короткий
Новый пароль:
НЕУДАЧНЫЙ ПАРОЛЬ: слишком простой
passwd: Использовано максимальное число попыток, заданное для службы
```

- Зададим пароль, соблюдая все установленные требования. Например, **2QyfM*0b4**. Пароль будет успешно задан (см. листинг).

```
[user2@locmsvspherealhost user]$ passwd
Изменяется пароль пользователя user2.
Смена пароля для user2.
(текущий) пароль Unix:
Новый пароль:
Повторите ввод нового пароля:
passwd: Все данные аутентификации успешно обновлены.
```

4.13 Конфигурационный файл `/etc/issue`

Конфигурационный файл `/etc/issue` позволяет задать текстовое содержание уведомления пользователю перед началом его идентификации и аутентификации для входа в систему. Например, с предупреждением о том, что в ней реализованы меры защиты информации и о необходимости соблюдения соответствующих правил обработки данных. Традиционно в конфигурационном файле присутствуют опции выдачи сведений об операционной системе и ядре. Дополнительно можно добавить опции выдачи текущих даты и времени, количества работающих пользователей и некоторых других сведений.

4.14 Конфигурационный файл `/etc/shadow`

Конфигурационный файл `/etc/shadow` содержит сведения об учетных записях и паролях пользователей в виде строк со следующей структурой:

```
username:$id$salt$hashed:lastchanged:min:max:warn:inactive:expire
```

Структура файла:

- **username** — имя пользователя;
- **id** — алгоритм шифрования: 1 (алгоритм MD5), 5 (SHA-256), 6 (SHA-512);
- **salt** — «соль», добавляемая к паролю строка из 10-20 случайных символов;
- **hashed** — зашифрованный пароль;
- **lastchanged** — дата последнего изменения пароля;
- **min** — минимальное число дней между двумя последовательными сменами паролей;
- **max** — срок действия пароля, т.е. максимальное число дней, в течение которых пароль будет активен;
- **warn** — за какое количество дней до срока истечения действия пароля пользователь будет уведомлен о том, что его необходимо сменить;

- **inactive** — количество дней после истечения срока действия пароля, спустя которое его учётная запись блокируется;
- **expire** — число дней, прошедших с момента блокирования учетной записи.

Если после имени пользователя **username** вместо **\$id\$salt\$hashed** стоит символ * либо последовательность из двух символов !!, то это означает, что попытки входа в систему от имени данного пользователя заблокированы.

5 Управление доступом

5.1 Введение

Средства управления доступом предоставляют возможности ограничения количества одновременно предоставляемых параллельных сеансов доступа пользователей к системе, блокирования сеанса доступа пользователя в систему после истечения установленного периода времени бездействия или по его запросу, поддержки и сохранения атрибутов безопасности, связанных с информацией в процессе её хранения и обработки, разделения полномочий пользователей и администраторов, обеспечивающих функционирование системы, реализации различных методов управления доступом, типов доступа и правил разграничения доступа, назначения приоритетов для использования субъектами доступа вычислительных ресурсов, квотирования предоставляемых вычислительных ресурсов, а также другие возможности.

5.2 Установка и изменение прав доступа к файлам и директориям

Утилита `chmod` позволяет устанавливать и изменять права доступа к файлам и директориям. Она принимает описания прав доступа в двух нотациях: численной и буквенной, описываемой ниже.

В соответствии с буквенной нотацией пользователи, которые могут потенциально работать с файлом, разделяются на владельца (**u**), группу владельцев (**g**) и всех остальных пользователей (**o**), а файл может быть читаемым (**r**), записываемым (**w**) и исполняемым (**x**).

Описание прав доступа начинается с символа, соответствующего типу пользователей. Затем идет символ **+** для установки или символ **-** для снятия прав доступа, после чего описание заканчивается последовательностью символов, соответствующей правам доступа.

Например, для определения прав доступа, позволяющих читать и модифицировать файл `file`, может использоваться нотация:

```
chmod g+rw file
```

Для удаления всех прав доступа на директорию `/directory` для группы и остальных пользователей может использоваться нотация:

```
chmod go-rwx /directory.
```

Утилита поддерживает также следующие опции, перечисленные в таблице:

Таблица 10: Опции утилиты **chmod** и их значения

Опция	Значение
-R, --recursive	Рекурсивное изменение прав доступа для директорий и их содержимого.
-c, --changes	Подробно описывать действия для каждого файла, чьи права действительно изменяются.
-f, --silent, --quiet	Не выдавать сообщения об ошибке для файлов, чьи права не могут быть изменены.
-v, --verbose	Подробно описывать действие или отсутствие действия для каждого файла.
--version	Сообщить информацию о версии.
--help	Выводит справку по этой утилите и завершает работу.

Пример: сменим права для файла **file1** так, чтобы владелец файла имел права на чтение и запись, а группа и остальные пользователи — только на чтение:

```
[user@msvsphere ~]$ chmod u+rw g-wx o-wx file1
```

5.3 Назначение и изменение владельца файла и директории

Утилита **chown** позволяет назначить или изменить владельца файла или директории.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 11: Опции утилиты **chown** и их значения

Опция	Значение
-R, --recursive	Рекурсивное изменение прав доступа для директорий и их содержимого.
-c, --changes	Подробно описывать все изменения.
-f, --silent, --quiet	Не выдавать сообщения об ошибке.
-v, --verbose	Вывести подробное описание действия.
--version	Сообщить информацию о версии.
--help	Выводит справку по этой утилите и завершает работу.

Пример: назначим пользователя **user** владельцем файла **file**:

```
chown user file
```

Пример: выполним рекурсивный обход директории **directory** и назначим пользователя **user** владельцем всех вложенных файлов:

```
chown -R user directory
```

5.4 Изменение группы-владельца файла или директории

Утилита `chgrp` позволяет изменить группу-владельца файла или директории.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 12: Опции утилиты `chgrp` и их значения

Опция	Значение
<code>-R, --recursive</code>	Рекурсивное изменение группы для каталогов и всего их содержимого.
<code>-c, --changes</code>	Подробно описывать действия для каждого файла, чья группа действительно меняется.
<code>-f, --silent, --quiet</code>	Не выдавать сообщения об ошибке для файлов, чья группа не может быть изменена.
<code>-v, --verbose</code>	Подробно описывать действие или отсутствие действия для каждого файла.
<code>--version</code>	Сообщить информацию о версии.
<code>--help</code>	Вывести справку по этой утилите и завершить работу.

Пример: изменим группу-владельца файла `file` на новую группу `new_group`:

```
chgrp new_group file
```

5.5 Просмотр и изменение списков правил контроля доступа для файлов и директорий

Утилита `setfacl` позволяет просматривать и изменять списки правил контроля доступа для файлов и директорий.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 13: Опции утилиты `setfacl` и их значения

Опция	Значение
<code>-d</code>	Установить правила контроля доступа по умолчанию.
<code>-k</code>	Удалить правила контроля доступа по умолчанию.

продолжение на следующей странице

Таблица 13 – продолжение с предыдущей страницы

Опция	Значение
-s	Заменить правила контроля доступа заданными.
-m	Модифицировать правила контроля доступа.
-x	Удалить указанное правило контроля доступа.
-b	Удалить все правила контроля доступа.
-v	Вывести версию и выйти.
-h	Вывести справку об использовании утилиты и выйти.

Пример: удалим все правила контроля доступа к файлу `file`:

```
setfacl -b file
```

5.6 Просмотр списков контроля доступа

Утилита `getfacl` позволяет просматривать списки контроля доступа.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 14: Опции утилиты `getfacl` и их значения

Опция	Значение
-a, --access	Выводить список контроля доступа к файлам.
-d, --default	Выводить список контроля доступа по умолчанию.
-c, --omit-header	Не выводить заголовков с комментариями.
e, --all-effective	Выводить комментарии с действующими правами доступа для каждого пользователя.
-E, --no-effective	Не выводить комментарии с действующими правами доступа ни для одного пользователя.
-R, --recursive	Делать рекурсивный обход директории и выводить списки контроля доступа для каждого файла и директории.
-v, --version	Вывести версию и выйти.
-h, --help	Вывести справку об использовании утилиты и выйти.

Пример: посмотрим список контроля доступа для файла `cg.conf`:

```
[user@msvsphere ~]$ getfacl cg.conf
# file: cg.conf
# owner: user
# group: user
user::rwx
```

(продолжение на следующей странице)

(продолжение с предыдущей страницы)

```
group::r-x
other::r-x
```

Пример: зададим дополнительные компоненты списка контроля доступа для пользователя `user` и группы `user` по отношению к файлу `cg.conf`:

```
[user@msvsphere ~]$ setfacl -m g:user:rxw cg.conf
[user@msvsphere ~]$ setfacl -m u:user:rxw cg.conf
[user@msvsphere ~]$ getfacl cg.conf
# file: cg.conf
# owner: user
# group: user
user::rxw
user:user:rxw
group::r-x
group:user:rxw
mask::rxw
other::r-x
```

Пример: от имени администратора модифицируем списки контроля доступа для файлов, владельцем которых он является:

```
[root@msvsphere ~]# setfacl -m u:user:rxw ~/file2
[root@msvsphere ~]#
[root@msvsphere ~]# getfacl ~/file2
getfacl: Removing leading `/` from absolute path names
# file: root/file2
# owner: root
# group: root
user::rw-
user:user:rxw
group::r--
mask::rxw
other::r--
```

Пример: от имени администратора модифицируем списки контроля доступа для файлов, владельцем которых он не является:

```
[root@msvsphere ~]# setfacl -m u:user:rxw /home/user3/file2
[root@msvsphere ~]# setfacl -m u:user:rxw /home/user3/dir2
[root@msvsphere ~]#
[root@msvsphere ~]# getfacl /home/user3/file2
getfacl: Removing leading `/` from absolute path names
# file: home/user3/file2
# owner: user3
```

(продолжение на следующей странице)

(продолжение с предыдущей страницы)

```
# group: user3
user::rwx
user:user:rwx
group:---
mask::rwx
other:---

[root@msvsphere ~]# getfacl /home/user3/dir2
getfacl: Removing leading `/` from absolute path names
# file: home/user3/dir2/
# owner: user3
# group: user3
user::rwx
user:user:rwx
group:---
mask::rwx
other:---
```

Пример: от имени администратора удалим списки контроля доступа для объектов, владельцем которых он является:

```
[root@msvsphere ~]# setfacl -b ~/file2
[root@msvsphere ~]# getfacl ~/file2
getfacl: Removing leading `/` from absolute path names
# file: root/file2
# owner: root
# group: root
user::rw-
group::r--
other::r--
```

Пример: от имени администратора удалим списки контроля доступа для объектов, владельцем которых он не является:

```
[root@msvsphere ~]# setfacl -b /home/user3/file2
[root@msvsphere ~]# getfacl /home/user3/file2
getfacl: Removing leading `/` from absolute path names
# file: home/user3/file2
# owner: user3
# group: user3
user::rwx
group:---
other:---
```

Важно: Пользователь, не обладающий полномочиями администратора, не может

удалять списки контроля доступа, которые он не создавал.

5.7 Редактирование пользовательских квот для файловой системы

Утилита **edquota** позволяет редактировать пользовательские квоты для файловой системы.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 15: Опции утилиты **edquota** и их значения

Опция	Значение
-u, --user	Изменить пользовательскую квоту.
-g, --group	Изменить групповую квоту.
-p, --prototype protoname	= Дублировать квоты прототипного пользователя. Это обычный механизм, используемый для инициализации квот для групп пользователей.
-F, --format имя-формата	= Изменить квоту для указанного формата.
-f, --filesystem	Выполнять указанные операции только для заданной файловой системы. По умолчанию операция выполняется для всех файловых систем с квотой.
-t, --edit-period	Редактировать мягкие ограничения по времени для каждой файловой системы.
-T, --edit-times	Изменить время для пользователя или группы, когда принудительное ограничение установлено.

5.8 Конфигурационный файл `/etc/profile`

Конфигурационный файл `/etc/profile` используется для задания элементов окружения оболочки пользователя. Например, в нём определяются глобальные переменные:

- **PATH** — переменная среды, используемая для указания оболочке списка каталогов, которые будут просматриваться при поиске исполняемых файлов;
- **USER** — имя пользователя при входе в ОС;
- **LOGNAME** — то же, что и **USER**. Некоторые программы считывают значение этой глобальной переменной вместо **USER**;

- **MAIL** — имя файла, в который записывается локальная почта пользователя, а также его расположение;
- **HOSTNAME** — имя хоста;
- **HISTSIZE** — количество исполненных команд, сохраняемых в истории;
- **HISTCONTROL** — политики в отношении команд, сохраняемых в истории. По умолчанию задано значение **ignoredups**, то есть команда, полностью совпадающая с одной из уже записанных в историю, не сохраняется. Если задать политику **ignorespace**, то будут игнорироваться как дублирующиеся команды, так и те, что начинаются с символа пробела.

Также в конфигурационном файле задаётся маска, используемая для определения конечных прав доступа для пользователя.

```

# /etc/profile

# System wide environment and startup programs, for login setup
# Functions and aliases go in /etc/bashrc

# It's NOT a good idea to change this file unless you know what you
# are doing. It's much better to create a custom.sh shell script in
# /etc/profile.d/ to make custom changes to your environment, as this
# will prevent the need for merging in future updates.

pathmunge () {
    case ":{PATH}:" in
        *:"$1":*)
            ;;
        *)
            if [ "$2" = "after" ] ; then
                PATH=$PATH:$1
            else
                PATH=$1:$PATH
            fi
    esac
}

if [ -x /usr/bin/id ]; then
    if [ -z "$EUID" ]; then
        # ksh workaround
        EUID=`/usr/bin/id -u`
        UID=`/usr/bin/id -ru`
    fi
    USER="`/usr/bin/id -un`"
    LOGNAME=$USER
    MAIL="/var/spool/mail/$USER"
fi

# Path manipulation
if [ "$EUID" = "0" ]; then
    pathmunge /usr/sbin
    pathmunge /usr/local/sbin
else
    pathmunge /usr/local/sbin after
    pathmunge /usr/sbin after
fi

```

Рис. 3: Фрагмент конфигурационного файла `/etc/profile`

Пример: определим время бездействия при локальной терминальной сессии равным двум минутам (120 с). Для этого в файле `/etc/profile` после строк

```
HOSTNAME= '/usr/bin/hostname 2>/dev/null'
HISTSIZE=1000
```

Добавим строку `TMOUТ=120`. Там же, в строке

```
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL
```

Необходимо добавить параметр `TMOUТ`:

```
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE TMOUТ HISTCONTROL
```

Для подтверждения вступления изменений в силу надо будет завершить сеанс и зарегистрироваться в системе заново. Тогда появится сообщение, что после двух минут бездействия время ожидания ввода вышло, в результате чего интерактивный сеанс был закрыт.

```
[user@msvsphere ~]$ su
Пароль
[root@msvsphere user]# timed out waiting for input: auto-logout
[user@msvsphere ~]$
```

5.9 Конфигурационный файл `/etc/security/limits.conf`

Конфигурационный файл `/etc/security/limits.conf` может использоваться для задания модулю `pam_limits.so` дополнительных ограничений. Для этого каждая его строка включает четыре группы параметров, которые перечислены и описаны ниже:

Важно: По умолчанию все ограничения отключены — все строки закомментированы.

- **<domain>:**

имя пользователя, имя группы с синтаксисом `@group`, подстановочный знак `*` для записи по умолчанию, подстановочный знак `%`, который также может использоваться с синтаксисом `%group` для ограничения `maxlogin`;

- **<type>:**

- `soft` для установки мягких ограничений;
- `hard` для установки жестких ограничений.

- **<item>:**

- `core`: ограничивает размер файла ядра в Кб;

- **data**: максимальный размер данных в Кб;
- **fsize**: максимальный размер файла в Кб;
- **memlock**: максимальное адресное пространство, предустановленное в памяти, в Кб;
- **nofile**: максимальное количество открытых файлов;
- **rss**: максимальный размер резидентного набора в Кб;
- **stack**: максимальный размер стека в Кб;
- **cpu**: максимальное время процессора в MIN;
- **nproc**: максимальное количество процессов;
- **as**: ограничение адресного пространства в Кб;
- **maxlogins**: максимальное количество логинов для этого пользователя;
- **maxsyslogins**: максимальное количество входов в систему;
- **priority**: приоритет процессов пользователя;
- **locks**: максимальное количество блокировок файлов, которое может быть обеспечено пользователем;
- **sigpending**: максимальное количество ожидающих сигналов;
- **msgqueue**: максимальный объем памяти, используемый очередями сообщений POSIX, в байтах;
- **nice**: приоритет для запуска процессов утилитой **nice**;
- **rtprio**: максимальный приоритет в реальном времени.

```

# /etc/security/limits.conf
#
#This file sets the resource limits for the users logged in via PAM.
#It does not affect resource limits of the system services.
#
#Also note that configuration files in /etc/security/limits.d directory,
#which are read in alphabetical order, override the settings in this
#file in case the domain is the same or more specific.
#That means, for example, that setting a limit for wildcard domain here
#can be overridden with a wildcard setting in a config file in the
#subdirectory, but a user specific setting here can be overridden only
#with a user specific setting in the subdirectory.
#
#Each line describes a limit for a user in the form:
#
#<domain>          <type> <item> <value>
#
#Where:
#<domain> can be:
#   - a user name
#   - a group name, with @group syntax
#   - the wildcard *, for default entry
#   - the wildcard %, can be also used with %group syntax,
#       for maxlogin limit
#
#<type> can have the two values:
#   - "soft" for enforcing the soft limits
#   - "hard" for enforcing hard limits
#
#<item> can be one of the following:
#   - core - limits the core file size (KB)
#   - data - max data size (KB)
#   - fsize - maximum filesize (KB)
#   - memlock - max locked-in-memory address space (KB)
#   - nofile - max number of open file descriptors

```

Рис. 4: Фрагмент конфигурационного файла /etc/security/limits.conf

Пример: ограничим число параллельных сеансов доступа для каждой учетной записи пользователя. Для этого добавим в конфигурационный файл строку следующего содержания:

```
username hard maxlogins 2
```

Тогда, при условии, что пользователь **username** открыл локальную сессию (учитывая, что при входе в графический сеанс открываются сразу две сессии пользователя) и попытался зайти в систему через ssh-соединение (потенциально ещё один активный сеанс), ему будет выведено сообщение **Too many logins for 'username'** и это соединение будет заблокировано.

5.10 Конфигурационный файл `/etc/fstab`

Конфигурационный файл `/etc/fstab` используется для настройки параметров монтирования различных блочных устройств, разделов на диске и файловых систем. Он состоит из набора так называемых определений, каждое из которых занимает свою строку и состоит из шести полей, разделённых пробелами или символами табуляции:

```
fs_spec fs_file fs_vfstype fs_mntops fs_freq fs_passno
```

Поля предназначены для задания следующих параметров:

- **fs_spec**

Физическое размещение файловой системы, по которому определяется конкретный раздел или устройство хранения для монтирования. Вместо указания размещения файловой системы явным образом можно воспользоваться её уникальным идентификатором UUID.

- **fs_file**

Точка монтирования, куда монтируется корень файловой системы.

- **fs_vfstype**

Тип файловой системы. Поддерживаются следующие типы: `adfs`, `affs`, `autofs`, `coda`, `coherent`, `cramfs`, `devpts`, `efs`, `ext2`, `ext3`, `ext4`, `hfs`, `hpfs`, `iso9660`, `jfs`, `minix`, `msdos`, `nvpfs`, `nfs`, `ntfs`, `proc`, `qnx4`, `reiserfs`, `romfs`, `smbfs`, `sysv`, `tmpfs`, `udf`, `ufs`, `umsdos`, `vfat`, `xenix`, `xf`.

- **fs_mntops**

Опции монтирования файловой системы. Основные опции: `defaults`, `noauto`, `user`, `owner`, `comment`, `nofail`.

- **fs_freq**

Предназначено для использования утилитой создания резервных копий в файловой системе. Возможные значения: `0` и `1`. Если указано `1`, то утилита создаст резервную копию.

- **fs_passno**

Предназначено для использования программой `fsck` при необходимости проверки целостности файловой системы; возможные значения: `0`, `1` и `2`. Значение `1` указывается только для корневой файловой системы (то есть файловой системы с точкой монтирования `/`). Для остальных файловых систем для проверки утилитой `fsck` задаётся значение `2`. При значении `0` — проверка выполняться не будет.

По умолчанию конфигурационный файл включает:

```
/dev/mapper/MSVSphere-root / xfs defaults 0 0
```

Файловая система `/dev/mapper/MSVSphere-root` примонтирована в каталог `/`,

тип файловой системы — **xfс**, используемые опции — **defaults**, резервная копия данных не создаётся (**fs_freq=0**), проверка целостности файловой системы не выполняется (**fs_passno=0**).

```
UUID=b1bfe9b0-96ea-4876-883c-a9f1b6c74b /boot ext4 defaults 1 2
```

Файловая система с идентификатором **b1bfe9b0-96ea-4876-883c-a9f1b6c74b** смонтирована в **/boot**, тип файловой системы — **ext4**, используемые опции — **defaults**, резервная копия данных создаётся (**fs_freq=1**), проверка целостности файловой системы выполняется (**fs_passno=2**).

```
/dev/mapper/MSVSphere-swap swap defaults 0 0
```

Файловая система **/dev/mapper/MSVSphere-swap** является разделом подкачки **swap**, используемые опции — **defaults**, резервная копия данных не создаётся (**fs_freq=0**), проверка целостности файловой системы не выполняется (**fs_passno=0**).

```
# /etc/fstab
# Created by anaconda on Tue Jun 20 11:58:05 2023
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/msvsphere-root /                xfs      defaults    0 0
UUID=8e41c721-164e-455c-bd65-b60ad5ad7cb4 /boot          xfs      defaults
```

Рис. 5: Фрагмент конфигурационного файла `/etc/fstab`

6 Регистрация событий безопасности

6.1 Введение

Средства регистрации событий безопасности предоставляют возможности включения и исключения событий безопасности в совокупность событий, подвергающихся регистрации, регистрации событий безопасности; предоставления регистрируемой информации в понятном и защищенном от несанкционированного доступа виде; обеспечения непрерывности процесса регистрации при превышении журналом регистрации определенного размера; выборочного просмотра, поиска, сортировки и упорядочения регистрируемой информации; изготовления соответствующих отчетов, а также другие возможности.

6.2 Создание и удаление правил регистрации событий безопасности

Утилита `auditctl` позволяет формировать, добавлять или удалять правила регистрации событий безопасности.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 16: Опции утилиты `auditctl` и их значения

Опция	Значение
<code>-b backlog</code>	Установить максимальное количество доступных для записи данных регистрации буферов. Значение по умолчанию — 64.
<code>--backlog_wait_time</code>	Установить время ожидания до постановки новой партии данных регистрации событий безопасности в очередь и последующей их обработки при достижении предельного значения.
<code>-e [0..2]</code>	Установить флаг блокировки: <code>0</code> позволит на время отключить регистрацию, <code>1</code> позволит включить её обратно, а <code>2</code> позволит защитить конфигурацию регистрации от изменений.

продолжение на следующей странице

Таблица 16 – продолжение с предыдущей страницы

Опция	Значение
-f [0..2]	Установить способ обработки для флага сбоя. Эта опция позволяет определить, каким образом ядро будет обрабатывать критические ошибки. Значение по умолчанию: 1. Для систем с повышенными требованиями к безопасности значение 2 может быть более предпочтительным.
-h	Краткая помощь по аргументам командной строки.
-i	Игнорировать ошибки при чтении правил из файла.
-l	Вывести список всех правил по одному правилу в строке.
-k ключ	Установить на правило ключ фильтрации. Ключ фильтрации — это произвольная текстовая строка длиной не больше 31 символа. Ключ помогает уникально идентифицировать записи, генерируемые в ходе аудита за точкой наблюдения.
-m текст	Послать в систему регистрации событий пользовательское сообщение. Это возможно только из аккаунта учетной записи суперпользователя root .
-p [r w x a]	Установить фильтр прав доступа для точки наблюдения. r (чтение), w (запись), x (исполнение), a (изменение атрибута).
-r частота	Установить ограничение скорости выдачи сообщений в секунду (0 — нет ограничения). Если эта частота ненулевая, и она превышает в ходе аудита, флаг сбоя выставляется ядром для выполнения соответствующего действия. Значение по умолчанию: 0.
-R файл	Читать правила из файла. Правила должны быть расположены по одному в строке и в том порядке, в каком они должны исполняться. Владельцем файла с правилами должен быть суперпользователь root . Данный файл не должен быть доступен для чтения любым другим пользователям, в противном случае операция с опцией не будет позволена.
-s	Получить статус регистрации событий.
-a список, действие	Добавить правило с указанным действием к концу списка.
-A список, действие	Добавить правило с указанным действием в начало списка.

продолжение на следующей странице

Таблица 16 – продолжение с предыдущей страницы

Опция	Значение
-d список, действие	Удалить правило с указанным действием из списка. Правило удаляется только в том случае, если полностью совпали и имя системного вызова, и поля сравнения.
-D	Удалить все правила и точки наблюдения.
-c	Продолжить загружать правила, несмотря на появление ошибки. Таким образом можно отследить конечный результат загрузки правил. Если хотя бы одно из правил не загрузилось, код возврата будет ненулевой.
-S [Имя или номер системного вызова all]	Любой номер или имя системного вызова может быть использован. Также возможно использование ключевого слова all . Если какой-либо процесс выполняет указанный системный вызов, то служба регистрации генерирует соответствующую запись. Если значения полей сравнения заданы, а системный вызов не указан, правило будет применяться ко всем системным вызовам. В одном правиле может быть задано несколько системных вызовов — это положительно сказывается на производительности, поскольку заменяет обработку нескольких правил.
F [n=v n!=v n<v n>v n<=v n>=v n&v n&=v]	Задать поле сравнения для правила. Атрибуты поля следующие: объект, операция, значение. Можно задать до 64 полей сравнения в одной команде. Каждое новое поле должно начинаться с -F . Служба регистрации событий будет генерировать запись, если произошло совпадение по всем полями сравнения. Допустимо использование одного из следующих 8 операторов: равно, не равно, меньше, больше, меньше либо равно, больше либо равно, битовая маска (n&v) и битовая проверка (n&=v). Битовая проверка выполняет операцию and над значениями и проверяет, равны ли они. Битовая маска просто выполняет операцию and . Поля, оперирующие с идентификатором пользователя, могут также работать с именем пользователя — программа автоматически получит идентификатор пользователя из его имени.
-A list,action	Добавить правило в начало списка list с действием action .

продолжение на следующей странице

Таблица 16 – продолжение с предыдущей страницы

Опция	Значение
<code>-C [f=f f!=f]</code>	Сравнить значения полей между собой. Формат задания сравнения: поле, оператор, поле. Можно в одной команде сравнивать несколько пар полей одновременно. Перед каждой новой парой записывается опция <code>-C</code> . Опция снабжена двумя операторами: <code>=</code> и <code>!=</code> . Доступные для сравнения поля: <code>-w путь</code> (добавить точку наблюдения за файловым объектом, находящимся по указанному пути) и <code>-W путь</code> (удалить точку наблюдения за файловым объектом, находящимся по указанному пути).

Пример: добавим правило аудита, осуществляющее наблюдение за доступом к файлу `/etc/profile`:

```
auditctl -w /etc/profile -p rw -k profile
```

6.3 Добавление правила регистрации событий безопасности

Утилита **autrace** позволяет добавлять правила регистрации событий безопасности для того, чтобы следить за использованием системных вызовов в указанном процессе. Она поддерживает опцию `-r`, с помощью которой можно ограничить сбор информации о системных вызовах только теми, которые необходимы для анализа использования ресурсов.

Пример: с помощью утилиты **autrace** от имени администратора получим информацию из журналов аудита:

```
[root@msvsphere ~]# autrace /bin/date
Waiting to execute: /bin/date
Пн апр 9 22:56:19 MSK 2023
Cleaning up
Trace complete. You can locate the records with 'ausearch -i -p 12438'
```

6.4 Поиск данных регистрации событий безопасности

Утилита `ausearch` используется для поиска данных регистрации событий безопасности по различным критериям.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 17: Опции утилиты `ausearch` и их значения

Опция	Значение
<code>-a, --event audit-event-id</code>	Искать события с заданным идентификатором события.
<code>-c, --comm comm-name</code>	Искать события с заданным именем исполняемого файла.
<code>-f, --file file-name</code>	Искать события с заданным именем файла.
<code>-tm, --terminal terminal</code>	Искать события с заданным терминалом.
<code>-x, --executable executable</code>	Искать события с заданной исполняемой программой.
<code>--session Login-Session-ID</code>	Искать события с заданным идентификатором сессии.
<code>-ua, --uid-all all-user-id</code>	Искать события, у которых любой из идентификаторов пользователя совпадает с заданным идентификатором пользователя.
<code>-ue, --uid-effective effective-user-id</code>	Искать события с заданным эффективным идентификатором пользователя.
<code>-ui, --uid user-id</code>	Искать события с заданным идентификатором пользователя.
<code>-ga, --gid-all all-group-id</code>	Искать события с заданным эффективным или обычным идентификатором группы.
<code>-ge, --gid-effective effective-group-id</code>	Искать события с заданным эффективным идентификатором группы или именем группы.
<code>-gi, --gid group-id</code>	Искать события с заданным идентификатором группы или именем группы.
<code>-hn, --host host-name</code>	Искать события с заданным именем узла. Имя узла может быть именем узла, полным доменным именем или цифровым сетевым адресом.
<code>-k, --key key-string</code>	Искать события с заданным ключевым словом.
<code>-p, --pid process-id</code>	Искать события с заданным идентификатором процесса.
<code>-pp, --ppid parent-process-id</code>	Искать события с заданным идентификатором родительского процесса.

продолжение на следующей странице

Таблица 17 – продолжение с предыдущей страницы

Опция	Значение
<code>-sc, syscall-name-or-value</code>	<code>--success</code> Искать события с заданным системным вызовом.
<code>-o, SE-Linux-context-string</code>	<code>--object</code> Искать события с заданным объектом SELinux.
<code>-e, SE-Linux-context-string</code>	<code>--context</code> Искать события с заданным контекстом SELinux.
<code>-s, SE-Linux-context-string</code>	<code>--subject</code> Искать события с заданным субъектом SELinux.
<code>-sv, success-value</code>	<code>--success</code> Искать события с заданным флагом успешного выполнения. Допустимые значения: yes (успешно) и no (неудачно).
<code>-te, [end-date] [end-time]</code>	<code>--end</code> Искать события, которые произошли раньше или во время указанной временной точки.
<code>-ts, [start-date] [start-time]</code>	<code>--start</code> Искать события, которые произошли после или во время указанной временной точки.
<code>-w, --word</code>	Совпадение с полным словом. Поддерживается для имени файла, имени узла, терминала и контекста SELinux.
<code>-uu, uuid_гостевой_системы</code>	<code>--uuid</code> Искать событие в гостевой ОС с заданным UUID.
<code>-vm, имя_гостевой_системы</code>	<code>--vm-name</code> Искать событие в гостевой ОС с заданным именем.
<code>--just-one</code>	Остановить поиск после появления первого события, удовлетворяющего критериям поиска.
<code>-e, exit-code-or-errno</code>	<code>--exit</code> Искать события по заданному системному вызову: коду возврата или номеру ошибки.
<code>--input-logs</code>	Использовать место нахождения файла логов, обозначенное в <code>/etc/audit/auditd.conf</code> .
<code>-h, --help</code>	Выдать справку об утилите.

6.5 Генерация отчетов по данным регистрации событий безопасности

Утилита `aureport` позволяет генерировать отчеты по данным регистрации событий безопасности.

Утилита поддерживает следующие опции, перечисленные в таблице:

Таблица 18: Опции утилиты **aureport** и их значения

Опция	Значение
-u, --user	Отчет о пользователях.
-e, --event	Отчет о событиях.
-f, --file	Отчет о файлах.
-p, --pid	Отчет о процессах.
-s, --syscall	Отчеты о системных вызовах.
-t, --log	Отчет о временных рамках отчета.
-x, --executable	Отчет об исполняемых объектах.
-tm, --terminal	Отчет о терминалах.
-l, --login	Отчет о попытках входа в систему.
-au, --auth	Отчет о всех попытках аутентификации.
-c, --config	Отчет об изменениях конфигурации.
-m, --mods	Отчет об изменениях пользовательских учетных записей.
--tty	Отчёт о нажатиях пользователя на клавиатуре.
-ma, --mac	Отчет о событиях в системе мандатного управления доступом.
--success	Для обработки в отчетах выбирать только удачные события. По умолчанию показываются и удачные, и неудачные события.
--failed	Для обработки в отчетах выбирать только неудачные события. По умолчанию показываются и удачные, и неудачные события.
-te, --end [дата] [время]	Искать события, которые произошли раньше или во время указанной временной точки.
-ts, --start [дата] [время]	Искать события, которые произошли после или во время указанной временной точки.
--node имя_узла	Выбрать события, связанные с узлом, имя которого указано после ключа. Можно указать несколько имён узлов. По умолчанию информация собирается со всех узлов.
--summary	Генерировать итоговый отчет, который дает информацию только о количестве элементов в том или ином отчете.
--input-logs	Использовать место нахождения файла логов, обозначенное в <code>/etc/audit/auditd.conf</code> .

6.6 Конфигурационный файл `/etc/audit/auditd.conf`

Конфигурационный файл `/etc/audit/auditd.conf` содержит параметры настройки средств регистрации событий безопасности, в том числе:

Параметр	Описание
<code>log_file</code>	Полное имя файла, в котором будут храниться данные регистрации событий безопасности.
<code>log_group</code>	Группа, являющаяся владельцем файла регистрации.
<code>log_format</code>	Формат хранения данных регистрации. Возможные значения: <code>raw</code> (данные записываются в том виде, в каком они были получены от ядра операционной системы) и <code>nolog</code> (запись данных отключается).
<code>priority_boost</code>	Приоритет выполнения службы регистрации.
<code>flush</code>	Режим работы службы регистрации. Возможные значения: <code>none</code> (не использовать какие-либо политики записи, т.е. дополнительные действия), <code>incremental</code> (запись с периодичностью, определенной параметром <code>freq</code>), <code>data</code> (запись данных в синхронном режиме), <code>sync</code> (запись в синхронном режиме и данных, и метаданных файла).
<code>freq</code>	Максимальное число регистрационных записей, которые могут храниться в буфере перед записью буферизованных данных на диск. Используется, только когда параметр <code>flush</code> имеет значение <code>incremental</code> .
<code>num_logs</code>	Максимальное число файлов регистрации на диске. Используется, только когда параметр <code>max_log_file_action</code> имеет значение <code>rotate</code> . Значение параметра не должно превышать 99.
<code>disp_qos</code>	Режим передачи данных между службой регистрации и диспетчером. Возможные значения: <code>lossy</code> (блокирование запрещено, т.е. служба регистрации может не передавать диспетчеру некоторые данные о событиях, если очередь данных о событиях полна. При этом данные регистрации будут записаны на диск, если только значение параметра <code>log_format</code> не равно <code>nolog</code>), <code>lossless</code> (блокирование разрешено, т.е. запись данных регистрации о событиях на диск будет остановлена, пока не освободится место в очереди).
<code>dispatcher</code>	Место расположения исполняемого файла программы диспетчера.

продолжение на следующей странице

Таблица 19 – продолжение с предыдущей страницы

Параметр	Описание
<code>name_format</code>	Порядок разрешения имен хостов. Возможные значения: <code>none</code> (имя не используется), <code>hostname</code> (имя, возвращенное через запрос <code>gethostname</code>), <code>fqd</code> (полное имя хоста, возвращенное через DNS запрос), <code>numeric</code> (IP-адрес), <code>user</code> (строка, определенная в параметре <code>name</code>).
<code>max_log_file</code>	Максимальный размер файла регистрации в мегабайтах, по достижении которого будет выполнено действие, определенное параметром <code>max_log_file_action</code> . Возможные действия: <code>ignore</code> (ничего не делать), <code>syslog</code> (отправить предупреждение в syslog), <code>suspend</code> (остановить запись данных регистрации событий на диск), <code>rotate</code> (произвести ротацию файлов регистрации в соответствии с параметром <code>num_logs</code>), <code>keep_logs</code> (осуществить ротацию, не удаляя при этом старые файлы).
<code>space_left</code>	Величина в мегабайтах, определяющая размер оставшегося дискового пространства, по достижении которого будет выполнено действие, определенное параметром <code>space_left_action</code> . Возможные действия: <code>ignore</code> (ничего не делать), <code>syslog</code> (отправить предупреждение в syslog), <code>email</code> (отправить письмо аккаунту, определенному в <code>action_mail_acct</code>), <code>exec</code> (выполнить скрипт), <code>suspend</code> (остановить запись на диск и перевести систему в single mode), <code>halt</code> (выключить систему).
<code>admin_space_left</code>	Величина в мегабайтах оставшегося свободного пространства на диске для предупреждения администратора о том, что надо добавить/очистить свободное пространство. Величина должна быть меньше чем <code>space_left</code> . Действия, которые можно определить в <code>admin_space_left_action</code> , аналогичны <code>space_left_action</code> .
<code>disk_full_action</code>	Действия, выполняемые при заполнении всего дискового пространства. Аналогичны <code>space_left_action</code> .
<code>disk_error_action</code>	Действия, выполняемые при возникновении дисковой ошибки. Аналогичны <code>space_left_action</code> .

7 Ограничение программной среды

7.1 Введение

Средства ограничения программной среды предоставляют возможности установки программного обеспечения доверенным образом; применения типовых наборов различных программных конфигураций; управления запуском программного обеспечения, в том числе определения запускаемых программ, настройки параметров запуска и контроля за их запуском; реагирования на попытки запуска, произведенные в нарушение установленных правил, а также другие возможности.

7.2 Включение программ в автозагрузку

Утилита `chkconfig` позволяет включать программы в автозагрузку с целью их автоматического запуска при старте операционной системы.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 20: Опции утилиты `chkconfig` и их значения

Опция	Значение
<code>--level levels</code>	Определяет уровни, на которых соответствующая программа должна выполняться. Уровни указываются на месте параметра <code>levels</code> в качестве строки целочисленных значений в диапазоне от 0 до 6. Так, например, при передаче опции <code>--level 35</code> утилите будет передано указание на уровни 3 и 5 соответственно.
<code>--no-redirect</code>	Если утилита запущена в системе, использующей утилиту <code>systemd</code> в качестве системы инициализации, то <code>chkconfig</code> будет перенаправлять команды в <code>systemd</code> , если у данной службы существует соответствующий файл, предназначенный для таких обращений. Данная опция отключает процесс перенаправления утилите <code>systemd</code> и обеспечивает работу только с символьными ссылками в директориях <code>/etc/rc[0-6].d</code> .
<code>--add name</code>	Добавляет новую службу для управления утилитой <code>chkconfig</code> . Имя службы указывается на месте параметра <code>name</code> .

продолжение на следующей странице

Таблица 20 – продолжение с предыдущей страницы

Опция	Значение
<code>--del name</code>	Удаляет службу, имя которой указывается на месте параметра <code>name</code> , из-под управления утилитой <code>chkconfig</code> . Также из директорий <code>/etc/rc[0-6].d</code> удаляются любые символьные ссылки, указывающие на удаляемую службу.
<code>--override name</code>	Производит переопределение настроек службы, имя которой указывается на месте параметра <code>name</code> , вместо базовых настроек.
<code>--list name</code>	Выводит все службы, доступные для <code>chkconfig</code> , а также показывает их статус на каждом уровне (вкл/выкл). Если опции передать аргументом имя некоторой службы, которое указывается на месте параметра <code>name</code> , то будет выведена информация только об указанной службе.

7.3 Управление системными службами

Утилита `systemctl` позволяет управлять системными службами.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 21: Опции утилиты `systemctl` и их значения

Опция	Значение
<code>-t, --type</code>	Указывает на тип так называемого юнита (службы, сокета, устройства и т.п.). Может быть в виде списка наименований типов, разделенных запятой, если требуется указать более, чем на один тип.
<code>-a, --all</code>	При выведении списка юнитов вывести абсолютно все загруженные юниты вне зависимости от их статуса, включая те из них, которые являются неактивными.
<code>start [имя сервиса]</code>	Запускает работу сервиса с указанным именем.
<code>stop [имя сервиса]</code>	Останавливает работу сервиса с указанным именем.
<code>reload [имя сервиса]</code>	Перезагружает конфигурацию сервиса с указанным именем.
<code>restart [имя сервиса]</code>	Перезапускает сервис с указанным именем.

продолжение на следующей странице

Таблица 21 – продолжение с предыдущей страницы

Опция	Значение
<code>try-restart</code> <code>[имя сервиса]</code>	Перезапускает сервис с указанным именем, если данный сервис уже работает на момент запуска утилиты.
<code>reload-or-restart</code> <code>[имя сервиса]</code>	Перезагрузить конфигурацию сервиса с указанным именем, если сервис поддерживает такую команду, или выполнить перезапуск службы. Если на момент запуска утилиты указанная служба не была запущена, то она запустится после успешного выполнения команды.
<code>reload-or-try-restart</code> <code>[имя сервиса]</code>	Перезагрузить конфигурацию сервиса с указанным именем, если сервис поддерживает такую команду, или выполнить перезапуск службы. Если на момент запуска утилиты указанная служба не была запущена, то указанная команда не произведет никаких действий.
<code>kill</code> <code>[имя сервиса]</code>	Осуществить принудительную остановку работы службы с указанным именем.
<code>is-active</code> <code>[имя сервиса]</code>	Осуществляет проверку, активна ли на момент запуска утилиты служба с указанным именем. Если служба активна, или хотя бы одна из служб, переданных в качестве аргумента данной команде, активна (в случае, если были переданы наименования более, чем одной службы), выведется нулевое значение. В противном случае — ненулевое.
<code>is-failed</code> <code>[имя сервиса]</code>	Проверяет, были ли проблемы при запуске указанной службы или служб. Если хотя бы у одной из служб возникали проблемы, будет выведено нулевое значение.
<code>enable</code> <code>[имя сервиса]</code>	Добавляет указанный сервис (или их множество) в автозапуск.
<code>disable</code> <code>[имя сервиса]</code>	Убирает указанный сервис (или их множество) из автозапуска.
<code>is-enabled</code> <code>[имя сервиса]</code>	Проверяет, находится ли указанная служба (или службы, в случае, если в качестве аргумента был передан список наименований) в автозапуске. Если хотя бы одна из указанных служб находится в автозапуске, будет выведено нулевое значение.
<code>--version</code>	Вывести информацию о версии утилиты.
<code>-h, --help</code>	Вывести справочную информацию об утилите.

Пример: проверим статус сервера печати.

Для этого выполним следующую команду:

```
[root@msvsphere ~]# systemctl status cups
cups.service - CUPS Printing Service
  Loaded: loaded (/usr/lib/systemd/system/cups.service; disabled;
  ↳ vendor preset: enabled)
  Active: inactive (dead)
```

Пример: разрешим автоматический запуск сервера печати CUPS при загрузке системы.

Для этого выполним следующую команду:

```
[root@msvsphere ~]# systemctl enable cups
Created symlink from /etc/systemd/system/multi-user.target.wants/cups.
  ↳ service to /usr/lib/systemd/system/cups.service.
Created symlink from /etc/systemd/system/printer.target.wants/cups.
  ↳ service to /usr/lib/systemd/system/cups.service.
Created symlink from /etc/systemd/system/sockets.target.wants/cups.
  ↳ service to /usr/lib/systemd/system/cups.socket.
Created symlink from /etc/systemd/system/multi-user.target.wants/cups.
  ↳ path to /usr/lib/systemd/system/cups.path.
```

7.4 Настройка запуска программ по расписанию

Утилита **crontab** позволяет настраивать запуск программ по расписанию.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 22: Опции утилиты **crontab** и их значения

Опция	Значение
-u	Указывает пользователя, чье расписание должно редактироваться.
-l	Вывод текущего файла расписания.
-r	Удаление текущего файла расписания.
-e	Редактирование файла расписания.

Таблица расписания состоит из шести колонок, разделяемых пробелами или символами табуляции. Первые пять колонок задают время выполнения (минута, час, день, месяц, день недели). В них может находиться число, список чисел, разделённых запятыми, диапазон чисел, разделённых дефисом, символы * или /. После полей времени указывается пользователь, от которого запускается программа. Все остальные символы в строке интерпретируются как выполняемая программа с её параметрами.

Пример: установим с помощью утилиты **crontab** ограничения на доступ к системе по времени, с 10:28 до 10:30. Команда **passwd -l user2** блокирует возможность авторизации, дописывая символ восклицательного знака к строке пароля в файле **/etc/shadow**. Команда **passwd -u user2** производит обратную операцию, снимая тем самым блокировку. Заполним файл расписания и выполним команду **service crond restart**:

```
[root@msvsphere user]# crontab -e
crontab: Installing new crontab
[root@msvsphere user]# service crond restart
Redirecting to /bin/systemctl restart crond.service
[root@msvsphere user]# crontab -l
28 10 * * * /usr/bin/passwd -l user2
30 10 * * * /usr/bin/passwd -u user2
[root@msvsphere user]#
```

7.5 Управление программными пакетами

Утилита **rpm** позволяет управлять так называемыми программными пакетами, т.е. управлять их установкой, обновлением, проверкой и удалением.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 23: Опции утилиты **rpm** и их значения

Опция	Значение
-i, --install	Установка нового пакета.
-u, --upgrade	Установка или обновление уже установленного пакета до новой версии. При этом после установки пакета все другие версии удаляются.
-f, --freshen	Обновление пакета, но только если предыдущая версия уже установлена.
--nodeps	Не выполнять проверку зависимостей перед установкой или обновлением пакета.
--nosuggest	Не предлагать пакет(ы) для разрешения отсутствующих зависимостей.
--noorder	Не выполнять переупорядочивание пакетов для установки. Список пакетов обычно переупорядочивается для удовлетворения зависимостей.
--oldpackage	Разрешает обновить или заменить пакет более старой версией.

продолжение на следующей странице

Таблица 23 – продолжение с предыдущей страницы

Опция	Значение
<code>--replacefiles</code>	Установить пакеты, даже если они заменяют файлы от других установленных пакетов.
<code>--replacepkgs</code>	Установить пакеты, даже если они уже установлены в систему.
<code>--includedocs</code>	Устанавливать файлы с документацией.
<code>--excludedocs</code>	Не устанавливать файлы с документацией.
<code>-e, --erase</code>	Удалить заданный пакет.
<code>--allmatches</code>	Удалить все версии пакета.
<code>--nodeps</code>	Не проверять зависимости перед удалением пакетов.
<code>--test</code>	Выполнить только проверку установки пакета.
<code>-q, --query</code>	Вывести информацию о пакете.
<code>-a, --all</code>	Выполняет запрос ко всем установленным пакетам.
<code>--changelog</code>	Вывести информацию об изменениях в пакете.
<code>-l, --list</code>	Вывести список файлов в пакете.
<code>-P, --provides</code>	Вывести функциональность, предоставляемую пакетом.
<code>-R, --requires</code>	Вывести пакеты, от которых зависит этот пакет.
<code>-v, --verify</code>	Выполнить проверку метаданных пакета и его контрольной суммы.
<code>--version</code>	Вывести номер версии утилиты.
<code>--help</code>	Вывести справку об использовании утилиты.

7.6 Установка последней версии пакета/группы пакетов

Утилита `dnf` используется для установки последней версии пакета или группы пакетов с учетом существующих зависимостей.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 24: Опции утилиты `dnf` и их значения

Опция	Значение
<code>install</code>	Используется для установки последней версии пакета с учетом существующих зависимостей.
<code>reinstall</code>	Используется для переустановки пакета с идентичной версией.
<code>update</code>	Используется для обновления всех пакетов в системе.
<code>download</code>	Используется для загрузки пакета из репозитория.

продолжение на следующей странице

Таблица 24 – продолжение с предыдущей страницы

Опция	Значение
<code>downgrade</code>	Используется для понижения версии пакета с версии, установленной на данный момент, до предыдущей самой высокой версии или указанной версии.
<code>remove</code>	Используется для удаления указанных пакетов из системы, а также для удаления пакетов, зависящих от удаляемых пакетов.
<code>info</code>	Используется для вывода описаний и общей информации о доступных пакетах.
<code>search</code>	Используется для поиска пакетов.
<code>list</code>	Используется для вывода различной информации о доступных пакетах.
<code>repolist all</code>	Используется для вывода списка всех репозиториев.
<code>clean</code>	Используется для удаления различных данных, накапливающихся со временем в кэше утилиты.
<code>history</code>	Используется для вывода истории использования утилиты.
<code>groupinstall</code>	Используется для установки последней версии всех пакетов из группы с учетом существующих зависимостей.
<code>groupupdate</code>	Используется для обновления всех пакетов из группы.
<code>groupremove</code>	Используется для удаления всех пакетов из группы.
<code>groupinfo</code>	Используется для вывода списка пакетов, относящихся к группе.
<code>grouplist</code>	Используется для вывода имен всех существующих групп пакетов.
<code>provides</code>	Используется, чтобы выяснить, какой пакет предоставляет тот или иной файл.
<code>repoquery --requires</code>	Вывести зависимости неустановленного пакета.
<code>repoquery --requires --resolve</code>	Вывести список пакетов, которые необходимы для удовлетворения зависимостей.
<code>-v, --verbose</code>	Запустить с большим количеством отладочной информации.
<code>-d, --debuglevel</code>	Устанавливает уровень отладки.
<code>-h, --help</code>	Вывести справку и выйти.

8 Стирание данных

8.1 Введение

Средства стирания данных предоставляют возможности безвозвратного удаления ставших ненужными данных и обеспечения недоступности остаточной информации путем многократной перезаписи использованных мест памяти специальными последовательностям.

8.2 Заполнение случайными числами места, занятого файлами

Утилита **shred** позволяет заполнять случайными числами место, занятое файлами.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 25: Опции утилиты **shred** и их значения

Опция	Значение
-f, --force	Изменить права для разрешения записи, если это необходимо.
-n, --iterations=N	Перезаписать файл N раз вместо 3-х по умолчанию.
--random-source=FILE	Перезаписать файл случайными данными, взятыми из файла с именем FILE .
-s, --size=N	Перезаписать только N байт. Можно использовать суффиксы K, M, G для указания размерности: килобайт, мегабайт, гигабайт.
-u, --remove	Обрезать и удалить файл после перезаписи. По умолчанию файлы не удаляются.
-v, --verbose	Показывать ход выполнения.
-x, --exact	Не округлять размер файла до следующего целого блока.
-z, --zero	На последней итерации перезаписать файл нулями.
--version	Показать версию утилиты и выйти.
--help	Показать справку и выйти.

Пример: заполним место, занятое файлом **filename**, с последующим удалением файла.

Для этого выполним следующую команду:

```
shred -u -z filename
```

8.3 Стирание данных в свободном пространстве раздела, в котором находится директория

Утилита `sfill` позволяет стирать данные в свободном пространстве раздела, в котором находится заданная директория. Стирание производится в четыре шага:

1. Однократная перезапись числами 255 (0xFF).
2. Пятикратная перезапись случайными числами.
3. Двадцатисемикратная перезапись специальными числами.
4. И еще один раз пятикратная перезапись случайными числами.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 26: Опции утилиты `sfill` и их значения

Опция	Значение
<code>-f</code>	Выполнение более быстрым образом за счет пропуска второго и четвертого шагов перезаписи случайными числами.
<code>-i</code>	Очистка свободного пространства только индексного дескриптора, но не свободного пространства жесткого диска.
<code>-I</code>	Очистка свободного пространства только жесткого диска без затрагивания свободного пространства индексного дескриптора.
<code>-l</code>	Выполнение более быстрым образом за счет пропуска третьего и четвертого шагов или путем выполнения только одного шага перезаписи данных нулевыми значениями, если эту опцию задать дважды (например, <code>sdmem -l -l</code>).
<code>-v</code>	Работа будет сопровождаться выводом динамической строки, показывающей прогресс её выполнения.
<code>-z</code>	На четвертом шаге вместо перезаписи случайными числами выполнять перезапись нулями.

Пример: выполним очистку свободного пространства.

Для этого выполним следующую команду:

```
[root@msvsphere ~]# sfill -vz /mnt/
Using /dev/urandom for random input.
Wipe mode is secure (38 special passes)
```

(продолжение на следующей странице)

(продолжение с предыдущей страницы)

```

Wiping now ...
Creating /mnt/00000000.000 ... ***** Wiping
↳ inodes ...
Done ... Finished
[root@msvsphere ~]#

```

8.4 Стирание данных в разделах подкачки

Утилита **sswap** позволяет стирать данные в разделах подкачки. Алгоритм стирания данных абсолютно такой же, как и у утилиты **sfill**.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 27: Опции утилиты **sswap** и их значения

Опция	Значение
-f	Выполнение более быстрым образом за счет пропуска второго и четвертого шагов перезаписи случайными числами.
-l	Выполнение более быстрым образом за счет пропуска третьего и четвертого шагов или путем выполнения только одного шага перезаписи данных нулевыми значениями, если эту опцию задать дважды.
-v	Работа будет сопровождаться выводом динамической строки, показывающей прогресс её выполнения.
-z	На четвертом шаге вместо перезаписи случайными числами выполнять перезапись нулями.

8.5 Стирание данных в оперативной памяти

Утилита **sdmem** позволяет стирать данные в оперативной памяти. Алгоритм стирания данных почти такой же, как и у утилиты **sfill**, но с тем отличием, что на первом шаге однократная перезапись производится числами 0 (0x00).

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 28: Опции утилиты **sswap** и их значения

Опция	Значение
-f	Выполнение более быстрым образом за счет пропуска второго и четвертого шагов перезаписи случайными числами.
-l	Выполнение более быстрым образом за счет пропуска третьего и четвертого шагов или путем выполнения только одного шага перезаписи данных нулевыми значениями, если эту опцию задать дважды.
-v	Работа будет сопровождаться выводом динамической строки, показывающей прогресс её выполнения.

9 Контроль целостности

9.1 Введение

Средства контроля целостности предоставляют возможности контроля целостности обрабатываемых данных и используемого программного обеспечения.

9.2 Вычисление и сверка контрольной суммы файла

Утилита `md5sum` позволяет вычислять контрольные суммы файлов по алгоритму MD5 и осуществлять их сверку с другими контрольными суммами, хранящимися в файле.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 29: Опции утилиты `md5sum` и их значения

Опция	Значение
<code>-b, --binary</code>	Позволяет считывать данные из файлов в двоичном режиме.
<code>-t, --text</code>	Позволяет считывать данные из файлов в текстовом режиме.
<code>-c, --check</code>	Позволяет осуществить сверку рассчитанного значения контрольной суммы с некоторым другим значением контрольной суммы, хранящимся в файле, имя которого должно быть передано утилите в качестве аргумента.
<code>--tag</code>	Выводить рассчитанную контрольную сумму в формате BSD.
<code>--quiet</code>	При сверке контрольных сумм позволяет не выводить сообщение OK для каждого успешного случая сверки контрольных сумм.
<code>--status</code>	При сверке контрольных сумм позволяет в конце работы утилиты не выводить ничего, кроме кода сверки контрольных сумм.
<code>--strict</code>	При сверке контрольных сумм позволяет выводить ненулевое значение для неправильно отформатированных строк контрольной суммы.
<code>-w</code>	При сверке контрольных сумм позволяет выводить предупреждения о неправильно отформатированных строках контрольной суммы.
<code>--version</code>	Показать версию утилиты и выйти.

продолжение на следующей странице

Таблица 29 – продолжение с предыдущей страницы

Опция	Значение
<code>--help</code>	Показать справку и выйти.

Пример: подсчитаем контрольную сумму файла с журналом аудита.

Для этого выполним следующую команду:

```
[root@msvsphere ~]# md5sum /var/log/audit/audit.log
7125d47d351f46de50e73aaf8df016f5 /var/log/audit/audit.log
```

9.3 Проверка целостности данных

Утилита **aide** предоставляет возможности проверки целостности данных.

1. Перед началом использования необходимо установить утилиту **aide** с помощью следующей команды (необходимо по запросу системы указать пароль суперпользователя root):

```
sudo dnf install aide
```

2. Затем необходимо инициализировать базу данных для хранения состояний файлов с помощью следующей команды (необходимо по запросу системы указать пароль суперпользователя root):

```
sudo aide --init
```

В конфигурации по умолчанию команда **aide --init** проверяет только файлы и директории, заданные в конфигурационном файле **/etc/aide.conf**. Чтобы добавить необходимые файлы и директории в базу AIDE, а также изменить параметры, необходимо внести соответствующие изменения в конфигурационный файл **/etc/aide.conf**.

3. Перед началом использования инициализированной базы данных удалите **.new** из имени файла с помощью следующей команды:

```
sudo mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

4. Для изменения места расположения базы AIDE необходимо в файле **/etc/aide.conf** изменить значение параметра **DBDIR**. Для обеспечения дополнительной безопасности рекомендуется хранить базу данных, конфигурацию и файл **/usr/sbin/aide** в безопасном месте.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице.

Таблица 30: Опции утилиты **aide** и их значения

Опция	Значение
<code>-i, --init</code>	Инициализирует базу данных хранения состояний файлов. Созданная после инициализации база данных должна быть размещена в таком месте, из которого при осуществлении последующей проверки данные могли бы быть считаны. В противном случае при попытке осуществления проверки будет выведена информация об ошибке считывания.
<code>-C, --check</code>	Проверяет базу данных состояний файлов на наличие несоответствий, для чего необходимо иметь инициализированную базу данных. Эта опция выполняется по умолчанию, если не указать никакой другой опции.
<code>-u, --update</code>	Проверяет базу данных и обновляет её. При этом входные и выходные базы данных должны быть разными.
<code>-E, --compare</code>	Сравнивает две базы данных, которые должны быть определены в конфигурационном файле.
<code>-D, --config-check</code>	Осуществляет считывание данных из конфигурационного файла, уведомляя обо всех обнаруженных ошибках.
<code>-c, --config=configfile</code>	Позволяет считывать конфигурацию из указанного файла вместо используемого по умолчанию конфигурационного файла <code>./aide.conf</code> .
<code>-B, --before="configparam"</code>	Позволяет обрабатывать указанные конфигурационные параметры до считывания данных из конфигурационного файла.
<code>-A, --after="configparam"</code>	Позволяет обрабатывать указанные конфигурационные параметры после считывания данных из конфигурационного файла.
<code>-V verbosity_level, --verbose=verbosity_l</code>	Определяет детальность обработки данных. На месте параметра <code>verbosity_level</code> может находиться целочисленное значение в диапазоне от 0 до 255 . По умолчанию значение данного параметра равно 5 . Если вызвать данную опцию, но не присвоить целочисленное значение параметру <code>verbosity_level</code> , то ему автоматически будет присвоено значение, равное 20 . Этот параметр переопределяет значение, установленное в конфигурационном файле.
<code>-r reporter, --report=reporter</code>	Определяет место (URL), в которое утилита должна отправлять свои результаты работы.

продолжение на следующей странице

Таблица 30 – продолжение с предыдущей страницы

Опция	Значение
-------	----------

Пример: проверим систему, сравним `aide.db` и текущее состояние системы, без обновления `aide.db.new`.

Для этого выполним следующую команду:

```
[root@msvsphere user3]# aide --check
AIDE, version 0.15.1
### All files match AIDE database. Looks okay!
```

Если произошло изменение файлов системы, то будет получено соответствующее уведомление:

```
[root@msvsphere user3]# aide --check
AIDE 0.15.1 found differences between database and filesystem!
Start timestamp: 2023-03-28 09:12:21
Summary:
  Total number of files: 264416
  Added files:           1
  Removed files:        0
  Changed files:        3
```

Рекомендуется проводить проверку каждый день. Вы можете настроить проведение ежедневной проверки с помощью команды `cron`. Например, настроим проверку на 03:00, добавив в файл `/etc/crontab` следующую строку:

```
00 3 * * * root /usr/sbin/aide --check
```

Также, мы рекомендуем обновлять основную базу AIDE после обновлений системы и пакетов, а также изменений в конфигурационных файлах.

Для обновления основной базу AIDE выполните следующую команду:

```
sudo aide --update
```

Будет создан файл `/var/lib/aide/aide.db.new.gz`. Чтобы начать использовать вновь созданный файл для проверки целостности данных, удалите `.new` из имени файла.

10 Обеспечение надёжного функционирования

10.1 Введение

Средства обеспечения надёжного функционирования предоставляют возможности резервного копирования и восстановления данных и программного обеспечения при сбоях и отказах, а также возможности функционирования отдельных экземпляров системы на нескольких технических средствах в отказоустойчивом режиме, обеспечивающем доступность сервисов и данных при выходе из строя одного из технических средств или при исчерпании вычислительных ресурсов.

10.2 Архивация файлов и директорий

Утилита **tar** позволяет архивировать файлы и директории со всеми их поддиректориями и файлами, а затем восстанавливать их из архива, т.е. является удобным средством для создания резервных копий.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленным в таблице:

Таблица 31: Опции утилиты **tar** и их значения

Опция	Значение
-c, --create	Создать новый архив.
-r, --append	Присоединить файлы к концу архива.
--delete	Удалить файл из архива.
-t, --list	Вывести список содержимого архива.
-A, --catenate, --concatenate	Присоединить существующий архив к другому архиву.
-x, --extract, -get	Извлечь файлы из архива.
-u, --update	Добавить в архив более новые версии файлов.
-C, --directory=DIR	Сменить директорию перед выполнением операции на DIR .
--f, --file=ARCHIVE	Вывести результат в архивный файл или в устройство ARCHIVE .
-d, --diff	Осуществить проверку на наличие различий между архивом и некоторой файловой системой.
-v, --verbose	Выводить подробную информацию о процессе выполнения команды.

Пример: в примере директория **mydir** и все её поддиректории сначала сохраняются в файле **myarch.tar**:

```
tar cf myarch.tar mydir
```

а затем извлекаются из архива:

```
tar xf myarch.tar
```

А этот скрипт организует хранение четырех последних резервных копий директории `/var/www` в директории `/opt/backup/www-backup`. Первая версия будет всегда иметь номер 0, последняя — номер 3. При создании новых версий старые будут удаляться. Сами резервные копии хранятся в сжатом виде.

```
#!/bin/bash
cd /opt/backup/www-backup
rm www-dump-3.tar.gz
cp www-dump-2.tar.gz www-dump-3.tar.gz
cp www-dump-1.tar.gz www-dump-2.tar.gz
cp www-dump-0.tar.gz www-dump-1.tar.gz
tar --selinux --acls --xattrs --czf www-dump-0.tar.gz /var/www
```

10.3 Создание архивов и извлечение файлов из них

Утилита `cpio` используется для создания архивов и извлечения файлов из них, а также для копирования файлов в целях их переноса из текущей директории в другую. Поддерживает множество различных архивных форматов. При извлечении файлов из архива утилита автоматически распознает, каким типом обладает архив, с которым она взаимодействует.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 32: Опции утилиты `cpio` и их значения

Опция	Значение
<code>-o, --create</code>	Копировать файлы в архив.
<code>-A, --append</code>	Добавить файлы в архив. Может быть использована только в связке с опцией <code>-o</code> .
<code>-i, --extract</code>	Копирует файлы из архива или выводит список содержимого некоторого архива.
<code>-p, --pass-through</code>	Копирует файлы из одной файловой структуры в другую, комбинируя при этом режимы работы, использующиеся при передаче опций <code>-i</code> и <code>-o</code> , но не используя при этом архивы.

продолжение на следующей странице

Таблица 32 – продолжение с предыдущей страницы

Опция	Значение
-a, --reset-access-time	Сбрасывает времена обращения к входным файлам после их копирования, так что при использовании данной опции будет нельзя распознать, что файлы были скопированы.

Пример: в примере сначала флеш-носитель монтируется как устройство `/mnt`:

```
mount /dev/sdb4 /mnt
```

Затем создается и записывается на флеш-носитель резервная копия директории `/lib`:

```
find /lib/ | cpio -o > /mnt/2/backup.cpio
```

Для того чтобы восстановить все файлы в директорию `/lib` из созданной ранее архивной копии, необходимо выполнить следующую команду:

```
cpio -ivmd /lib/\* < /mnt/2/backup.cpio
```

10.4 Резервное копирование данных

Утилита **amanda** обладает возможностью резервного копирования данных, хранящихся на множестве компьютеров в вычислительной сети. Она реализует клиент-серверную модель и использует следующие утилиты:

- клиентская утилита **amandad**, взаимодействующая с сервером системы.

Во время своего выполнения вызывает другие утилиты:

- **selfcheck** (проверка конфигурации клиента);
- **sendsize** (оценка объема резервной копии);
- **sendbackup** (выполнение операции резервного копирования);
- **amcheck** (проверка конфигурации системы).

- серверная утилита **amdump**, инициирующая все операции резервного копирования.

Во время своего выполнения использует другие утилиты и контролирует их выполнение:

- **planner** (определение того, что надо копировать);
- **driver** (интерфейс к внешнему устройству);

- **dumper** (связывается с клиентским процессом **amandad**);
- **taper** (запись данных на внешнее устройство);
- **amreport** (подготовка сообщения о выполненном копировании).

- **административные утилиты:**

- **amcheck** (проверка готовности системы к работе);
- **amlabel** (записать метку на сменный носитель перед использованием в системе);
- **amcleanup** (очистить систему после неплановой перезагрузки сервера или после непланового завершения операции резервного копирования);
- **amflush** (переписать данные из дискового кэша на внешний носитель);
- **amadmin** (выполнение большого количества различных административных операций).

- **утилиты восстановления данных:**

- **amrestore** (восстановление данных с носителей, на которых записаны резервные копии, выполненные системой);
- **amrecover** (программа для интерактивного восстановления данных с резервных копий).

10.5 Создание дисковых RAID-массивов

Утилита **mdadm** позволяет создавать так называемые дисковые RAID-массивы с использованием технологии распределения данных по нескольким дискам с целью достижения избыточности, отказоустойчивости, сокращения задержек и/или увеличения скорости чтения и записи, а также для улучшения возможностей восстановления данных в случае отказа.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 33: Опции утилиты **mdadm** и их значения

Опция	Значение
-A, --assemble	Режим сборки ранее созданного массива и его активации.
-B, --build	Режим сборки массива без суперблоков.
-C, --creat	Режим сборки нового массива.

продолжение на следующей странице

Таблица 33 – продолжение с предыдущей страницы

Опция	Значение
-F, --follow, --monitor	Режим слежения за состоянием устройств.
-G, --grow	Режим расширения или уменьшения размера массива.
-N, --name	Устанавливает имя массива.
-n, --raid-devices	Указывает количество активных устройств в массиве.
-x, --space-device	Указывает количество запасных устройств в массиве.
-z, --size	Указывает объем пространства, используемого для каждого диска.
-l, --level	Устанавливает уровень массива.
-c, --config	Указывает файл конфигурации. По умолчанию <code>/etc/mdadm.conf</code> .
-f, --fail	Помечает перечисленные устройства как неисправные.
-S, --stop	Деактивирует массив и освобождает все ресурсы.
-V --version	Выводит информацию о версии утилиты.
-h, --help	Выводит справку об утилите.

11 Фильтрация сетевого потока

11.1 Введение

Средства фильтрации сетевого потока предоставляют возможности фильтрации входящих и исходящих сетевых потоков на основе установленного набора правил с учетом атрибутов безопасности и используемых сетевых протоколов, а также управления правилами фильтрации сетевых потоков; регистрации и учета выполнения проверок при фильтрации сетевых потоков.

11.2 Настройка файрвола (брандмауэра)

Утилита `firewall-cmd` позволяет настраивать работу файрвола (брандмауэра), осуществляющего фильтрацию сетевых потоков при помощи определения так называемых зон, иными словами, наборов правил, которые управляют трафиком на основе уровня доверия к той или иной сети.

Существуют следующие зоны:

- **drop** — самый низкий уровень доверия к сети. Весь входящий трафик сбрасывается без ответа. Поддерживаются только исходящие соединения;
- **block** — эта зона похожа на предыдущую, но при этом входящие запросы сбрасываются с сообщением `icmp-host-prohibited` или `icmp6-adm-prohibited`;
- **public** — эта зона представляет публичную сеть, которой нельзя доверять, однако поддерживает входящие соединения в индивидуальном порядке;
- **external** — зона внешних сетей. Поддерживает маскировку NAT, благодаря чему внутренняя сеть остается закрытой, но с возможностью получения доступа;
- **internal** — обратная сторона зоны **external**. Компьютерам в этой зоне можно доверять.

Доступны дополнительные сервисы:

- **dmz** — используется для компьютеров, расположенных в DMZ (зонах изолированных компьютеров, которые не будут иметь доступа к остальной части сети). Поддерживает только некоторые входящие соединения;
- **work** — зона рабочей сети. Большинству машин в сети можно доверять, доступны дополнительные сервисы;
- **home** — зона домашней сети. Окружению можно доверять, но поддерживаются только определённые пользователем входящие соединения;
- **trusted** — всем машинам в сети можно доверять.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 34: Опции утилиты `firewall-cmd` и их значения

Опция	Значение
<code>--state</code>	Вывести состояние файрвола.
<code>--reload</code>	Перезагрузить правила из постоянной конфигурации.
<code>--complete-reload</code>	Жёсткая перезагрузка правил с разрывом всех соединений.
<code>--runtime-to-permanent</code>	Перенести настройки <code>runtime</code> в постоянную конфигурацию.
<code>--permanent</code>	Использовать постоянную конфигурацию.
<code>--get-default-zone</code>	Отобразить зону, используемую по умолчанию.
<code>--set-default-zone</code>	Установить зону по умолчанию.
<code>--get-active-zones</code>	Отобразить активные зоны.
<code>--get-zones</code>	Отобразить все доступные зоны.
<code>--get-services</code>	Вывести предопределенные сервисы.
<code>--list-all-zones</code>	Вывести конфигурацию всех зон.
<code>--new-zone</code>	Создать новую зону.
<code>--delete-zone</code>	Удалить зону.
<code>--list-all</code>	Вывести всё, что добавлено, из выбранной зоны.
<code>--list-services</code>	Вывести все сервисы, добавленные к зоне.
<code>--add-service</code>	Добавить сервис к зоне.
<code>--remove-service</code>	Удалить сервис из зон.
<code>--list-ports</code>	Отобразить порты, добавленные к зоне.
<code>--add-port</code>	Добавить порт к зоне.
<code>--remove-port</code>	Удалить порт из зоны.
<code>--query-port</code>	Показать, добавлен ли порт к зоне.
<code>--list-protocols</code>	Вывести протоколы, добавленные к зоне.
<code>--add-protocol</code>	Добавить протокол к зоне.
<code>--remove-protocol</code>	Удалить протокол из зоны.
<code>--list-source-ports</code>	Вывести порты источника, добавленные к зоне.
<code>--add-source-port</code>	Добавить порт-источник к зоне.
<code>--remove-source-port</code>	Удалить порт-источник из зоны.
<code>--list-icmp-blocks</code>	Вывести список блокировок icmp.
<code>--add-icmp-block</code>	Добавить блокировку icmp.
<code>--remove-icmp-block</code>	Удалить блокировку icmp.
<code>--add-forward-port</code>	Добавить порт для перенаправления в NAT.
<code>--remove-forward-port</code>	Удалить порт для перенаправления в NAT.
<code>--add-masquerade</code>	Включить NAT.
<code>--remove-masquerade</code>	Удалить NAT.

Пример: настройка правила блокировки адреса получателя может выглядеть следующим образом:

```
[root@msvsphere ~]# firewall-cmd --permanent --direct --add-rule ipv4
→filter OUTPUT 0 -d 192.168.10.20 -j DROP
success
```

Пример: настройка правила отбрасывания всех входящих соединений по протоколу IPv4 может выглядеть следующим образом:

```
[root@msvsphere ~]# firewall-cmd --permanent --direct --add-rule ipv4
→filter INPUT 0 -j DROP
success
```

Пример: настройка правила отбрасывания всех исходящих пакетов UDP может выглядеть следующим образом:

```
[root@msvsphere ~]# firewall-cmd --permanent --direct --add-rule ipv4
→filter OUTPUT 0 -p upd -j DROP
success
```

11.3 Конфигурационный файл /etc/firewalld/firewalld.conf

Конфигурационный файл /etc/firewalld/firewalld.conf содержит основные параметры конфигурации для файрвола `firewalld`.

- **DefaultZone** — устанавливает зону по умолчанию для соединений или интерфейсов;
- **MinimalMark** — с этой опцией блок меток может быть зарезервирован для частного использования. Используются только отметки над этим значением. Значение по умолчанию равно **100**;
- **CleanupOnExit** — если `firewalld` останавливается, он очищает все правила. Если для этого параметра установлено значение **no** или **false**, текущие правила останутся нетронутыми. Значением по умолчанию является **yes** или **true**;
- **Lockdown** — если эта опция включена, изменения `firewalld` с интерфейсом D-Bus будут ограничены приложениями, которые перечислены в белом списке блокировки. Значением по умолчанию является **no** или **false**;
- **IPv6_rpfilter** — если эта опция включена, выполняется проверка фильтра обратного пути для пакета для IPv6. Если ответ на пакет будет отправлен через тот же интерфейс, на который поступил пакет, пакет совпадет и будет принят.

В противном случае он будет отброшен. Для IPv4 `rp_filter` управляется с помощью `sysctl`;

- **IndividualCalls** — если этот параметр отключен, используются комбинированные вызовы `restore`, а не отдельные вызовы, чтобы применить изменения к файрволу. Использование отдельных вызовов увеличивает время, необходимое для применения изменений;
- **LogDenied** — добавление правил ведения журнала непосредственно перед отклонением и удалением правил в цепочках **INPUT**, **FORWARD** и **OUTPUT** для правил по умолчанию, а также окончательных правил отклонения и отбрасывания в зонах для настроенного типа пакета канального уровня. По умолчанию установлено **off** — отключение ведения журнала.
- **AutomaticHelpers** — для безопасного использования протокола IPv4 **iptables** и помощников по отслеживанию соединений этот параметр рекомендуется отключить. Возможные значения: **yes**, **no**, **system**. По умолчанию установлено **system**;
- **FirewallBackend** — выбирает реализацию брандмауэра. Возможные значения: **nftables** (по умолчанию) или **iptables**. Это относится ко всем примитивам **firewalld**. Единственным исключением являются прямые и сквозные правила, которые всегда используют традиционные **iptables**, **ip6tables** и **ebtables**.

12 Мониторинг функционирования

12.1 Введение

Средства мониторинга функционирования предоставляют возможности слежения и сбора информации о выполнении пользовательских процессов и состоянии сетевого трафика.

12.2 Анализ системных журналов

Утилита `logwatch` позволяет проводить анализ системных журналов по различным критериям с возможностью составления отчетов.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 35: Опции утилиты `logwatch` и их значения

Опция	Значение
<code>--detail level</code>	Уровень детализации отчета. Может быть положительным целым числом или <code>high</code> , <code>med</code> , <code>low</code> , которые соответствуют целым числам 10, 5 и 0 соответственно.
<code>--debug level</code>	Уровень отладки. Может варьироваться от 0 до 100.
<code>--logfile</code> <code>log-file-group</code>	Обрабатывать только набор указанных файлов журналов.
<code>--service</code> <code>service-name</code>	Обрабатывать только указанную службу.
<code>--print</code>	Вывести результаты на экран.
<code>--mailto address</code>	Отправить результаты по указанному адресу электронной почты.
<code>--save file-name</code>	Сохранить вывод в указанный файл вместо отображения на экране или отправки по электронной почте.
<code>--range range</code>	Диапазон дат для обработки.
<code>--archives</code>	Искать в архивных журналах.
<code>--logdir directory</code>	Обрабатывать файлы журналов из указанного каталога, а не из каталога по умолчанию.
<code>--hostname hostname</code>	Обрабатывать файлы журналов только указанного хоста.

12.3 Получение информации о выполняемых процессах

Утилита **top** предназначена для получения информации о выполняемых процессах.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 36: Опции утилиты **top** и их значения

Опция	Значение
-u	Отображать только процессы с заданным идентификатором или именем пользователя.
-S	Отображать системные процессы.
-n	Изменить число отображаемых процессов на заданное число.
-i	Работа в интерактивном режиме. Задается по умолчанию.
-I	Не отображать бездействующие процессы. По умолчанию отображаются как активные, так и бездействующие процессы.
-c	Переключение отображения командных строк на отображение имён программ и наоборот.
-s	Задаёт временной интервал задержки между обновлениями экрана. По умолчанию 5 секунд.
-b	Работа в пакетном режиме. Может использоваться для отправки результатов в другие программы или в файл.
-o	Задаёт имя поля, по которому будет осуществляться сортировка. Используется в основном для пакетного режима.
-w	Задаёт форматирование вывода по ширине. Количество строк считается неограниченным.
-v	Показать версию утилиты и выйти.
-h	Показать справку и выйти.

12.4 Получение информации о состоянии текущих процессов

Утилита **ps** используется для получения информации о состоянии текущих процессов.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 37: Опции утилиты **ps** и их значения

Опция	Значение
-u	Выводить информацию только о процессах с заданными списком эффективными идентификационными номерами или идентификаторами пользователей.
-Y	Выводить информацию только о процессах с заданными списком реальными идентификационными номерами или идентификаторами пользователей.
-g	Выводить информацию только о процессах с заданными списком идентификационными номерами групп.
-G	Выводить информацию только о процессах с заданными списком реальными идентификационными номерами групп.
-a	Выводить информацию о состоянии наиболее часто запрашиваемых процессов.
-e	Выводить информацию для всех процессов.
-d	Выводить информацию о всех процессах, кроме лидеров сеансов.
-p	Выводить информацию только для запущенных процессов.
-G	Выводить информацию о процессах, чьи реальные номера групп указаны в заданном списке.
-o	Выводить информацию в заданном формате.

12.5 Мониторинг и анализ сетевого трафика

Утилита **tcpdump** предназначена для мониторинга и анализа сетевого трафика.

Состоит из двух частей: захват пакетов с копированием их в так называемый буфер и отображение захваченных пакетов из буфера.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 38: Опции утилиты `tcpdump` и их значения

Опция	Значение
-i	Задаёт интерфейс, с которого необходимо анализировать трафик.
-y	Устанавливает тип канала передачи данных для использования во время захвата пакетов.
-e	Включает вывод данных канального уровня.
-v	Вывод дополнительной информации.
-w	Задаёт имя файла, в котором будет сохраняться собранная информация.
-r	Захватывать только трафик, предназначенный данному узлу.
-q	Переводит работу в «бесшумный режим», в котором пакет анализируется на транспортном уровне, а не на сетевом.
-t	Отключает вывод меток времени.
-A	Вывод пакетов в формате ASCII без заголовков канального уровня.
-B	Установить размер буфера захвата.
-D	Вывести список доступных сетевых интерфейсов, на которых может осуществляться захват пакетов.

12.6 Получение информации о сеансах пользователей

Утилита `ac` предназначена для получения информации о сеансах пользователей.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 39: Опции утилиты `ac` и их значения

Опция	Значение
-p	Выводить итоговое время сеансов каждого пользователя.
-d	Кроме общих итогов, выводить итоги за каждый день.
-a	При выводе ежедневных итогов не пропускать дни, когда входов в систему не было.
-y	Выводить год при отображении даты.
-z	Если итоговое значение равно нулю, то выводить его. По умолчанию не выводится.

продолжение на следующей странице

Таблица 39 – продолжение с предыдущей страницы

Опция	Значение
-v	Вывести номер версии.
-h	Вывести краткую справку.

12.7 Получение информации о последних выполненных командах

Утилита `lastcomm` позволяет получить информацию о последних выполненных командах.

Режимы работы утилиты и выполняемые функции задаются набором опций, перечисленных в таблице:

Таблица 40: Опции утилиты `lastcomm` и их значения

Опция	Значение
-E	Выводить время начала процесса выполнения команды.
-S	Выводить время завершения процесса выполнения команды.
-c	Выводить количество использованного процессорного времени.
-e	Выводить количество использованного прошедшего времени.
-s	Выводить количество использованного системного времени.
-u	Выводить количество использованного пользовательского времени.
-f	Использовать заданный файл в качестве источника учетных данных. Он может быть либо стандартным, либо расширенным файлом учёта процесса.
-x	Использовать текущий расширенный файл учёта процесса.

