

УТВЕРЖДЕН

ЛКНВ.11102-01 99 01-ЛУ

ОПЕРАЦИОННАЯ СИСТЕМА АЛЬТ 8 СП  
(ОС АЛЬТ 8 СП)

Руководство по комплексу средств защиты  
ЛКНВ.11102-01 99 01

Листов 229

Инд. № подл.	Подп. и дата	Взам. инв. №	Инд. № дубл.	Подп. и дата

2024

Литера О

## АННОТАЦИЯ

Данный документ описывает комплекс средств защиты (КСЗ) программного изделия (ПИ) «Операционная система Альт 8 СП» ЛКНВ.11102-01, сокращенное наименование – ОС Альт 8 СП, **релиз 10** для процессоров VLIW архитектуры **Эльбрус**.

Далее в документе будет использоваться альтернативное наименование ПИ: ОС Альт СП.

Версия: 1.0.

Руководство по КСЗ состоит из пяти основных частей, в которых раскрываются основные вопросы применения, структуры и функционирования комплекса средств защиты ОС Альт СП.

В первом разделе приводится краткий обзор возможностей ОС Альт СП по защите данных.

Во втором разделе приводится описание модели защиты. Описываются принципы защиты, используемые в ОС Альт СП.

В третьем разделе приводятся типовые функции и задачи безопасности администратора, а также описываются конкретные инструменты для выполнения этих функций и задач.

В четвертом разделе приводятся указания по эксплуатации и проверке целостности комплекса средств защиты.

Пятый раздел содержит информацию по обновлению ОС Альт СП.

## СОДЕРЖАНИЕ

1. Общие сведения.....	7
2. Структура КСЗ.....	8
2.1. Логическая структура .....	8
2.2. Алгоритм работы КСЗ .....	9
2.3. Описание модели защиты.....	11
2.3.1. Субъекты доступа .....	11
2.3.2. Объекты доступа .....	11
2.3.3. Основные положения модели защиты .....	12
2.3.4. Модель идентификации и аутентификации .....	13
2.3.5. Модель дискреционного разграничения доступа .....	15
2.3.6. Модель управления памятью .....	25
2.3.7. Модель контроля целостности.....	27
2.3.8. Модель защиты ввода и вывода на отчуждаемый физический носитель.....	33
2.3.9. Модель обеспечения доверенной загрузки средств вычислительной техники .....	35
2.3.10. Модель сопоставления пользователя с устройством .....	35
2.3.11. Модель системы протоколирования событий.....	36
2.3.12. Средства сбора сетевой статистики и фильтрации сетевых пакетов.....	39
2.3.13. Средства контроля запуска компонентов программного обеспечения.....	39
3. Управление КСЗ .....	40
3.1. Использование API библиотеки polkit.....	40
3.1.1. Файлы действий .....	40
3.1.2. Файлы правил .....	42
3.1.3. Журналирование действий polkit .....	44

3.2. Средства управления учетными записями пользователей.....	45
3.2.1. Общая информация.....	46
3.2.2. Обязка passwd.....	46
3.2.3. Добавление нового пользователя.....	47
3.2.4. Добавление/редактирование пользователей в графической оболочке и в веб-интерфейсе.....	52
3.2.5. Настройка парольных ограничений.....	54
3.2.6. Настройка неповторяемости пароля.....	58
3.2.7. Модификация уже имеющихся пользовательских записей.....	60
3.2.8. Удаление пользователей.....	62
3.2.9. Оповещение пользователя после успешного входа в ОС о его предыдущем входе.....	62
3.2.10. Разрешение (запрет) действий пользователей, разрешенных до идентификации и аутентификации.....	63
3.3. Средства управления дискреционными ПРД.....	64
3.3.1. Команда chmod.....	64
3.3.2. Команда chown.....	66
3.3.3. Команда chgrp.....	67
3.3.4. Команда umask.....	68
3.3.5. Команда chattr.....	68
3.3.6. Команда lsattr.....	71
3.3.7. Команда mksock.....	71
3.3.8. Команда mkfifo.....	71
3.3.9. Команда getfacl.....	72
3.3.10. Команда setfacl.....	73
3.3.11. Элементы ACL.....	74
3.3.12. Автоматически созданные права доступа.....	75
3.4. Средства управления и очистки памяти.....	76
3.4.1. Управление механизмом очистки оперативной памяти.....	76

3.4.2. Очистка дисковой памяти .....	78
3.5. Средства контроля ввода-вывода .....	79
3.5.1. Средства взаимодействия с устройствами ввода-вывода .....	79
3.5.2. Средства контроля использования интерфейсов ввода (вывода) информации на машинные носители данных .....	80
3.5.3. Ограничения при помощи правил API библиотеки polkit .....	80
3.5.4. Контроль при помощи правил polkit .....	82
3.5.5. Ограничения при помощи правил Udev .....	83
3.5.6. Управление монтированием блочных устройств .....	87
3.5.7. Настройка ограничений на использование внешних носителей.....	87
3.5.8. Настройка ограничения доступа к файловой системе USB-устройства.....	110
3.5.9. Контроль доступа к портам.....	117
3.6. Средства контроля целостности .....	120
3.6.1. Утилита integalert .....	120
3.6.2. Активация средств контроля целостности .....	124
3.6.3. Подсистема IMA/EVM.....	127
3.6.4. Проверка хешей установленных пакетов rpm.....	135
3.6.5. Интеграция Icinga.....	139
3.7. Средства резервного копирования .....	140
3.7.1. Резервное копирование при помощи утилиты rsync .....	140
3.7.2. Пример настройки системы резервного копирования данных .....	143
3.7.3. Восстановление программного обеспечения при возникновении нештатных ситуаций.....	146
3.8. Средства управления протоколированием событий.....	146
3.8.1. Управление журналированием .....	146
3.8.2. Управление аудитом .....	158
3.8.3. Использование аудита .....	191
3.9. Средства контроля запуска компонентов программного обеспечения .....	210

3.9.1. Принцип работы control++ .....	210
3.9.2. Настройка .....	211
3.10. Надежное хранение данных .....	218
4. Указания по эксплуатации КСЗ .....	219
4.1. Порядок старта .....	219
4.2. Проверка правильности старта КСЗ.....	220
4.3. Периодическая проверка целостности КСЗ .....	220
4.3.1. Порядок проверки .....	220
4.3.2. Контроль целостности файлов паролей и списка групп .....	221
5. Обновление ОС Альт СП.....	225
Перечень сокращений .....	227

## 1. ОБЩИЕ СВЕДЕНИЯ

ОС Альт СП предназначена для группового и корпоративного использования в качестве средства автоматизации информационных, конструкторских и производственных процессов предприятий (организаций, учреждений).

ОС Альт СП поддерживает клиент-серверную архитектуру и может обслуживать процессы как в пределах одной электронной вычислительной машины (ПЭВМ), так и процессы на других ПЭВМ через каналы передачи данных или сетевые соединения.

## 2. СТРУКТУРА КСЗ

### 2.1. Логическая структура

КСЗ представляет собой специальные пакеты программ операционной среды, входящие в состав ядра ОС и системных библиотек.

Основные логические компоненты КСЗ (рис. 1):

- средства управления доступом;
- средства регистрации и учета;
- средства контроля целостности;
- средства управления памятью.

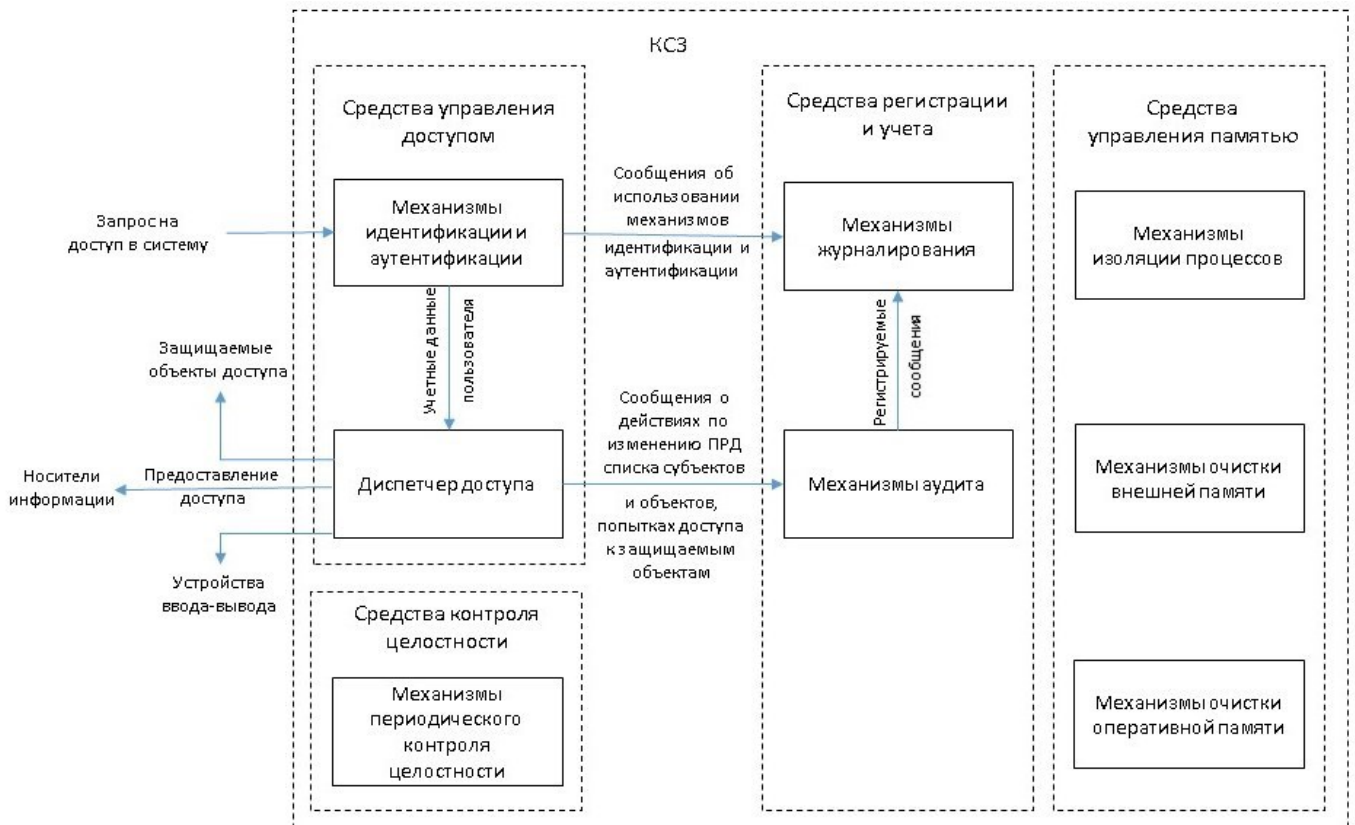


Рис. 1 – Логические компоненты КСЗ



КСЗ обеспечивает выполнение следующих функций безопасности:

- идентификация и аутентификация;
- управление доступом;
- регистрация событий безопасности;
- ограничение программной среды;
- изоляция процессов;
- защита памяти;
- контроль целостности;
- обеспечение надежного функционирования;
- фильтрация сетевого потока.

## 2.2. Алгоритм работы КСЗ

Функционирование КСЗ осуществляется в соответствии со следующим алгоритмом:

- 1) выполняется запуск ПЭВМ, по завершении запуска формируется соответствующее сообщение и передается механизмам аудита/журналирования;
- 2) после запуска ПЭВМ пользователем осуществляется запрос на доступ в ОС Альт СП, пользователь указывает свой уникальный идентификатор (далее – логин) и пароль, начинается процедура идентификации;
- 3) механизмы идентификации выполняют поиск логина учетной записи в виртуальных базах данных (далее – БД) пользователей, и, если логин отсутствует в виртуальных БД, то сеанс соединения с системой прекращается;
- 4) если механизмы идентификации находят логин пользователя в виртуальных БД, то идентификация считается пройденной успешно и начинается процесс аутентификации;
- 5) если указанный пароль не соответствует логину, то сеанс соединения с системой прекращается;

- 6) если указанный пользователем пароль соответствует его логину, то аутентификация считается пройденной успешно и пользователю предоставляется доступ в систему;
- 7) все сведения об использовании механизмов идентификации и аутентификации передаются механизмам аудита/журналирования;
- 8) доступ пользователя к объектам санкционируется дискреционным механизмом разграничения доступа;
- 9) все запросы пользователя на доступ к объектам фиксируются средствами регистрации событий;
- 10) каждый запрос пользователя к объекту доступа выполняется некоторым процессом;
- 11) средства управления памятью (находятся в ядре ОС) выделяют необходимый минимум свободной оперативной памяти для этого процесса;
- 12) при этом диспетчер доступа разграничивает доступ к памяти этого процесса в соответствии с дискреционными ПРД;
- 13) по завершении работы процесса память, занимаемая им, очищается и перераспределяется средствами управления памятью;
- 14) все операции по изменению ПРД, списка пользователей, списка объектов файловой системы, запросы на доступ к защищаемым объектам файловой системы, а также все попытки идентификации и аутентификации в системе регистрируются механизмами средств регистрации событий (журналирование, аудит);
- 15) программный комплекс контроля целостности osec (далее – ПК osec) обеспечивает контроль состояния всех ключевых элементов системы (ядра, механизмов КСЗ, сведений о субъектах и объектах системы) – фиксирует состояние системы в момент отсутствия несанкционированного доступа (НСД) в ОС и сохраняет сведения в файлы БД.

### 2.3. Описание модели защиты

Показатели защищенности от несанкционированного доступа к информации и общая модель защиты строятся на базе расширенной дискреционной модели защиты, модели идентификации и аутентификации, модели контроля целостности, модели протоколирования событий, а также на модели управления памятью.

#### 2.3.1. Субъекты доступа

Субъектами доступа являются пользователи ОС Альт СП: пользователь, администратор (пользователь с идентификатором root).

Функции, задачи пользователя, а также классы команд, используемых пользователем, и их описание приведены в документе «Руководство пользователя. ЛКНВ.11102-01 91 01».

Общие функции, задачи администратора, а также классы команд, используемых администратором, и их описание приведены в документе «Руководство администратора. ЛКНВ.11102-01 90 01».

#### 2.3.2. Объекты доступа

Объектами доступа являются:

- файлы (предназначены для хранения символьных и двоичных данных);
- каталоги (предназначены для организации доступа к файлам);
- символические и жесткие ссылки (предназначены для предоставления доступа к файлам, расположенным на любых носителях);
- файлы блочных и символьных устройств (предоставление интерфейса для взаимодействия с аппаратным обеспечением компьютера);
- каналы и сокеты (организация межпроцессного взаимодействия в операционной системе);
- механизмы IPC (разделяемая память, семафоры, очереди сообщений).

### 2.3.3. Основные положения модели защиты

В основе модели защиты лежит ряд положений о наличии у пользователей (групп пользователей) уникальных идентификаторов, о действительных субъектах и уровне реализации ПРД.

#### 2.3.3.1. Уникальный численный идентификатор пользователей

С каждым пользователем системы связан уникальный численный идентификатор – идентификатор пользователя (UID), который является ключом к соответствующей записи в БД пользователей, содержащей информацию о пользователях, включая их реальные и системные имена.

БД пользователей поддерживается и управляется администратором. UID – ярлык субъекта (номинальный субъект), которым система пользуется для определения прав доступа. Соответствие входных имен уникальным идентификаторам (UID) пользователей устанавливается в пространстве имен, специфицированном в файле `/etc/nsswitch.conf`. В ОС Альт СП по умолчанию используется файловое пространство имен: публичная учетная информация пользователей хранится в файле `/etc/passwd`, приватная – в файлах вида `/etc/tcb/<входное имя>/shadow`.

Вместо файлового пространства имен для организации БД пользователей может использоваться одна из реализаций протокола LDAP – пакет OpenLDAP, входящий в состав ОС Альт СП.

#### 2.3.3.2. Уникальный численный идентификатор группы пользователей

Каждый пользователь входит в одну или более групп. Группа – это список пользователей системы, имеющий собственный идентификатор (GID). Поскольку группа объединяет несколько пользователей системы, в терминах политики безопасности она соответствует понятию «множественный субъект». GID – ярлык множественного субъекта, которых у номинального субъекта может быть более одного. Таким образом, одному UID соответствует список GID.

Вхождение пользователя в «первичную» группу отражено в файле `/etc/passwd`, во все остальные – в файле `/etc/group`.

### 2.3.3.3. Действительный субъект

Роль действительного (работающего с объектами) субъекта играет процесс. Каждый запущенный процесс в системе снабжается уникальным идентификатором (PID, Process ID), отображается в таблице процессов и сопоставляется каталогу вида /proc/<PID>, содержащему различную информацию о работе процесса.

Каждый процесс снабжен единственным UID – идентификатором запустившего процесс номинального субъекта (пользователя). Процесс, порожденный некоторым процессом пользователя, наследует его UID.

Таким образом, все процессы, запускаемые по желанию пользователя, будут иметь его идентификатор. Все процессы, принадлежащие пользователю, образуют сеанс пользователя. Первый процесс сеанса пользователя порождается после прохождения процедур идентификации и аутентификации. При обращении процесса к объекту доступ предоставляется по результатам процедуры авторизации, то есть обработки запроса на основе дискреционных ПРД.

### 2.3.3.4. Уровень реализации механизма ПРД

Механизм ПРД реализован в ядре ОС, что обеспечивает его правильное функционирование при использовании любых компонентов, предоставляемых ОС.

### 2.3.4. Модель идентификации и аутентификации

Модель идентификации и аутентификации пользователя функционирует в соответствии со следующим алгоритмом:

- пользователь посылает запрос на доступ к системе;
- автоматически системой вызывается программа login (используется для запуска нового сеанса в системе), которая выводит приглашение login на терминал пользователя;
- пользователь предъявляет свое входное имя (далее – логин) и пароль;

- модули NSS перехватывают логин пользователя и осуществляют его поиск в файлах виртуальной БД пользователей системы (для конфигурации источников виртуальной БД пользователей используется файл `/etc/nsswitch.conf`): в файлах `/etc/passwd`, `/etc/shadow`, `/etc/group`;
- если модули NSS находят логин пользователя в файлах `/etc/passwd`, `/etc/shadow`, `/etc/group`, то передают их модулям PAM, и затем начинается процесс аутентификации;
- модули PAM сравнивают логин и пароль, предъявленные пользователем, со значениями, хранящимися в файлах `/etc/passwd`, `/etc/shadow`, `/etc/group`;
- если введенные имя и пароль субъекта соответствуют хранящимся значениям, КСЗ предоставляет доступ субъекту в ОС Альт СП, информация о результате попытки доступа фиксируется средствами регистрации событий;
- если введенные имя и пароль субъекта не идентичны значениям, хранящимся в базе данных, КСЗ отклоняет запрос доступа субъекта в ОС Альт СП (для выполнения повторной попытки аутентификации субъект должен инициировать новый запрос доступа), информация о результате попытки доступа сохраняется в системном журнале;
- в конфигурации файла `/etc/nsswitch.conf` можно указать несколько источников файлов БД пользователей, например, в качестве источника указать дерево каталогов LDAP;
- если логин и пароль пользователя отсутствуют в файлах `/etc/passwd`, `/etc/shadow`, `/etc/group`, модули NSS осуществляют поиск в дереве каталогов LDAP, и затем передают их модулям PAM;
- модули PAM сравнивают логин и пароль, предъявленные пользователем, со значениями, хранящимися в дереве каталогов LDAP;
- если введенные имя и пароль субъекта соответствуют хранящимся значениям, КСЗ предоставляет доступ субъекту в ОС Альт СП, информация о результате попытки доступа фиксируется средствами регистрации событий;

- в случае необходимости реализации сетевой аутентификации пользователей в системе предусмотрена возможность аутентификации с помощью Kerberos с хранением информации о пользователях в дереве каталогов LDAP (Kerberos может использоваться и для осуществления локальной аутентификации пользователей);
- вся информация о результатах попыток доступа регистрируется средствами регистрации событий.

Схема алгоритма приведена на рис. 2.

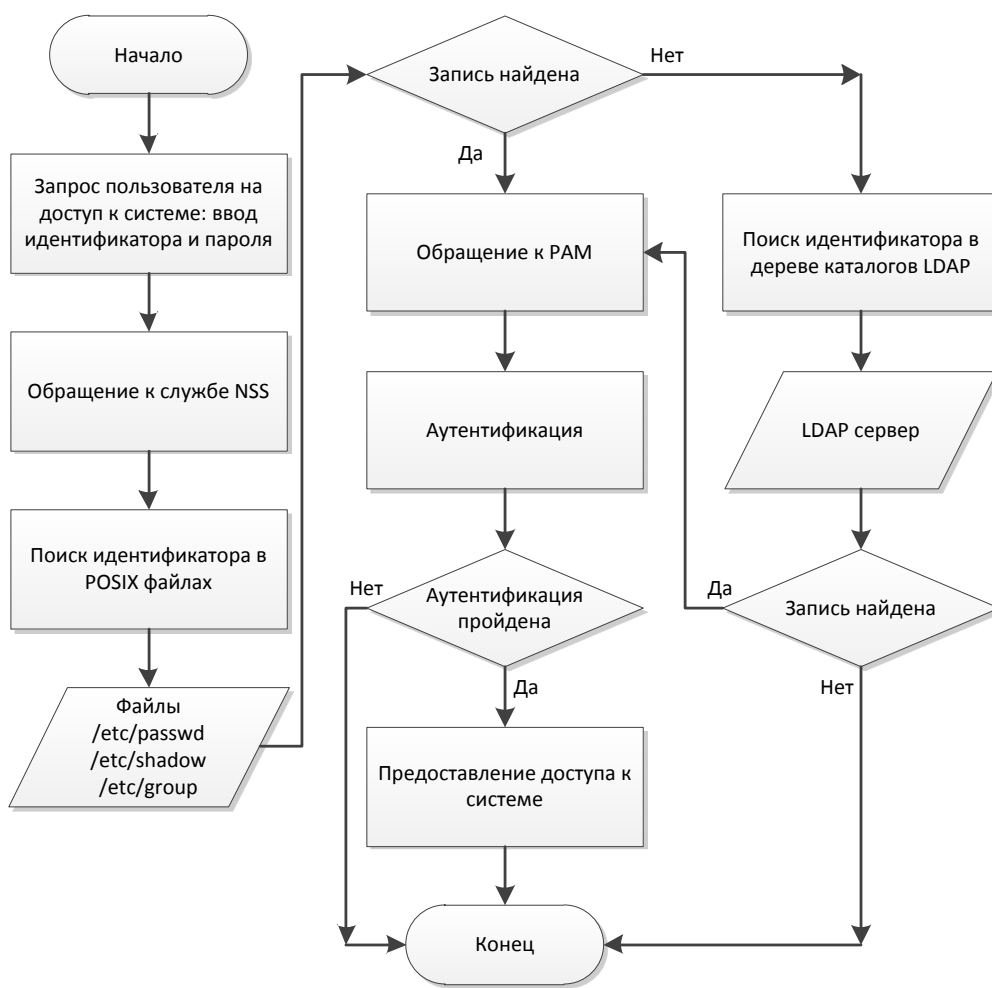


Рис. 2 – Алгоритм идентификации и аутентификации

### 2.3.5. Модель дискреционного разграничения доступа

#### 2.3.5.1. Общие сведения

Механизм дискреционного разграничения доступа реализован в ядре ОС Альт СП и заключается в том, что на защищаемые именованные объекты

устанавливаются (автоматически при их создании) базовые ПРД в виде идентификаторов номинальных субъектов (UID и GID), которые вправе распоряжаться доступом к данному объекту и правами доступа к объекту.

Различают три вида доступа: чтение, запись и исполнение. При обращении процесса к объекту (с запросом доступа определенного вида) система проверяет совпадение идентификаторов владельцев процесса и владельцев файла в определенном порядке и, в зависимости от результата, применяет ту или иную группу прав.

В случае, если текущими правилами разрешено (санкционировано), то права доступа файлового объекта могут быть изменены.

Кроме общей схемы разграничения доступа, ОС Альт СП поддерживает также ACL, с помощью которых можно для каждого объекта задавать права всех субъектов на доступ к нему.

Механизм дискреционного разграничения доступа обеспечивает возможность санкционированного изменения списка пользователей и списка защищаемых файловых объектов.

Механизм дискреционного разграничения доступа затрагивает:

- механизмы IPC;
- файловые системы Ext2/Ext3/Ext4;
- сетевая файловая система NFS;
- виртуальная файловая система VFS;
- файловые системы JFS, XFS, ReiserFS.

К защищаемым объектам доступа относятся:

- обычные файлы;
- каталоги;
- жесткие ссылки (права доступа на жесткие ссылки будут в точности дублировать права доступа файлов, на которые они ссылаются);
- блочные и символьные устройства;
- доменные сокеты (IPC-сокеты);



- каналы межпроцессного взаимодействия (именованные);
- механизмы IPC (разделяемая память, очереди сообщений и др.).

**Примечание.** Механизм дискреционного разграничения доступа ОС Альт СП не распространяется на сетевые сокет (INET-сокеты), ссылки, устройства и каналы.

### 2.3.5.2. Модель защиты файлов

#### 2.3.5.2.1. Общая схема разграничения доступа

В ОС Альт СП различают следующие типы файлов:

- обычные файлы (предназначены для хранения символьных и двоичных данных);
- каталоги (предназначены для организации доступа к файлам);
- символические и жесткие ссылки (предназначены для предоставления доступа к файлам, расположенным на любых носителях);
- файлы блочных и символьных устройств (предоставление интерфейса для взаимодействия с аппаратным обеспечением компьютера);
- каналы и сокеты (организация межпроцессного взаимодействия в операционной системе).

Определяются три вида доступа:

- чтение (`read, r`);
- запись (`write, w`);
- исполнение (`execution, x`).

Описанные далее механизмы действуют для ссылок, устройств, каналов и сокетов.

Чтение для файла означает право получать содержимое по индексному дескриптору. Для каталога – означает право получать список имен объектов, содержащихся в нем. При этом, если доступ на чтение к каталогу запрещен, процесс не сможет получить список имен, однако доступ непосредственно к файлу, находящемуся в каталоге, регулируется правом использования (исполнения) каталога, а не правом чтения.

Запись для файла означает право модифицировать содержимое по индексному дескриптору. Для каталога – означает право модифицировать список файлов. Без права на использование (исполнение) каталога право на запись практически неприменимо.

Исполнение для файла означает право запускать его в качестве программы. Различают бинарные исполняемые файлы, которые непосредственно загружаются в память в виде процесса (возможно, посредством динамической компоновки с разделяемыми библиотеками) и сценарии, для выполнения которых запускается процесс из другого файла, а текущий файл отдается ему в качестве параметра командной строки (следовательно, для работы запускаемого сценария требуется также доступ на чтение).

Для каталога доступ на использование (исполнение) означает право преобразовывать имена объектов, находящихся в каталоге, в индексные дескрипторы. Список имен файлов в каталоге, доступном процессу на чтение, но не на использование, будет виден, но сами файлы останутся недоступны.

Для работы с блочными и символьными устройствами в ОС при монтировании создаются специальные файлы, обеспечивающие произвольный или последовательный доступ соответственно типу устройства, которому они назначаются. Права доступа для учетных записей пользователя и вызываемых процессов назначаются на соответствующий созданный файл.

Права доступа к локальным сокетам назначаются на специальный файл сокета по заданному пути, через который к сокету будут обращаться любые локальные процессы путем чтения/записи из него. При использовании сетевого сокета, создается абстрактный объект, привязанный к слушающему порту операционной системы и сетевому интерфейсу, затем ему присваивается INET-адрес, который имеет адрес интерфейса и слушающего порта, и далее обращение будет происходить к абстрактному объекту согласно назначенным правам.

Права доступа именованного канала аналогичны правам доступа к файлу. Обращение к именованному каналу осуществляется так же, как и к обычному

файлу. В связи с этим, для работы с именованными каналами процессам необходимо предоставлять права доступа для чтения (записи) из (в) канал. При создании канала необходимо учитывать, что каналы создаются с правами доступа «0666», модифицированными маской прав доступа `umask` вызывающего процесса. Также утилита создания канала требует право на запись в родительский каталог.

Права доступа к символьным ссылкам всегда выглядят как `rwXrwxrwx`, поскольку при использовании ссылки драйвер файловой системы пересчитывает реальный путь к файлу и применяет права доступа, определенные для реального пути уже без учета самой символьной ссылки.

При вычислении прав доступа принимается во внимание уровень доступа процесса к файлу, который вычисляется следующим образом:

- если UID файла и актуальный UID процесса совпадают, процесс считается владельцем файла;
- в противном случае, если GID файла совпадает с актуальным GID процесса или входит в список групп, процесс считается членом группы;
- если оба условия не выполнены, процесс считается чужим по отношению к файлу.

Права доступа включают список из девяти атрибутов (битов) файла, записываемых в форме `rwXrwxrwx`: по три вида доступа (чтение – `read`, запись – `write`, исполнение – `execute`) для трех групп – пользователя-владельца (`u`), группы-владельца (`g`) и всех остальных (`o`) соответственно. Каждый пункт в этом списке может быть либо разрешен, либо запрещен (равен «1» или «0»). Также если некоторый доступ запрещен на некотором уровне, вместо символа пишется знак «-». Атрибуты неотторжимы от файла, так как хранятся в его метаданных (индексном дескрипторе), и не зависят от количества имен (ссылок на файл) и их расположении в дереве каталогов.

Описанные выше права выставляются с помощью функции `umask` (`user file creation mode mask`). `umask` одинаковым образом работает для всех объектов: каждый установленный бит `umask` запрещает выставление соответствующего бита

прав. Исключением из этого запрета является бит исполняемости, который для обычных файлов зависит от создающей программы (трансляторы ставят бит исполняемости на создаваемые файлы, другие программы – нет), соответственно, исключением являются сокеты и каналы межпроцессного взаимодействия и монтируемые аппаратные устройства. В случае каталогов `umask` следует общему правилу.

Права доступа файлового объекта могут быть изменены, если это разрешено текущими правилами (санкционировано). Модифицировать права доступа может только процесс-владелец (пользователь-владелец) файла, либо суперпользовательский (запущенный от имени пользователя `root`) процесс (UID процесса = 0).

Суперпользовательский процесс имеет право подменять свои актуальные UID и (или) GID на произвольные идентификаторы системы и восстанавливать исходные идентификаторы при необходимости. Процесс, порожденный некоторым процессом с измененными идентификаторами, наследует не актуальные, а системные (не измененные) идентификаторы.

Процесс, запущенный с идентификатором обычного пользователя, не имеет права изменять его, а также идентификатор группы и список группы. Для временного повышения или изменения прав предусмотрен механизм подмены актуального UID и (или) GID при запуске программы из файла. Владелец файла или администратор может установить атрибут SETUID (SUID) исполняемому файлу. В случае, когда этот файл загружается в память в качестве программы (это исключает запускаемые сценарии), актуальный UID дочернего процесса окажется равным UID файла (UID пользователя-владельца файла), а не UID родительского процесса, как в стандартном случае. Аналогично при запуске из файла с атрибутом SETGID (SGID) дочерний процесс унаследует GID не у родительского процесса, а у файла. Для того чтобы некоторый пользователь мог осуществить изменение объема прав посредством запуска такого файла, достаточно, чтобы у него был доступ на

использование (запуск) этого файла и файловая система, содержащая файл, была смонтирована без запрета на SUID/SGID.

Таким образом, осуществляется управляемая передача прав одного пользователя другому. Права суперпользователя, соответственно, могут передаваться только с санкции суперпользователя и только в объеме функциональности, предоставляемой конкретной программой.

### 2.3.5.3. Access Control List

Расширенная модель дискреционного управления доступом контролирует доступ субъектов к объектам согласно специальным спискам, называемых списками контроля доступа. Списки контроля доступа «Access Control List» (далее – ACL) содержат данные обо всех субъектах (пользователях или группах ACL) и их правах доступа к требуемому объекту или типу объекта.

Информация о правах доступа представлена в виде последовательности байт, называемой маской прав доступа субъекта к объекту. В случае, если отсутствует отдельная маска прав доступа субъекта к объекту, права к родительскому объекту будут установлены в соответствии со значениями, принятыми для вершины иерархии наследственности. Наверху иерархии наследственности находится значение параметра ACL по умолчанию для каждого типа объекта.

В ОС Альт СП для каждого файла определен/задан ACL, который содержит основные (владелец, группа-владелец, остальные) и расширенные атрибуты доступа.

Расширенные атрибуты определяют права доступа к файлу для отдельного пользователя или отдельной группы пользователей.

Доступ к файлу определяется по следующему алгоритму в указанной последовательности:

- если пользователь является владельцем файла, то доступ осуществляется в соответствии с правами владельца;
- если для пользователя существует специальная запись в ACL, то используется эта запись;

- если пользователь является членом группы-владельца файла, то доступ осуществляется в соответствии с правами группы-владельца.

В ACL просматриваются записи о группах. Как только найдена первая группа, членом которой является данный пользователь, дальнейший просмотр прекращается и используются права доступа этой группы.

В случае, если пользователь не является членом ни одной из указанных в списке групп, используются права, определенные для остальных.

ACL используются для доступа к файлам и директориям.

#### 2.3.5.4. Модель защиты механизмов IPC (межпроцессного взаимодействия)

##### 2.3.5.4.1. Сокеты

Сокет домена UNIX (Unix domain socket, UDS) или IPC-сокеты (сокеты межпроцессного взаимодействия) – конечная точка обмена данными между процессами, работающими в одной и той же системе UNIX.

Доменные соединения UNIX являются по существу байтовыми потоками, схожими с сетевыми соединениями, но при этом все данные остаются внутри одного компьютера (то есть обмен данными происходит локально).

UDS использует файловую систему как адресное пространство имен, то есть они представляются процессами как индексные дескрипторы в файловой системе (системой создается специальный файл сокета по заданному пути). Это позволяет двум различным процессам открывать один и тот же сокет для взаимодействия между собой (через файл сокета любые локальные процессы смогут общаться путем чтения/записи из него). Однако, конкретное взаимодействие, обмен данными, не использует файловую систему, а только буферы памяти ядра.

Несмотря на то, что другие процессы распознают файлы сокетов как элементы каталога, чтение и запись файлов сокета могут осуществлять только те процессы, между которыми установлено соответствующее соединение.

##### 2.3.5.4.2. Каналы

Именованные каналы (named pipes, FIFOs) подобны сокетам, поскольку тоже используются для взаимодействия между процессами, однако, в отличие от сокетов,

в именованных каналах данные передаются только в одном направлении. Именованный канал создается явно с помощью команды `mkfifo`, и два различных процесса могут обратиться к нему по имени.

Неименованный канал представляет собой программный однонаправленный канал передачи данных между двумя родственными процессами (родителем и потомком). Посторонний субъект вмешаться в обмен данными не может, так как обращение к неименованным каналам осуществляется только через механизм файловых дескрипторов, которые наследуются при порождении нового процесса.

#### 2.3.5.4.3. System V IPC

При использовании системы межпроцессного взаимодействия System V IPC объектами доступа являются семафоры, очереди сообщений и разделяемая память. Доступ к этим объектам осуществляется с помощью ключей, функционально эквивалентных файловым дескрипторам. У каждого объекта System V IPC имеется набор прав доступа к нему, аналогичный набору для файлов, то есть разрешение чтения содержимого объекта для владельца, группы и остальных пользователей, разрешение изменения объекта и разрешение управления объектом.

#### 2.3.5.5. Алгоритм функционирования модели дискреционного управления доступом

Модель дискреционного управления доступом функционирует в соответствии со следующим алгоритмом:

- 1) субъект дискреционного доступа выполняет запрос на доступ к объекту;
- 2) выполняется проверка прав доступа субъекта доступа к объекту в соответствии со списками контроля доступа;
- 3) в списках контроля доступа осуществляется поиск записи, в которой определены разрешения для данного субъекта доступа;
- 4) если запись для субъекта доступа найдена, выполняется установка разрешений для субъекта, после чего субъект получает доступ к объекту дискреционного доступа с установленными правами доступа;

- 5) если запись для субъекта доступа не найдена, выполняется проверка соответствия идентификаторов субъекта доступа и владельца объекта, если идентификаторы совпадают, для субъекта устанавливаются права в соответствии с разрешениями, используемыми для пользователя-владельца объекта;
- 6) если идентификаторы не совпадают, выполняется проверка соответствия идентификаторов групп для субъекта и объекта, если идентификаторы совпадают, для субъекта устанавливаются права в соответствии с разрешениями, используемыми для группы-владельца объекта;
- 7) если идентификаторы не совпадают, для субъекта устанавливаются права в соответствии с разрешениями, используемыми для пользователей, не являющихся владельцами объекта.

Алгоритм функционирования модели дискреционного управления доступом приведен на рисунке (рис. 3).

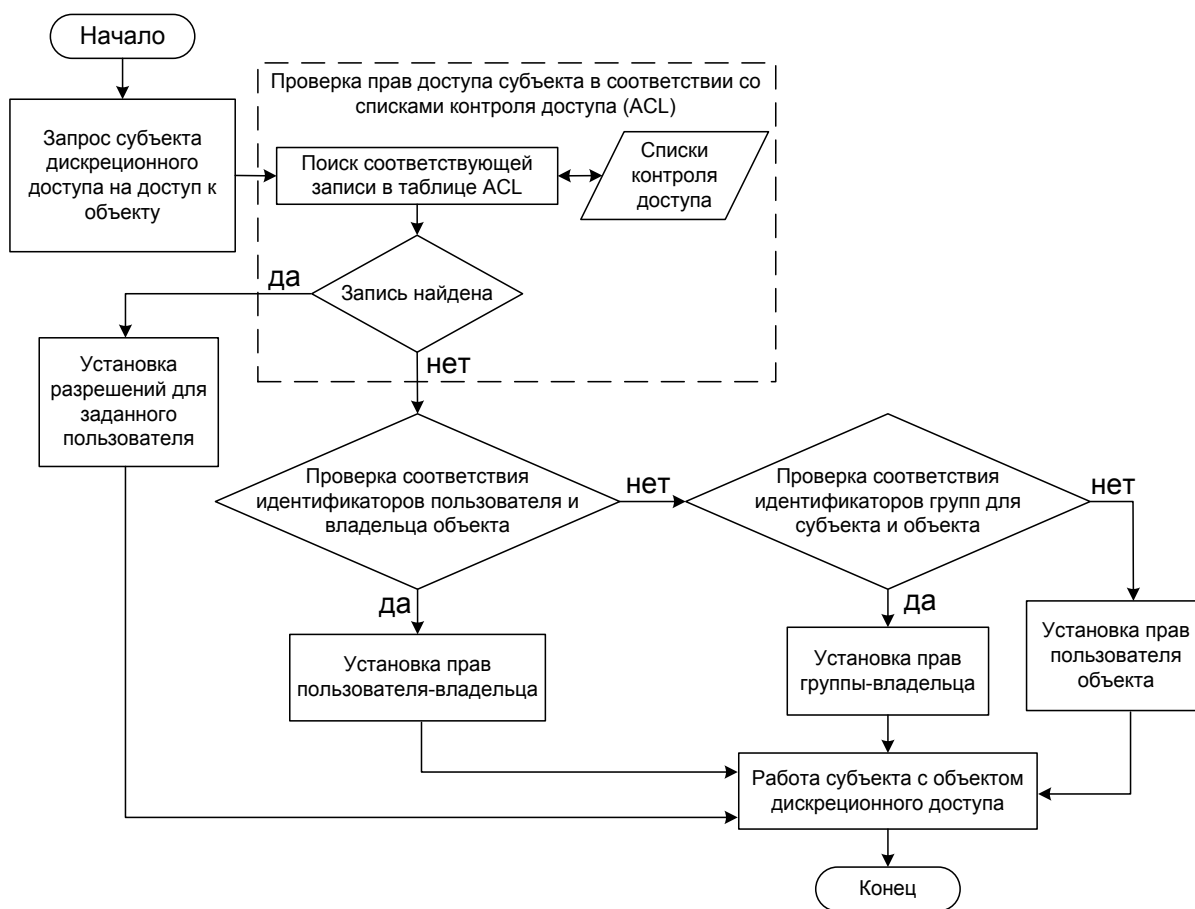


Рис. 3 – Алгоритм функционирования модели дискреционного управления доступом



### 2.3.6. Модель управления памятью

#### 2.3.6.1. Средства управления оперативной памятью

В качестве схемы управления памятью в ОС Альт СП используется виртуальная память.

Информация, с которой работает программа (активный процесс), располагается в оперативной памяти. В схемах виртуальной памяти процесс предполагает, что вся необходимая ему информация имеется в основной памяти (выделенной ему памяти).

Для этого занимаемая процессом память разбивается на несколько частей (например, страниц). Логический адрес (логическая страница), к которому обращается процесс, динамически транслируется в физический адрес (физическую страницу).

В тех случаях, когда страница, к которой обращается процесс, не находится в физической памяти, нужно организовать ее подкачку с диска. Для контроля наличия страницы в памяти вводится специальный бит присутствия, входящий в состав атрибутов страницы в таблице страниц.

Таким образом, в наличии всех компонентов процесса в основной памяти необходимости нет. Вследствие такой организации, размер памяти, занимаемой процессом, может быть больше, чем размер оперативной памяти.

##### 2.3.6.1.1. Изоляция

Каждый процесс работает со своими виртуальными адресами (в своем виртуальном адресном пространстве), трансляция которых в физические выполняется на аппаратном уровне с помощью ядра ОС Альт СП.

Пользовательский процесс лишен возможности напрямую обратиться к страницам основной памяти, занятым информацией, относящейся к другим процессам. В результате процессы становятся изолированными друг от друга.

Физическая память распределяется независимо от распределения виртуальной памяти отдельного процесса.

### 2.3.6.1.2. Очистка памяти

В ходе выполнения в ОС Альт СП санкционированных процессов, связанных с обработкой информации, в оперативной памяти ПЭВМ присутствует остаточная информация.

Очистка оперативной памяти осуществляется посредством записи нулей или маскирующей информации в память при ее освобождении (перераспределении).

Необходимость подобных мер объясняется следующими причинами:

- создание буферных и «теневых» областей памяти с целью обеспечения необходимой производительности;
- функционирование механизма виртуальной памяти, расширяющего объем оперативной памяти за счет внешней;
- использование буфера обмена данными между процессами (приложениями).

Таким образом, по завершении работы активного процесса КСЗ осуществляет очистку оперативной памяти (RAM-памяти), предоставляемой этому процессу. Очистка производится записью нулей или маскирующей информации в память при ее назначении пользователю или освобождении.

Очистка освобождаемых областей оперативной памяти происходит в процессе перевода ядром ОС каждой страницы памяти в разряд «неиспользуемых» (free). Это означает, в числе прочего, что ни одна страница из числа неиспользуемых не будет содержать данных, которые там размещала ОС Альт СП или приложения в процессе работы системы. Ядро высвобождает страницы, начинающиеся с указанной, размера [размер\_страницы \* (2 ^ кратность)]. Область возвращается в массив свободных областей в соответствующую позицию и после этого происходит попытка объединить несколько областей для создания одной большего размера.

В работающей системе информация об очистке освобождаемых областей памяти доступна в каталоге виртуальной служебной файловой системы /sys/kernel/mm/sanitize\_memory/. Здесь файл level содержит значение параметра smem, а файл count – количество памяти в байтах, обработанных

средствами очистки. Таким образом, если ядро загружено без поддержки очистки освобождаемых областей памяти, указанный каталог не создается.

Очистка освобождаемых областей памяти не распространяется на swar. В связи с этим, если требуется использование этой функции, swar не должен использоваться (не должен быть подключен или должен вообще отсутствовать).

#### 2.3.6.2. Средства управления внешней памятью

Внешняя память, используемая ОС, располагается на отдельном разделе диска, представленном в файловой системе специальным файлом, доступ к которому непосредственно из программы контролируется дискреционными ПРД. По умолчанию доступ к разделам диска имеет только доверенный субъект или член группы «disk».

При первоначальном назначении или при перераспределении внешней памяти КСЗ может ограничивать доступ субъекта к остаточной HDD-информации через механизм «безопасного удаления» файлов (специальный атрибут файла, указывающий на необходимость перезаписи физической области носителя диска после удаления файла). Еще одним способом является использование команды shred, обеспечивающей безопасное удаление файлов.

#### 2.3.7. Модель контроля целостности

Под целостностью подразумевается свойство неизменности исполняемого программного кода и настроек.

По перечню параметров, определяемому в конфигурационном файле, создается база данных, в которой содержится определенный набор параметров, описывающих состав и конфигурацию программного кода. Считается, что эти параметры должны оставаться неизменными в процессе функционирования системы, а любое изменение одного из них является сигналом о том, что была произведена попытка несанкционированного доступа. Также на основе различных алгоритмов создается контрольная сумма, которая должна оставаться неизменной. База данных, созданная в условиях, когда попытка нелегального доступа невозможна, считается эталонной.

Выбранные объекты системы с параметрами, занесенными в эталонную базу данных, хранятся в их текущем состоянии в качестве эталонных. Хранение файлов организовывается с помощью средств резервного копирования и восстановления и производится на отдельных резервных носителях.

Созданная в процессе работы база данных считается рабочей. Рабочая база данных сравнивается с эталонной во время загрузки ОС Альт СП и далее периодически в процессе работы. Параметры сравнения определяются конфигурационным файлом.

В случае совпадения баз данных считается, что целостность системы не нарушена.

В случае возникновения события несоответствия рабочей и эталонной баз данных, информация о событии заносится в системный журнал при помощи средств протоколирования. При изменении в контролируемых файлах система принудительно перейдет в однопользовательский режим (см. п. 3.6.2.3) с возможностью после ввода пароля администратора просмотреть системный журнал или внести исправления.

При необходимости, с помощью средств резервного копирования и восстановления извлекается последняя рабочая копия хранимых объектов системы с корректными контрольными суммами. Далее выполняется замена измененных (поврежденных) объектов извлеченными.

#### 2.3.7.1. Средства контроля целостности

В ОС Альт СП для обнаружения различий между двумя состояниями системы предусмотрено использование ПК osec. Также ПК osec предназначен для поиска потенциально опасных файлов, например, файлов с установленными битами прав смены идентификаторов пользователя (suid), группы (sgid) и с общедоступной записью.

ПК osec состоит из двух частей:

- osec – программа сбора данных, результаты работы по умолчанию подаются в неформатированном виде в стандартный поток вывода stdout;

- osec\_reporter – программа-фильтр для создания отчетов, принимает на вход неформатированный вывод ПК osec и представляет данные в удобном виде для чтения (результаты работы также подаются в стандартный поток вывода stdout).

ПК osec может работать в режимах «только для чтения» и «чтение-запись» (по умолчанию). В режиме «чтение-запись» ПК osec сообщает об обнаруженных изменениях и сохраняет новое состояние системы в свою базу данных. Для каждого контролируемого каталога ПК osec создает уникальный файл базы данных и помещает его в каталог базы данных, указанный в опции -D (db\_path).

### 2.3.7.2. Подсистема IMA/EVM

IMA/EVM (Integrity measurement architecture and Extended verification module) – подсистема, позволяющая осуществлять контроль целостности файловой системы, реализует механизм замкнутой программной среды. В ОС Альт СП IMA встроена в ядро ОС. IMA включает в себя две подсистемы – IMA-measurement (измерение) и IMA-appraisal (оценка). Первая собирает хеш-образы файлов, вторая сравнивает собранный хеш с сохраненным хешем и запрещает доступ в случае несоответствия (рис. 4).

Собранные хеш-образы хранятся в расширенных атрибутах файловой системы. Модуль расширенной проверки (EVM) предотвращает несанкционированные изменения этих расширенных атрибутов в файловой системе.

IMA/EVM основывается на работе LSM модуля. LSM (Linux Security Module) – фреймворк, позволяющий изменить стандартное поведение программы посредством перехвата системного вызова и передачи управления модулям безопасности системы (рис. 5).

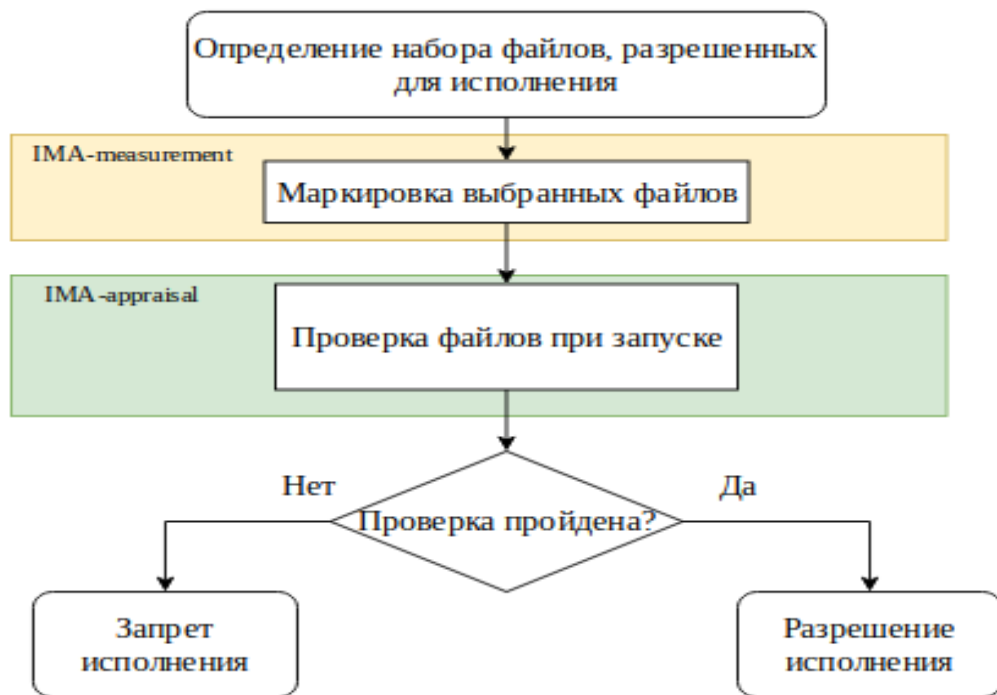


Рис. 4 – Схема функционирования

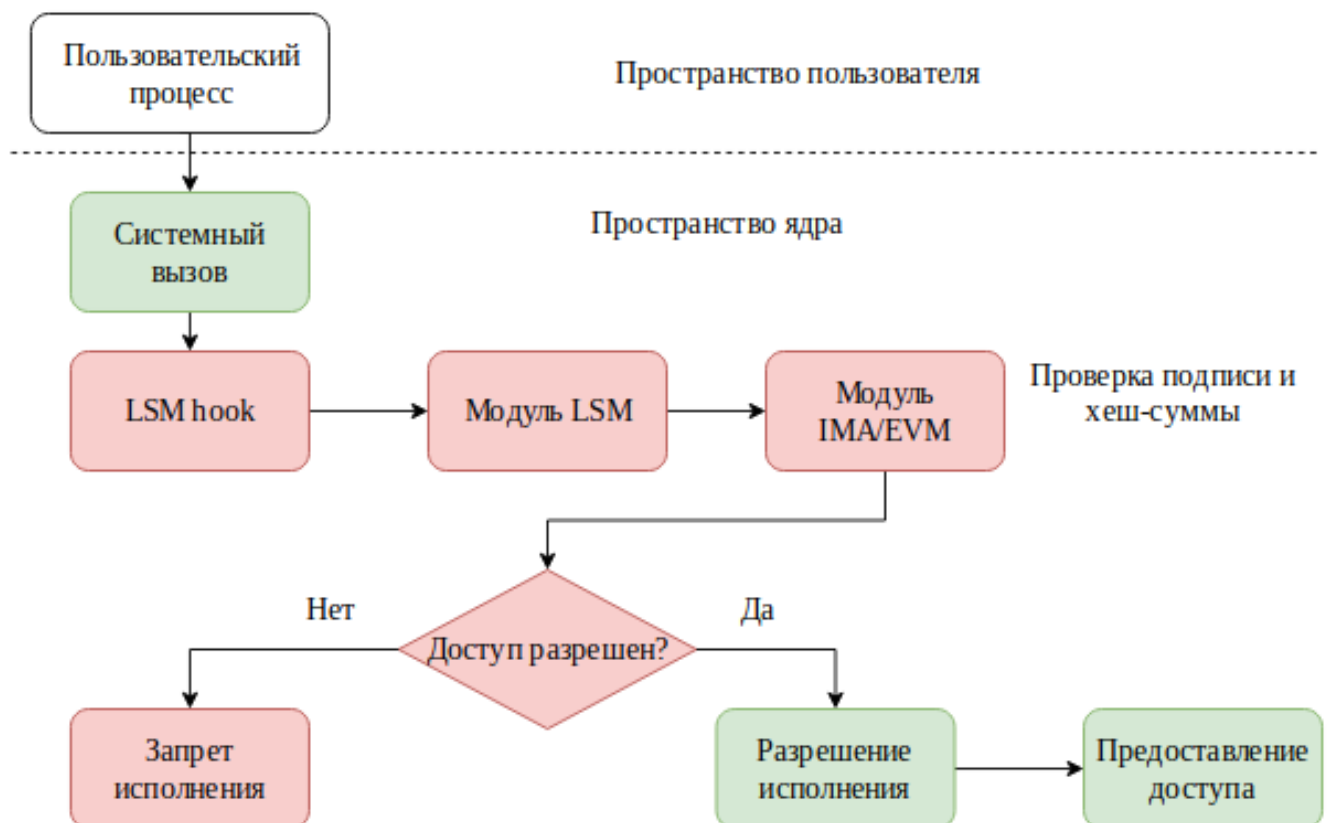


Рис. 5 – Перехват системного вызова и передача управления модулям безопасности

### 2.3.7.3. Контроль целостности КСЗ при загрузке ОС

Схема проверки целостности при загрузке системы показана на рис. 6.

К параметрам командной строки IMA относятся `ima_appraise`, `ima_policy`.

`ima_appraise` может принимать одно из четырех значений:

- `enforce` – «жесткий» режим, разрешает запуск программ только при прохождении проверки. В этом режиме IMA оценивает файлы в соответствии с политикой. Доступ к оцениваемому файлу запрещается, если хеш отсутствует или не соответствует собранному значению;
- `log` – режим «журналирования», аналогичен «жесткому» режиму, за исключением того, что доступ к измененному файлу не запрещается, а будет только зарегистрирован;
- `off` – отключает все оценки. Сохраненные хеши не проверяются, и новые хеши не создаются и не обновляются;
- `fix` – «мягкий» режим, используется на этапе настройки системы и разрешает запуск любых программ, регистрируя события несанкционированного доступа в журнале. Запуск в этом режиме применяется для первичной маркировки системы.

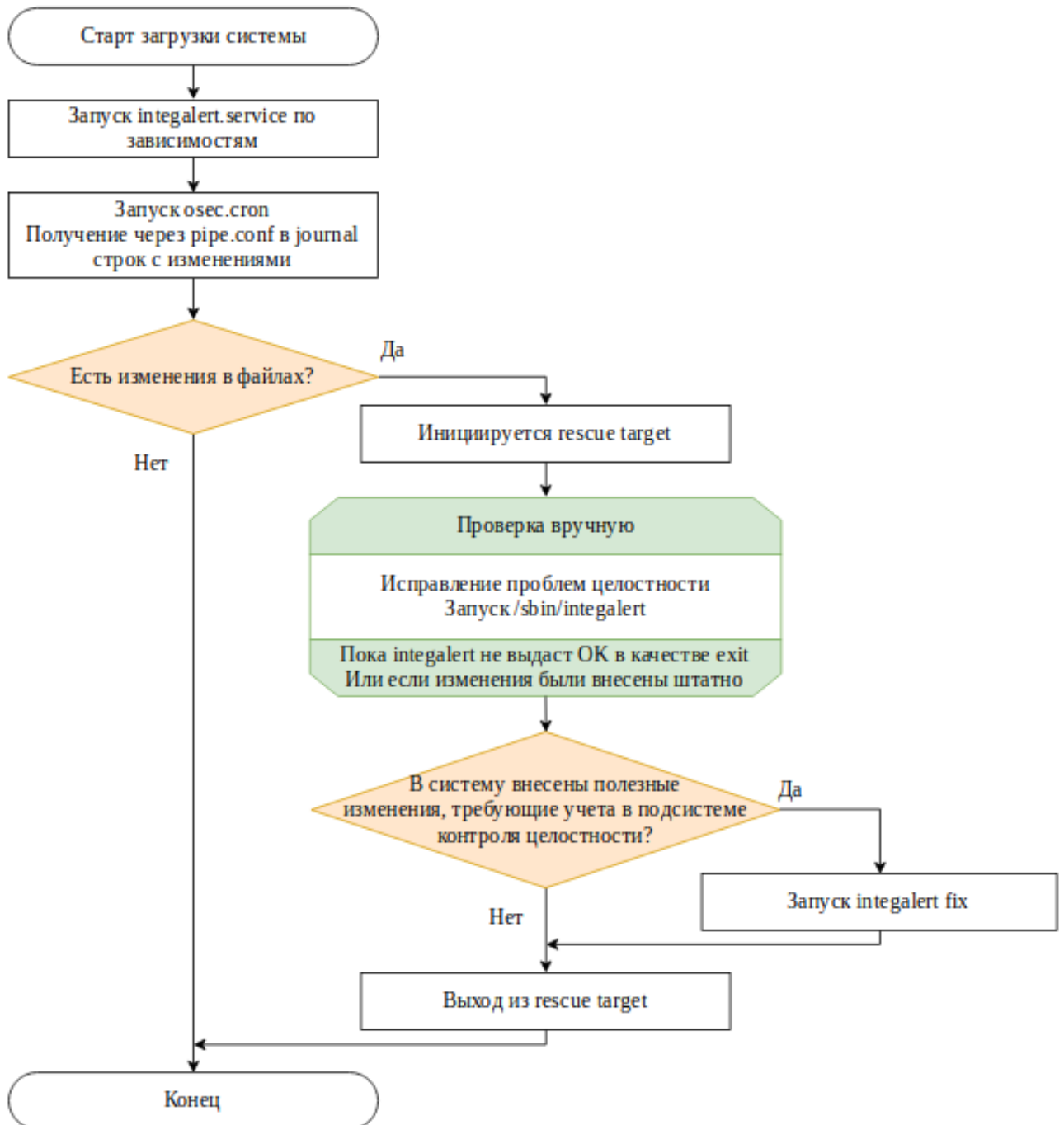


Рис. 6 – Схема проверки целостности при загрузке системы

ima\_policy может принимать одно из трех значений:

- tcb – измеряет все исполняемые файлы, все файлы, помеченные для выполнения (например, разделяемые библиотеки), все загруженные модули ядра и все загруженные прошивки. Кроме того, измеряются также файлы, открытые для чтения пользователем root;



- `appraise_tcb` – оценивает все файлы, принадлежащие пользователю `root`;
- `secure_boot` – оценивает все загруженные модули, прошивку, ядро и политики IMA.

`ima_policy` может быть указано несколько раз, и результатом является объединение политик.

Переопределить политику целостности (`ima_policy`) по умолчанию можно в файле `/etc/integrity/policy`.

В `/etc/integrity/config` можно настроить перечень каталогов для подписи и другие опции, такие как EVM.

### 2.3.8. Модель защиты ввода и вывода на отчуждаемый физический носитель

#### 2.3.8.1. Модель взаимодействия с устройствами ввода-вывода

Отчуждаемые физические носители могут рассматриваться относительно ОС Альт СП с двух точек зрения:

- как блочные или символьные устройства ввода-вывода;
- как блочное устройство, которое может быть смонтировано.

В первом случае устройство представляет собой специальный файловый объект, доступ к которому контролируется дискреционными ПРД обычным образом и, следовательно, ввод-вывод остается в рамках контроля этих правил. Во втором случае отчуждаемый носитель информации содержит в себе образ файловой системы (далее – ФС), которая и хранит данные. Данный носитель может быть смонтирован в заданный каталог, и при этом ФС носителя становится частью (представленной в виде поддерева) корневой ФС. Доступ к объектам данной ФС подчиняется дискреционным ПРД обычным образом и, следовательно, ввод-вывод на отчуждаемый носитель остается в рамках контроля этих правил.

Для ОС возможность санкционированного монтирования конкретным пользователем конкретных носителей с конкретными ФС определяется администратором.

### 2.3.8.2. Модель осуществления печати

В ОС Альт СП основной системой печати является сервер печати Common UNIX Printing System (CUPS).

В состав CUPS входят следующие компоненты:

- диспетчер очереди печати (планировщик);
- система фильтрации;
- Back-end-система.

Сервер печати CUPS функционирует в виде отдельной службы и может управляться выделенным администратором либо общим администратором (предусмотрена возможность частично передавать права по управлению заданиями пользователя). Сервер печати CUPS имеет собственный веб-интерфейс для администрирования, работающий через Internet Printing Protocol (далее – IPP), а также CUPS использует IPP в качестве основы для управления заданиями и очередями.

Сервер печати CUPS работает следующим образом:

- сервер печати принимает задание на печать от программы (активного процесса) и передает его диспетчеру очереди печати или планировщику;
- диспетчер очереди печати добавляет задание на печать в соответствующую очередь;
- диспетчер очереди печати передает задание на печать в соответствии с очередью системе фильтрации;
- система фильтрации обрабатывает данные: осуществляет все необходимые преобразования данных в соответствии с применяемыми для этого задания фильтрами и переводит их в формат понятный принтеру;
- Back-end-система отправляет переформатированные данные на устройства печати.

2.3.9. Модель обеспечения доверенной загрузки средств вычислительной техники

#### 2.3.9.1. Поддержка режима «Secure Boot»

ОС Альт СП поддерживает режим «Secure Boot» – функцию UEFI, предотвращающую запуск не авторизованных операционных систем и программного обеспечения во время запуска компьютера. В режиме «Secure Boot» загрузка различных ОС может выполняться только с заранее определенных постоянных носителей (например, только с жесткого диска) после успешного завершения специальных процедур: проверки целостности технических и программных средств ПЭВМ (с использованием механизма пошагового контроля целостности) и идентификации/аутентификации пользователя.

При помощи режима «Secure Boot» UEFI-совместимая прошивка может проверить подлинность исполняемых ей внешних компонентов (загрузчиков, драйверов и UEFI OptionROM). Эти исполняемые компоненты должны быть подписаны электронно-цифровой подписью, которая проверяется во время загрузки, и в случае ее полного отсутствия, повреждения, отсутствия в списке доверенных (db) или присутствия в списке запрещенных (dbx) запуск соответствующего компонента не происходит.

По умолчанию прошивка UEFI будет загружать только загрузчики, подписанные ключом, встроенным в прошивку UEFI – то есть будет выполняться загрузка в режиме «Secure Boot» (безопасной загрузки) или Trusted Boot (доверенной загрузки).

Загрузчик ОС Альт СП подписан доверенным ключом. Таким образом, ОС Альт СП можно загружать в режиме «Secure Boot», что позволяет предотвратить подмену загрузчика.

#### 2.3.10. Модель сопоставления пользователя с устройством

ОС Альт СП обеспечивает ввод-вывод информации на запрошенное пользователем устройство как для произвольно используемых им устройств, так и для идентифицированных (при совпадении маркировки).

ОС Альт СП включает в себя механизм сопоставления пользователя с устройством, реализованный в ОС, а также обеспечивает при проверке совпадения маркировок носителя и пользователя применение дискреционных ПРД.

Кроме того, в ОС Альт СП поддерживаются ограничение или запрет использования внешних носителей при помощи правил API библиотеки polkit и (или) udev.

### 2.3.11. Модель системы протоколирования событий

Система протоколирования событий позволяет сортировать сообщения по источникам и степени важности и направлять их в различные пункты назначения: на терминалы пользователей, на другие автоматизированные рабочие места (далее – АРМ) и в специальные файлы – системные журналы. Системный журнал – это база данных с информацией, сохраняемой в текстовом формате и отсортированной по времени исполнения.

В системе регистрируются следующие типы событий:

- использование идентификационного и аутентификационного механизма;
- запрос на доступ к защищаемому ресурсу – открытие файла, запуск программы и другие действия по его чтению и изменению;
- создание и уничтожение объекта;
- действия по изменению правил разграничения доступа;
- попытки доступа и действия администратора.

Программы отсылают записи, предназначенные для протоколирования, системному демону, который идентифицирует тип каждой пришедшей записи и обрабатывает запись способом, определенным для данного типа.

Для каждого из регистрируемых событий в журналах указывается следующая информация:

- дата и время;
- субъект, осуществляющий регистрируемое действие;
- тип события (если регистрируется запрос на доступ, то указываются объект и тип доступа);

- успешно ли осуществилось событие (обслужен запрос на доступ или нет).

### 2.3.11.1. Механизм журналирования

Механизм журналирования в ОС Альт СП функционирует по следующему алгоритму:

- программы (источники регистрируемых данных) формируют простые текстовые сообщения о происходящих в них событиях и передают их на обработку в ядро, инициализируя при этом системный вызов;
- системный демон `syslogd` (пакет `rsyslogd`) сравнивает каждую пришедшую запись с правилами, которые находятся в файле конфигурации `/etc/syslog.conf`, и когда обнаруживается соответствие, `syslogd` обрабатывает запись описанным в `syslog.conf` способом;
- формирование сообщений о событиях и их передача происходит по определенным правилам (протокол Syslog);
- передача текстовых сообщений происходит с использованием сетевых или доменных сокетов;
- источники сообщений могут располагаться на разных машинах;
- все регистрируемые сообщения по умолчанию записываются в каталог системного журнала `/var/log`, однако при необходимости могут быть указаны и другие хранилища (для каждого демона может быть свое хранилище, или несколько хранилищ).

### 2.3.11.2. Механизм аудита

Механизм аудита состоит из нескольких компонентов (рис. 7):

- 1) модуль ядра – перехватывает системные вызовы (`syscalls`) и выполняет регистрацию событий;
- 2) служба `auditd` – записывает зарегистрированное событие в файл;
- 3) служба `audispd` – осуществляет пересылку сообщений (выступает в роли диспетчера) к другому приложению;
- 4) ряд вспомогательных программ:

- auditctl – программа, управляющая поведением системы аудита и позволяющая контролировать текущее состояние системы, создавать или удалять правила;
- aureport – программа, генерирующая суммарные отчеты о работе системы аудита;
- ausearch – программа, позволяющая производить поиск событий в журнальных файлах;
- autrace – программа, выполняющая аудит событий, порождаемых указанным процессом.



Рис. 7 – Составные компоненты аудита

В ОС Альт СП регистрируются следующие типы событий:

- запуск и завершение работы ОС Альт СП (перезагрузка, остановка);
- запуск и остановка приложений;
- выполнение системных вызовов;
- использование механизма идентификации и аутентификации;
- запрос на доступ к защищаемому ресурсу – открытие файла, запуск программы и другие действия по его чтению и изменению;
- создание и уничтожение объекта;
- действия по изменению ПРД;
- инициация сетевого соединения или изменение сетевых настроек и другие.

Программы отсылают записи, предназначенные для протоколирования, системному демону `auditd`, который идентифицирует тип каждой пришедшей записи и обрабатывает запись способом, определенным для данного типа.

Для каждого из регистрируемых событий в журналах указывается следующая информация:

- дата и время;
- субъект, осуществляющий регистрируемое действие;
- тип события (если регистрируется запрос на доступ, то указываются объект и тип доступа);
- успешность осуществления события (обслужен запрос на доступ или нет).

### 2.3.12. Средства сбора сетевой статистики и фильтрации сетевых пакетов

В качестве средств сбора статистики сетевого взаимодействия и фильтрации сетевых пакетов в ОС Альт СП используется утилита `iptables`.

Утилита `iptables` позволяет фильтровать сетевые пакеты по следующим параметрам:

- на основе сетевых адресов отправителя и получателя (IP-адреса, MAC-адреса);
- по протоколам `tcp`, `udp`, `icmp`;
- с учетом входного и выходного сетевого интерфейса;
- на основе используемого порта;
- с учетом даты и времени.

### 2.3.13. Средства контроля запуска компонентов программного обеспечения

В ОС Альт СП механизм контроля запуска компонентов программного обеспечения реализуется при помощи программы `control++`.

`Control++` позволяет переключать режимы, каждый из которых определяется своим файлом ограничений, а также своим набором описаний прав на файлы системы и запускаемым сценарием оболочки.

`Control++` позволяет также контролировать соответствие между параметрами установленного ранее режима и текущим состоянием системы.

### 3. УПРАВЛЕНИЕ КСЗ

Запуск ОС Альт СП выполняется автоматически после запуска ПЭВМ и отработки набора программ BIOS.

КСЗ стартует вместе с ОС Альт СП, проходя непосредственно через этапы досистемной загрузки, тестирования ядром окружения процессов и запуска файловой системы (далее – ФС), а также инициализации первичного процесса `init`.

#### 3.1. Использование API библиотеки `polkit`

API библиотеки `polkit` используется для предоставления непривилегированным процессам возможности выполнения действий, требующих прав администратора. При этом `Polkit` не наделяет процесс пользователя правами администратора, а позволяет контролировать, что разрешено, а что запрещено.

`Polkit` также позволяет пользователям получить временное разрешение посредством аутентификации либо администратора, либо пользователя.

Любой запрос на выполнение действия в системном контексте, поступивший от работающего пользовательского процесса, отслеживается `polkit`. В соответствии с имеющимися правилами, `polkit` принимает решение о том, может ли быть выполнено это действие, и если может, то при выполнении каких условий. Принятое решение (запрет, разрешение или разрешение с условием) передается системной программе, которая затем действует соответствующим образом. Таким образом, при взаимодействии пользовательского процесса (`Subject`) и привилегированного системного процесса (`Mechanism`) `polkit` выступает в качестве третьей стороны, принимающей решение о санкционированности действий.

##### 3.1.1. Файлы действий

Все политики `polkit` находятся в `/usr/share/polkit-1/actions/` в формате `*.policy`.



Имена файлов составлены из названия разработчика программного обеспечения (вендора), названия программы или группы действий и заканчиваются словом `policy`.

Имя каждого файла вполне соответствует той группе действий, которая в нем перечислена. Средняя часть имени файла – название программы или группы действий – является в данном случае смысловой.

Каждая политика представляет собой `xml`-файл, в котором описываются запросы к `polkit`. Каждый запрос имеет три условия, прописанных в секции `defaults`:

- запрос от любого пользователя – тег `<allow_any>`;
- запрос от неактивного пользователя – тег `<allow_inactive>`;
- запрос от активного пользователя – тег `<allow_active>`.

Внутри каждого тега прописывается возвращаемое значение. Используются следующие варианты значений:

- `yes` – предоставить разрешения;
- `no` – заблокировать разрешения;
- `auth_self` – пользователь должен ввести свой пароль для аутентификации;
- `auth_self_keep` – пользователь должен ввести свой пароль для аутентификации один раз за сессию, разрешение предоставляется для всей сессии;
- `auth_admin` – пользователь должен ввести пароль администратора при каждом запросе;
- `auth_admin_keep` – пользователь должен ввести пароль администратора, разрешение предоставляется для всей сессии.

Представленные в директории `/usr/share/polkit-1/actions/` правила являются принятыми по умолчанию. Допускается их изменение путем редактирования `XML`-файлов `.policy`. Однако необходимо учитывать, что данный метод не является рекомендуемым, так как при обновлении программ внесенные изменения будут перезаписаны настройками по умолчанию.

### 3.1.2. Файлы правил

Изменять правила формата `.policy` рекомендуется путем использования файлов типа `.rules`, которые переопределяют правила, установленные по умолчанию в файлах действий `.policy`.

Файлы `.rules` расположены в двух каталогах:

- `/etc/polkit-1/rules.d` – предполагается, что здесь располагаются некоторые файлы правил, подготовленные разработчиками дистрибутива и все файлы правил, подготовленные администратором. При персональной настройке правил располагать соответствующие файлы надо именно в этом каталоге;
- `/usr/share/polkit-1/rules.d` – данный каталог содержит файлы правил, которые написаны разработчиками приложений и дистрибутива. Размещать файлы со своими правилами здесь настоятельно не рекомендуется из-за того, что при обновлении программ сделанные изменения, скорее всего, пропадут.

Настраивать правила можно как правкой существующих файлов `.rules`, так и созданием новых в каталоге `/etc/polkit-1/rules.d`. Создание новых файлов `.rules` является более безопасным и надежным методом, поскольку позволяет быстро и без потерь вернуться к исходным настройкам путем удаления файла с некорректно заданным правилом, вызвавшим сбой.

Задавая имя файла, можно просто и надежно определить порядок чтения файлов `.rules`. Например, путем внесения в начале имени файла порядкового номера. При этом файлы гарантированно будут читаться в том же порядке, в котором возрастают эти числа.

Необходимо учитывать, что правила, которые содержатся в файлах, прочитанных раньше, переопределяют правила в файлах, прочитанных позже.

Алгоритм создания правила (все действия выполняются от `root`):

- 1) определить какую политику нужно изменить, для этого необходимо найти в `/usr/share/polkit-1/actions/` требуемую.

2) создать новое правило:

```
touch /etc/polkit-1/rules.d/99-vashe_pravilo.rules
```

3) открыть на редактирование созданный файл:

```
mcedit /etc/polkit-1/rules.d/99-vashe_pravilo.rules
```

## 4) вставить текст:

```

polkit.addRule(function(action, subject) {
  if (action.id.match("действие") &&
      subject.isInGroup("группапользователей")) {
    return polkit.Result.правило;
  }
});

```

где:

- действие – это значение id в элементе action в нужном файле действий .policy;
- группа пользователей – это одна из реально существующих в операционной системе групп. Например, это может быть та группа, которая была создана при заведении пользователя в системе;
- правило – это следующие значения: NO, YES, AUTH\_SELF, AUTH\_ADMIN, AUTH\_SELF\_KEEP, AUTH\_ADMIN\_KEEP.

## 3.1.3. Журналирование действий polkit

Используя правила polkit можно также делать записи в системный журнал. Метод log() записывает сообщение в системный журнал. Записи журнала используют флаг LOG\_AUTHPRIV, поэтому записи будут производиться в файл /var/log/secure.

Пример:

```

polkit.addRule(function(action, subject) {
  if (action.id == "действие") {
    polkit.log("action=" + action);
    polkit.log("subject=" + subject);
  }
});

```

В параметре action передается объект с информацией о совершенном процессе и связанные с этим действием параметры (например, если запрошенное действие монтирование съемного диска, то в параметре action будут переданы серийный номер диска, его id, файловая система и т. д.).

В параметре subject передается объект с информацией о пользователе, запустившем процесс.

Этот объект имеет следующие атрибуты:

- `id` – идентификатор процесса;
- `user` – имя пользователя;
- `groups` – список групп, в которые входит пользователь;
- `seat` – местонахождение субъекта (пустое значение, если местонахождение не локальное);
- `session` – сессия субъекта;
- `local` – `true`, только если местонахождение имеет локальный характер;
- `active` – `true`, только если сеанс активен.

### 3.2. Средства управления учетными записями пользователей

Создание, редактирование и удаление учетных записей пользователей выполняется администратором в соответствии с руководящими решениями объекта автоматизации. Идентификация пользователя (присвоение ему кода UID) может обеспечиваться как автоматически КСЗ, так и вручную администратором по своему усмотрению.

При добавлении пользователя в ОС администратор выдает ему регистрационное имя (идентификатор) для входа в систему и пароль, который служит для подтверждения идентификатора пользователя. В дальнейшем КСЗ обеспечивает аутентификацию пользователя, то есть его распознавание по имени и паролю. Вводимые пользователем символы пароля не отображаются на экране терминала.

**Примечание.** В случае работы ОС в графическом режиме символы пароля заменяются звездочками.

Администратор и (или) пользователь могут изменить пароль командой `passwd`. При вводе этой команды ОС Альт СП запрашивает ввод текущего пароля, а затем требует ввести новый пароль.

В случае, если предложенный пароль слишком прост, ОС Альт СП может попросить ввести другой. Также если предложенный пароль удовлетворителен, ОС Альт СП просит ввести его повторно с тем, чтобы убедиться в корректности ввода пароля.

### 3.2.1. Общая информация

Для всех пользователей и групп внутри ОС Альт СП введены собственные цифровые идентификаторы в категориях UID и GID соответственно.

Пользователь может входить в одну или несколько групп. По умолчанию он входит в группу, совпадающую с его именем. Для получения сведений о том, в какие еще группы входит пользователь, необходимо выполнить команду:

```
id user_name
```

где `user_name` – имя пользователя.

Например:

```
$ id test
uid=500(test) gid=500(test) группы=500(test),16(rpm)
```

Такая запись означает, что пользователь `test` (цифровой идентификатор 500) входит в группы `test` и `rpm`. Разные группы могут иметь разный уровень доступа к тем или иным каталогам; чем в большее количество групп входит пользователь, тем больше прав он имеет в системе.

**Примечание.** В связи с тем, что в дистрибутивах изделия большинство привилегированных системных утилит имеют не SUID-, а SGID-бит, будьте предельно внимательны и осторожны в переназначении групповых прав на системные каталоги.

### 3.2.2. Обязка passwd

Для обновления аутентификационных данных пользователя используется `passwd`. Обязка `passwd` поддерживает традиционные опции `passwd` и утилит `shadow`.

Синтаксис:

```
passwd [ОПЦИЯ...] [ИМЯ ПОЛЬЗОВАТЕЛЯ]
```

Опции:

- 1) `-d, --delete` – удалить пароль для указанной записи;
- 2) `-f, --force` – форсировать операцию;
- 3) `-k, --keep-tokens` – сохранить не устаревшие пароли;
- 4) `-l, --lock` – заблокировать указанную запись;
- 5) `--stdin` – прочитать новые пароли из стандартного ввода;
- 6) `-S, --status` – вывести отчет о статусе пароля в указанной записи;
- 7) `-u, --unlock` – разблокировать указанную запись;
- 8) `-?, --help` – показать справку и выйти;
- 9) `--usage` – дать короткую справку по использованию;
- 10) `-v, --version` – вывести версию программы.

**Примечание.** При успешном завершении команда `passwd` заканчивает работу с кодом выхода «0». Код выхода «1» означает, что произошла ошибка. Текстовое описание ошибки выводится на стандартный поток ошибок.

Только суперпользователь может обновить пароль другого пользователя.

### 3.2.3. Добавление нового пользователя

Для добавления нового пользователя необходимо выполнить команду `useradd`:

```
# useradd test
```

где `test` – имя нового пользователя (может быть выбрано любое имя, отличное от уже имеющихся в системе).

После добавления пользователя в систему необходимо выполнить установку пароля для учетной записи пользователя, для этого используется команда `passwd`:

```
# passwd test
```

После выполнения данной команды в консоль будет выведено сообщение, с предложением о вводе пароля для учетной записи пользователя `Enter new password:` (Введите новый пароль:), в ответ на которое необходимо ввести пароль (длина и состав пароля должны удовлетворять наложенным парольным ограничениям).

После ввода пароля для учетной записи пользователя, в консоль будет выведено сообщение с предложением о повторном вводе пароля для исключения ошибок при вводе `Retype new password:` (Повторите ввод пароля:), в ответ на которое необходимо выполнить повторный ввод пароля.

В случае, если введенные пароли совпали, в консоль будет выведено сообщение `passwd: all authentication tokens updated` (пароль: все маркеры проверки подлинности обновлены), что в свою очередь свидетельствует об успешной установке пароля для пользователя.

В результате описанных действий в системе будет создан пользователь `test` с заданным паролем.

В случае, если пароль оказался небезопасным, на экран будет выведено соответствующее сообщение с предупреждением:

1) если пароль короткий:

`«Weak password: too short»` («Ненадежный пароль: слишком короткий»)

2) если пароль не соответствует требованиям к классам используемых символов (пароль должен состоять из букв верхнего и нижнего регистра, цифр и других символов):

`«Weak password: not enough different characyers or classes for this length»` («Ненадежный пароль: недостаточно классовых различий между используемыми символами»)

3) если при повторном вводе пароли не совпали, в консоль будет выведено предупреждение, с предложением ввести новый пароль, а также ограничения на допустимы символы:

`«Sorry, Passwords do not match. Try again.  
You can now choose the new password or passphrase.  
A valid password should be a mix of upper and lower case letters,  
digits, and other characters. You can use a 8 character long  
password with characters from at least 3 of these 4 classes.  
An upper case letter that begins the password and a digit that  
ends it do not count towards the number of character classes  
used.  
A passphrase should be of at least 3 words, 6 to 40 characters  
long, and contain enough different characters.»`



Alternatively, if no one else can see your terminal now, you can pick this as your password: "high-worth\*outset"»

**Примечание.** Далее приводится перевод системного сообщения с предложением ввести новый пароль с учетом ограничений на допустимые символы:

«Пароли не совпадают. Попробуйте еще раз. Вы можете выбрать новый пароль или парольную фразу. Правильный пароль должен сочетать в себе прописные и строчные буквы, цифры и другие символы. Вы можете использовать пароли длиной из восьми символов, относящихся, по крайней мере, к трем из четырех классов. Буква верхнего регистра, с которой начинается пароль и цифра, которой пароль заканчивается, не учитываются при подсчете количества используемых классов символов. Парольная фраза должна состоять, по крайней мере, из трех слов длиной от 6 до 40 символов, содержащих разные классы символов. Кроме того, если информация, отображаемая в настоящее время на терминале, недоступна для просмотра посторонним лицам, в качестве пароля вы можете использовать «high-worth\*outset»

В дальнейшем пользователь может изменить свой пароль при помощи команды `passwd`.

В ОС Альт СП для проверки паролей на слабость используется подключаемый модуль аутентификации (Pluggable Authentication Modules, далее – PAM) `passwdqc`.

Программа `useradd` имеет множество параметров, которые позволяют менять ее поведение по умолчанию. В том числе предоставляется возможность принудительно указать, какой будет UID или к какой группе будет принадлежать пользователь.

Синтаксис программы `useradd`:

```
useradd [параметры] LOGIN
useradd -D
useradd -D [параметры]
```

При вызове без опции `-D`, команда `useradd` создает новую учетную запись пользователя, используя значения, указанные в командной строке плюс значения по умолчанию из системы. В зависимости от параметров командной строки, команда `useradd` будет обновлять системные файлы, а также может создать домашний каталог нового пользователя и скопировать исходные файлы.

Опции:

- 1) `-c, --comment КОММЕНТАРИЙ` – любая текстовая строка. Обычно здесь коротко описывается учетная запись, и в настоящее время используется как поле для имени и фамилии пользователя;
- 2) `-b, --base-dir БАЗОВЫЙ_КАТАЛОГ` – базовый каталог для пользователя. Если параметр `HOME_DIR` не указан, то базовый каталог определяется по имени пользователя;
- 3) `-d, --home ДОМАШНИЙ_КАТАЛОГ` – для создаваемого пользователя будет использован каталог `БАЗОВЫЙ_КАТАЛОГ` в качестве домашнего каталога. По умолчанию, это значение получается объединением имени пользователя с `БАЗОВЫМ_КАТАЛОГОМ` и используется как имя домашнего каталога;
- 4) `-e, --expiredate ДАТА_УСТАРЕВАНИЯ` – дата, когда учетная запись пользователя будет заблокирована. Дата задается в формате ГГГГ-ММ-ДД;
- 5) `-f, --inactive ДНЕЙ` – число дней, которые должны пройти после устаревания пароля, чтобы учетная запись заблокировалась навсегда. Если указано значение 0, то учетная запись блокируется сразу после устаревания пароля, а при значении `-1` данная возможность не используется. По умолчанию используется значение `-1`;
- 6) `-g, --gid ГРУППА` – имя или числовой идентификатор новой начальной группы пользователя. Идентификатор группы должен указывать на уже существующую группу. Идентификатор группы по умолчанию равен 1 или значению, указанному в файле `/etc/default/useradd`;
- 7) `-G, --groups ГРУППА1[,ГРУППА2,...[,ГРУППАН]]` – список дополнительных групп, в которых числится пользователь. Перечисление групп осуществляется через запятую, без промежуточных пробелов. На указанные группы действуют те же ограничения, что и для группы, указанной в параметре `-g`. По умолчанию пользователь входит только в начальную группу;
- 8) `-h, --help` – показать краткую справку и закончить работу;

- 9) `-m, --create-home` – если домашнего каталога пользователя не существует, то он будет создан;
- 10) `-k, --key КЛЮЧ=ЗНАЧЕНИЕ` – используется для изменения значений по умолчанию, хранимых в файле `/etc/login.defs` (`UID_MIN`, `UID_MAX`, `UMASK`, `PASS_MAX_DAYS` и других). Пример: `-k PASS_MAX_DAYS=-1` можно использовать при создании системной учетной записи, чтобы выключить устаревание пароля, даже если системная учетная запись вообще не имеет пароля;
- 11) `-N, --no-user-group` – не создавать группу с тем же именем, что и пользователь, но добавить пользователя к группе, указанной опцией `-g` или переменной группы в файле `/etc/default/useradd`;
- 12) `-o, --non-unique` – позволяет создать учетную запись с уже имеющимся (не уникальным) `UID`;
- 13) `-p, --password ПАРОЛЬ` – шифрованное значение пароля, которое возвращает функция `crypt`. По умолчанию учетная запись заблокирована;
- 14) `-s, --shell ОБОЛОЧКА` – имя регистрационной оболочки пользователя. Если задать пустое значение, то будет использована регистрационная оболочка по умолчанию;
- 15) `-u, --uid UID` – числовое значение идентификатора пользователя (`ID`). Оно должно быть уникальным, если не используется параметр `-o`. Значение должно быть неотрицательным. По умолчанию используется наименьшее значение `ID` большее 999 и большее любого другого значения пользователя. Значения от 0 до 999 обычно зарезервированы для системных учетных записей;
- 16) `-U, --user-group` – создать группу с тем же именем, что и пользователь, и добавить пользователя в эту группу.

При вызове команды `useradd` только с опцией `-D` покажет текущие значения по умолчанию:

```
# useradd -D
```

```
GROUP=100  
HOME=/home  
INACTIVE=-1  
EXRIFE=  
SHELL=/bin/bash  
SKEL=/etc/skel  
CREATE_MAIL_SPOOL=yes
```

### 3.2.4. Добавление/редактирование пользователей в графической оболочке и в веб-интерфейсе

Модуль Центра управления системой (ЦУС) «Локальные учетные записи» (пакет `alterator-users`) предназначен для администрирования локальных пользователей. Модуль «Локальные учетные записи» доступен как в GUI в экспертном режиме (раздел «Пользователи» → «Локальные учетные записи») (рис. 8), так и в веб-интерфейсе по адресу `https://ip-address:8080` (раздел «Пользователи» → «Локальные учетные записи») (рис. 9). Данный модуль позволяет создавать новых пользователей, редактировать и удалять уже существующих пользователей.

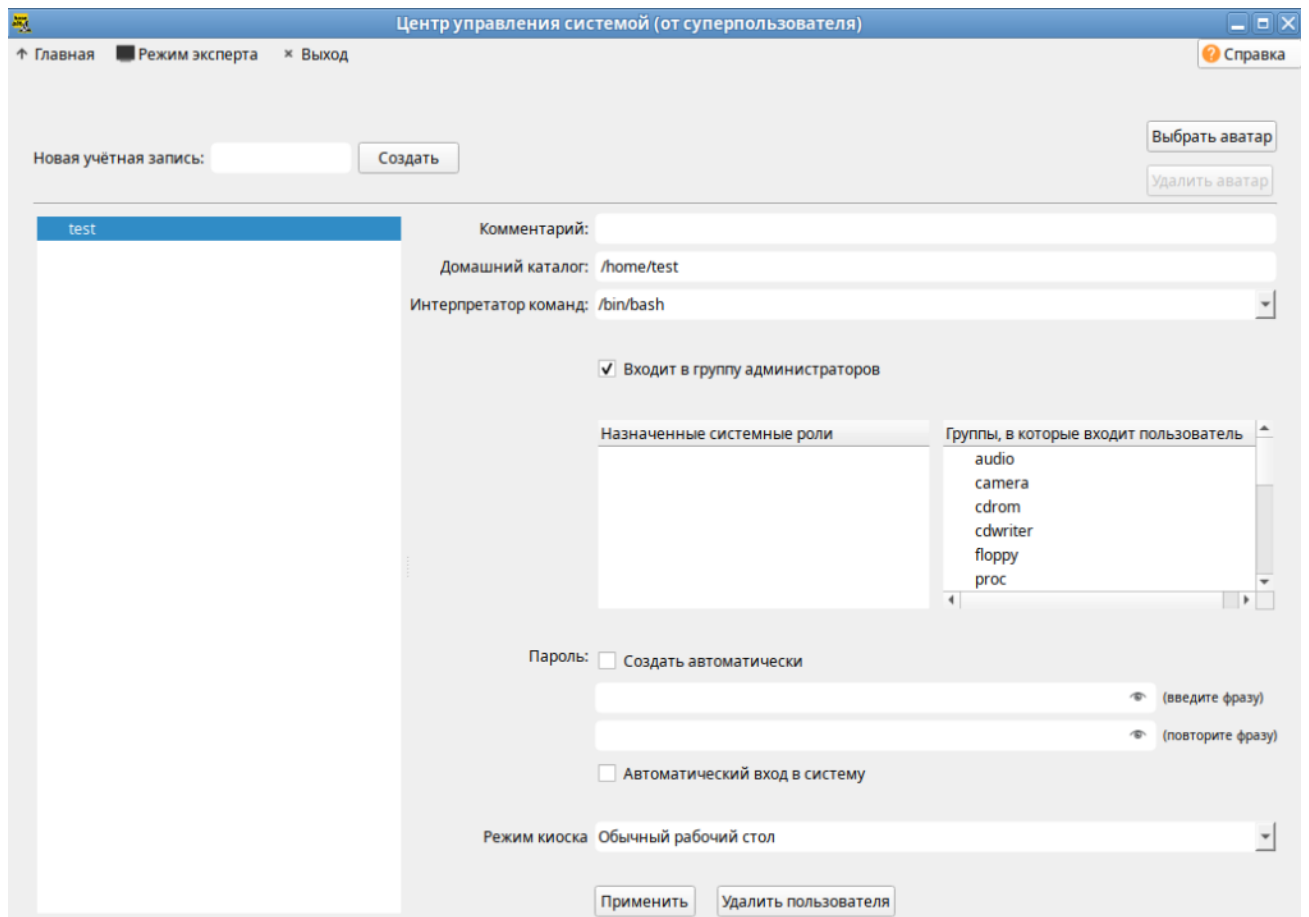


Рис. 8 – Управление локальными пользователями в графическом интерфейсе ЦУС

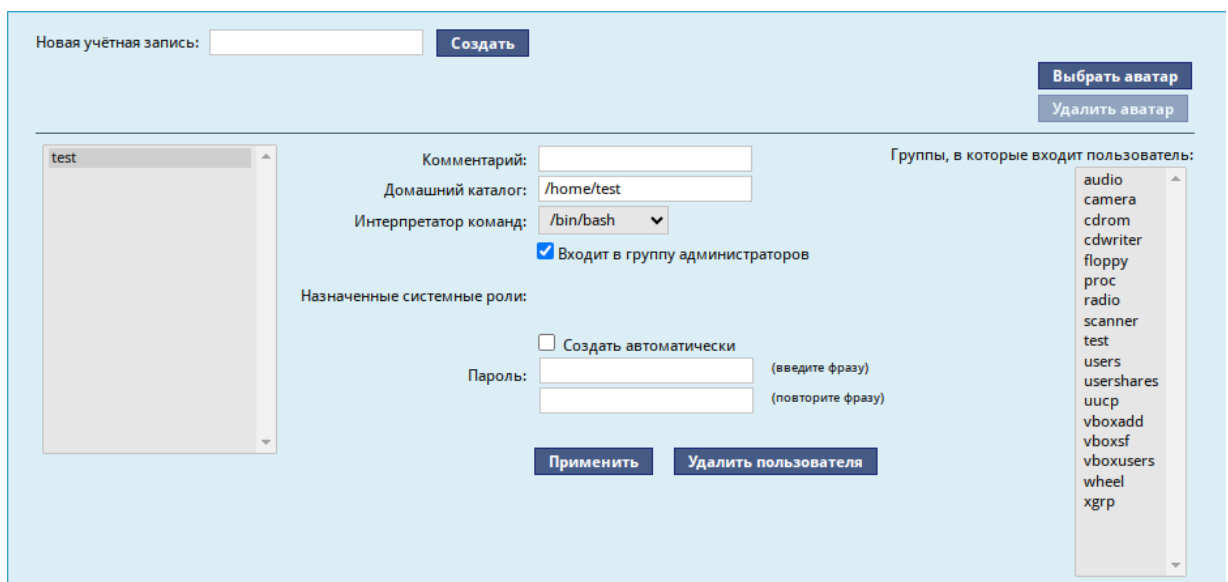


Рис. 9 – Управление локальными пользователями в веб-интерфейсе ЦУС

Для создания новой учетной записи необходимо ввести имя новой учетной записи и нажать кнопку «Создать», после чего имя отобразится в списке слева.

---

⚠ При создании пользователя через ЦУС нужно снимать отметку с пунктов «Входит в группу администраторов» и «Автоматический вход в систему».

---

Для дополнительных настроек необходимо выделить добавленное имя, либо, если необходимо изменить существующую учетную запись, выбрать ее из списка.

Каждой учетной записи можно задать:

- комментарий – произвольный комментарий к учетной записи. Часто здесь указывается реальные имя и фамилия пользователя;
- домашний каталог – каталог пользователя, в котором он будет иметь полные права. В случае регистрации пользователя в консоли работа начинается именно в этом каталоге. Обычно домашний каталог пользователя располагается в `/home/имя_пользователя`, где `имя_пользователя` – это имя учетной записи;
- интерпретатор команд – командная оболочка, запускаемая по умолчанию при регистрации пользователя в текстовой консоли. По умолчанию используется `/bin/bash`;
- входит в группу администраторов – при установленной отметке пользователь имеет возможность получить права администратора (`root`). Например, при помощи команды `su` – (для этого необходимо знать пароль администратора);
- пароль – пароль учетной записи может быть сгенерирован автоматически («Создать автоматически»), либо создан самостоятельно. Во втором случае необходимо ввести его подтверждение.

### 3.2.5. Настройка парольных ограничений

Настройка парольных ограничений производится в файле `/etc/passwdqc.conf`.

`passwdqc.conf` – файл конфигурации `libpasswdqc`. `libpasswdqc` – это простая библиотека для проверки надежности паролей. Помимо проверки простых

паролей, поддерживаются парольные фразы. Файл конфигурации может использоваться для переопределения стандартных настроек `libpasswdqc`.

Файл `passwdqc.conf` состоит из 0 или более строк следующего формата:

опция=значение

Пустые строки и строки, начинающиеся со знака решетка («#»), игнорируются. Символы пробела между опцией и значением не допускаются.

Опции, которые могут быть переданы в модуль (в скобках указаны значения по умолчанию):

`min=N0,N1,N2,N3,N4` (`min=disabled,24,11,8,7`) – минимально допустимая длина пароля.

Используемые типы паролей по классам символов (алфавит, число, спецсимвол, верхний и нижний регистр) определяются следующим образом:

- тип N0 используется для паролей, состоящих из символов только одного класса;
- тип N1 используется для паролей, состоящих из символов двух классов;
- тип N2 используется для парольных фраз, кроме этого требования длины, парольная фраза должна также состоять из достаточного количества слов;
- типы N3 и N4 используются для паролей, состоящих из символов трех и четырех классов, соответственно.

Ключевое слово `disabled` используется для запрета паролей выбранного типа N0 – N4 независимо от их длины.

**Примечание.** Каждое следующее число в настройке «min» должно быть не больше, чем предыдущее.

При расчете количества классов символов, заглавные буквы, используемые в качестве первого символа и цифр, используемых в качестве последнего символа пароля, не учитываются.

`max=N` (`max=40`) – максимально допустимая длина пароля. Эта опция может быть использована для того, чтобы запретить пользователям устанавливать пароли, которые могут быть слишком длинными для некоторых системных служб. Значение 8 обрабатывается особым образом: пароли длиннее 8 символов, не отклоняются, а

обрезаются до 8 символов для проверки надежности (пользователь при этом предупреждается).

`passphrase=N` (`passphrase=3`) – число слов, необходимых для ключевой фразы (значение 0 отключает поддержку парольных фраз).

`match=N` (по умолчанию `match=4`) – длина общей подстроки, необходимой для вывода, что пароль хотя бы частично основан на информации, найденной в символьной строке (значение 0 отключает поиск подстроки). Если найдена слабая подстрока пароль не будет отклонен; вместо этого он будет подвергаться обычным требованиям к прочности при удалении слабой подстроки. Поиск подстроки нечувствителен к регистру и может обнаружить и удалить общую подстроку, написанную в обратном направлении.

`similar=permit|deny` (`similar=deny`) – параметр `similar=permit` разрешает задать новый пароль, если он похож на старый (параметр `similar=deny` – запрещает). Пароли считаются похожими, если есть достаточно длинная общая подстрока, и при этом новый пароль с частично удаленной подстрокой будет слабым.

`random=N[,only]` (`random=42`) – размер случайно сгенерированных парольных фраз в битах (от 26 до 81) или 0, чтобы отключить эту функцию. Любая парольная фраза, которая содержит предложенную случайно сгенерированную строку, будет разрешена вне зависимости от других возможных ограничений.

Значение `only` используется для запрета выбранных пользователем паролей.

`enforce=none|users|everyone` (`enforce=users`) – параметр `enforce=users` задает ограничение задания паролей в `passwd` на пользователей без полномочий `root`. Параметр `enforce=everyone` задает ограничение задания паролей в `passwd` и на пользователей, и на суперпользователя `root`. При значении `none` модуль PAM будет только предупреждать о слабых паролях.

`retry=N` (`retry=3`) – количество запросов нового пароля, если пользователь с первого раза не сможет ввести достаточно надежный пароль и повторить его ввод.



Далее приводится пример задания следующих значений в файле `/etc/passwdqc.conf`:

```
min=8,7,4,4,4
enforce=everyone
```

В указанном примере пользователям, включая суперпользователя `root`, будет невозможно задать пароли:

- типа N0 (символы одного класса) – длиной меньше восьми символов;
- типа N1 (символы двух классов) – длиной меньше семи символов;
- типа N2 (парольные фразы), типа N3 (символы трех классов) и N4 (символы четырех классов) – длиной меньше четырех символов.

### 3.2.5.1. Управление сроком действия пароля

Для управления сроком действия паролей используется команда `chage`. `chage` изменяет количество дней между сменой пароля и датой последнего изменения пароля.

Синтаксис команды: `chage [параметр] логин`

Команда для просмотра справочной информации об утилите:

```
chage --help
```

Основные параметры:

- `-l` – просмотр информации о старении учетной записи пользователя;
- `-d, --lastday LAST_DAY` – изменяется значение `LAST_DAY` на день, когда был изменен пароль последний раз (число дней с 1 января 1970). Дата также может быть указана в формате ГГГГ-ММ-ДД;
- `-m, --mindays MIN_DAYS` – изменяется значение `MIN_DAYS` на минимальное число дней между сменой пароля. Значение 0 в этом поле обозначает, что пользователь может изменять свой пароль, когда угодно;
- `-M, --maxdays MAX_DAYS` – изменяется значение `MAX_DAYS` на максимальное число дней, в течение которых пароль еще действителен. Когда сумма `MAX_DAYS` и `LAST_DAY` меньше, чем текущий день, у

пользователя будет запрошен новый пароль до начала работы в системе. Эта операция может предваряться предупреждением (параметр `-w`);

- `-E, --expiredate EXPIRE_DAYS` – используется для задания даты, с которой учетная запись пользователя станет недоступной.

Параметр `EXPIRE_DAYS` – число дней с 1 января 1970. Дата также может быть указана в формате ГГГГ-ММ-ДД;

- `-I, --inactive INACTIVE` – используется для задания количества дней «неактивности», то есть дней, когда пользователь вообще не входил в систему, после которых его учетная запись будет заблокирована. Параметр `INACTIVE` есть количество «неактивных» дней. Значение 0 отключает этот режим;

- `-W, --warndays WARN_DAYS` – значение `WARN_DAYS` задает число дней до истечения срока действия пароля, начиная с которых пользователю будет выдаваться предупреждение о необходимости смены пароля.

Команда `chage` разрешена только для администратора, за исключением использования ее с параметром `-l` (или `-h`), который позволяет непривилегированным пользователям определить время, когда истекает их пароль или, когда устареет учетная запись.

**Примечание.** Для разрешения использования `chage` непривилегированным пользователем администратору выполнить настройку:  
`control chage public`

### 3.2.6. Настройка неповторяемости пароля

Для настройки неповторяемости паролей используется модуль `pam_pwhistory`, который сохраняет последние пароли каждого пользователя и не позволяет пользователю при смене пароля чередовать один и тот же пароль слишком часто.

Для настройки этого ограничения необходимо изменить файл `/etc/pam.d/system-auth-local-only` таким образом, чтобы он включал модуль `pam_pwhistory` после первого появления строки с паролем (рис. 10):

## ЛКНВ.11102-01 99 01

```
password    required    pam_passwdqc.so config=/etc/passwdqc.conf
password    required    pam_pwhistory.so debug use_authok remember=10 retry=3
```

```
##%PAM-1.0
auth        required    pam_tcb.so shadow fork nullok
account     required    pam_tcb.so shadow fork
password    required    pam_passwdqc.so config=/etc/passwdqc.conf
password    required    pam_pwhistory.so debug use_authok remember=10 retry=3
password    required    pam_tcb.so use_authok shadow fork nullok write_to=tcb
session     required    pam_tcb.so
```

Рис. 10

---

⚠ В данном случае системный каталог станет доступным для записи пользователям группы `pw_users` (создайте эту группу и включите туда пользователей).

---

⚠ База используемых паролей ведется в файле `/etc/security/opasswd`, в который пользователи должны иметь доступ на чтение и запись. При этом они могут читать хэши паролей остальных пользователей. Не рекомендуется использовать на многопользовательских системах.

---

Создайте файл `/etc/security/opasswd` и наделите правами на запись пользователей:

```
install -Dm0660 -gpw_users /dev/null /etc/security/opasswd
chgrp pw_users /etc/security
chmod g+w /etc/security
```

После добавления этой строки в файле `/etc/security/opasswd` будут храниться последние 10 паролей пользователя (содержит хэши паролей всех учетных записей пользователей) и при попытке использования пароля из этого списка будет выведена ошибка:

```
Password has been already used. Choose another.
```

В случае, если необходимо, чтобы проверка выполнялась и для суперпользователя `root`, в настройки нужно добавить параметр `enforce_for_root` (файл `/etc/pam.d/system-auth-local-only`):

```
password    required    pam_pwhistory.so    use_authok enforce_for_root remember=10 retry=3
```

### Параметры:

- `debug` – включить отладку через системный журнал;
- `enforce_for_root` – если эта опция установлена, проверка применяется и для пользователя `root`;
- `remember=N` – последние `N` паролей для каждого пользователя сохраняются в `/etc/security/opasswd` (по умолчанию 10);
- `retry=N` – запросить пароль не более `N` раз, прежде чем вернуться с ошибкой (по умолчанию 1).

### 3.2.7. Модификация уже имеющихся пользовательских записей

Для модификации уже имеющихся пользовательских записей применяется утилита `usermod`.

**Синтаксис:** `usermod [параметры] LOGIN`

#### Опции:

- 1) `-a, --append` – добавит пользователя в группу. Используется только совместно с опцией `-G`;
- 2) `-c, --comment КОММЕНТАРИЙ` – любая текстовая строка. Новое значение комментария;
- 3) `-d, --home ДОМАШНИЙ_КАТАЛОГ` – назначить новый домашний каталог для пользователя. Если введена также опция `-m`, содержимое текущего домашнего каталога пользователя будет перемещено в создаваемый. Если этого каталога не существовало – он будет создан;
- 4) `-e, --expiredate ДАТА_УСТАРЕВАНИЯ` – дата, когда учетная запись пользователя будет заблокирована. Дата задается в формате ГГГГ-ММ-ДД;
- 5) `-f, --inactive ДНЕЙ` – число дней, которые должны пройти после устаревания пароля, чтобы учетная запись заблокировалась навсегда. Если указано значение 0, то учетная запись блокируется сразу после устаревания пароля, а при значении `-1` данная возможность не используется. По умолчанию используется значение `-1`;

- 6) `-g, --gid ГРУППА` – имя или числовой идентификатор новой начальной группы пользователя. Идентификатор группы должен указывать на уже существующую группу;
  - 7) `-G, --groups ГРУППА1 [,ГРУППА2, ... [,ГРУППАН] ]]` – список дополнительных групп, в которых числится пользователь. Перечисление групп осуществляется через запятую, без промежуточных пробелов. На указанные группы действуют те же ограничения, что и для группы, указанной в параметре `-g`. Если пользователь состоит в группе, которая не была перечислена, пользователь будет удален из этой группы, это поведение может быть изменено, если использовать также параметр `-a`;
  - 8) `-l, --login NEW_LOGIN` – имя пользователя будет изменено с `LOGIN` на `NEW_LOGIN`. Никакие другие параметры пользователя не изменятся;
  - 9) `-L, --lock` – заблокировать пароль пользователя;
  - 10) `-h, --help` – показать краткую справку и закончить работу;
  - 11) `-m, --move-home` – переместить содержание домашнего каталога пользователя. Используется только совместно с опцией `-d`;
  - 12) `-p, --password ПАРОЛЬ` – шифрованное значение пароля, которое возвращает функция `crypt`;
  - 13) `-s, --shell ОБОЛОЧКА` – имя регистрационной оболочки пользователя. Если задать пустое значение, то будет использована регистрационная оболочка по умолчанию;
  - 14) `-u, --uid UID` – числовое значение идентификатора пользователя (ID). Оно должно быть уникальным, если не используется параметр `-o`. Значение должно быть неотрицательным. По умолчанию используется наименьшее значение ID большее 999 и большее любого другого значения пользователя. Значения от 0 до 999 обычно зарезервированы для системных учетных записей;
  - 15) `-U, --unlock` – разблокировать пароль пользователя.
- Добавить пользователя `test1` в группы `audio`, `rpm` и `test1`:

```
# usermod -G audio,rpm,test1 test1
```

Такая команда изменит список групп, в которые входит пользователь test1 – теперь входит в группы audio, rpm и test1.

Смена имени пользователя с test1 на test2:

```
# usermod -l test2 test1
```

Временно заблокировать возможность входа в систему пользователю test2:

```
# usermod -L test2
```

Разблокировать пользователя test2:

```
# usermod -U test2
```

Отключить интерактивный вход пользователю test2:

```
# usermod --shell /sbin/nologin test2
```

Изменения вступят в силу только при следующем входе пользователя в систему.

При не интерактивной смене или задании паролей для целой группы пользователей необходимо использовать утилиту `chpasswd`. На стандартный вход ей следует подавать список, каждая строка которого будет выглядеть как `<имя>:<пароль>`.

### 3.2.8. Удаление пользователей

Для удаления пользователей используется программа `userdel`.

Синтаксис:

```
userdel [-r] LOGIN
```

Удалить пользователя test2 из системы:

```
# userdel test2
```

Если будет дополнительно задан параметр `-r`, то будет уничтожен и домашний каталог пользователя. Нельзя удалить пользователя, если в данный момент он еще работает в системе.

### 3.2.9. Оповещение пользователя после успешного входа в ОС о его предыдущем входе

Если зайти в систему пользователем, например, test1 (по соображениям безопасности вводимые символы пароля не показываются на экране):

```
login: test1
Password:
Last login: Fri Mar 22 16:53:20 2024 from localhost on tty2
[test@host ~]$
```

То после успешной авторизации система оповещает пользователя о предыдущем входе в ОС:

```
Last login: Fri Mar 22 16:53:20 2024 from localhost on tty2
```

### 3.2.10. Разрешение (запрет) действий пользователей, разрешенных до идентификации и аутентификации

По умолчанию до аутентификации пользователям доступны следующие действия: возможность выключить/перезагрузить компьютер, поменять раскладку клавиатуры, посмотреть список пользователей.

Для того, чтобы скрыть список пользователей при входе в систему, необходимо в файл `/etc/lightdm/lightdm.conf` в разделе `[Seat:*]` дописать строку:

```
greeter-hide-users=true
```

Для того чтобы отключить возможность выключить/перезагрузить компьютер, создайте файл `/etc/polkit-1/rules.d/lightdm.rules` (см. п. 3.1):

```
# touch /etc/polkit-1/rules.d/lightdm.rules
```

В котором необходимо прописать:

```
polkit.addRule(function(action, subject) {
  if (action.id.indexOf("org.freedesktop.login1.") == 0) {
    if (subject.isInGroup("wheel")) {
      return polkit.Result.AUTH_SELF;
    }
  }
});
```

Если необходимо запретить только перезагрузку, то правило будет таким:

```
polkit.addRule(function(action, subject) {
  if (action.id.indexOf("org.freedesktop.login1.reboot") == 0) {
    if (subject.isInGroup("wheel")) {
      return polkit.Result.YES;
    } else {
      return polkit.Result.AUTH_SELF;
    }
  }
});
```

### 3.3. Средства управления дискреционными ПРД

#### 3.3.1. Команда `chmod`

Команда `chmod` изменяет права доступа указанного файла `FILE` в соответствии с правами доступа, указанными в параметре режим, который может быть представлен как в символьном виде, так и в виде восьмеричного числа, представляющего битовую маску новых прав доступа.

Синтаксис:

```
chmod [ОПЦИЯ]... РЕЖИМ[, РЕЖИМ]... ФАЙЛ...
chmod [ОПЦИЯ]... ВОСЬМЕРИЧНЫЙ-РЕЖИМ ФАЙЛ...
chmod [ОПЦИЯ]... --reference= ЭФАЙЛ ФАЙЛ...
```

Опции:

- 1) `-c`, `--changes` – тоже что и `--verbose`, будет сообщено только о выполненных изменениях;
- 2) `-f`, `--silent`, `--quiet` – не показывать большинство сообщений об ошибках;
- 3) `-v`, `--verbose` – выводить диагностическую информацию для каждого файла;
- 4) `--no-preserve-root` – не обрабатывать «/» специальным образом (по умолчанию);
- 5) `--preserve-root` – не выполнять рекурсивные операции с «/»;
- 6) `--reference=ИФАЙЛ` – использовать режим файла ИФАЙЛ;
- 7) `-R`, `--recursive` – рекурсивно изменять файлы и каталоги.

Формат символьного режима следующий:

```
[ugoa...][[+|=] [разрешения...]...]
```

Здесь разрешения – это ноль или более букв из набора `rwXst` или одна из букв из набора `ugo`.

Каждый аргумент – это список символьных команд изменения прав доступа, разделенный запятыми. Каждая такая команда начинается с нуля или более букв `ugoа`, комбинация которых указывает, чьи права доступа к файлу будут изменены: пользователя, владеющего файлом (`u`), пользователей, входящих в группу, к которой



принадлежит файл (g), остальных пользователей (o) или всех пользователей (a). Буква «а» эквивалентна ugo. Если не задана ни одна буква, то автоматически будет использована буква «а», но биты, установленные в umask, не будут затронуты.

Оператор «+» добавляет выбранные права доступа к уже имеющимся у каждого файла, «-» удаляет эти права. «=» присваивает только эти права каждому указанному файлу.

Буквы rwxXst задают биты доступа для пользователей: «r» – чтение, «w» – запись, «x» – выполнение (или поиск для каталогов), «X» – выполнение/поиск только если это каталог или же файл с уже установленным битом выполнения, «s» – задать ID пользователя и группы при выполнении, «t» – запрет удаления.

Числовой режим состоит не более чем из четырех восьмеричных цифр (от нуля до семи), которые складываются из битовых масок с разрядами «4», «2» и «1». Любые пропущенные разряды дополняются лидирующими нулями:

- первый разряд выбирает установку идентификатора пользователя (setuid) (4) или идентификатора группы (setgid) (2) или sticky-бита (1);
- второй разряд выбирает права доступа для пользователя, владеющего данным файлом: чтение (4), запись (2) и исполнение (1);
- третий разряд выбирает права доступа для пользователей, входящих в данную группу, с тем же смыслом, что и у второго разряда;
- четвертый разряд выбирает права доступа для остальных пользователей (не входящих в данную группу), опять с тем же смыслом.

**Примеры :**

1. Чтобы установить права, позволяющие владельцу читать и писать в файл, а членам группы и прочим пользователям только читать, надо сложить 0400, 0200, 0040 и 0004. Таким образом, команду можно записать двумя способами:

```
chmod 644 f1
chmod u=rw,go=r f1
```

2. Позволить всем выполнять файл f2:

```
chmod +x f2
```

3. Запретить удаление файла f3:

```
chmod +t f3
```

4. Дать всем права на чтение, запись и выполнение, а также на переустановку идентификатора группы при выполнении файла f4:

```
chmod =rwx,g+s f4
chmod 2777 f4
```

### 3.3.2. Команда chown

Команда `chown` изменяет владельца и (или) группу, владеющую каждым из указанных файлов, согласно заданным аргументам, которые интерпретируются в последовательном порядке.

Синтаксис:

```
chown [КЛЮЧ]...[ВЛАДЕЛЕЦ] [: [ГРУППА]] ФАЙЛ ...
chown [ОПЦИЯ]... --reference=ИФАЙЛ ФАЙЛ...
```

Опции:

- 1) `-c`, `--changes` – тоже что и `--verbose`, будет сообщено только о выполненных изменениях;
- 2) `-f`, `--silent`, `--quiet` – не показывать большинство сообщений об ошибках;
- 3) `-v`, `--verbose` – выводить диагностическую информацию для каждого файла;
- 4) `--no-preserve-root` – не обрабатывать «/» специальным образом (по умолчанию);
- 5) `--preserve-root` – не выполнять рекурсивные операции с «/»;
- 6) `--reference=ИФАЙЛ` – использовать владельца и группу файла ИФАЙЛ;
- 7) `-R`, `--recursive` – рекурсивно изменять файлы и каталоги.

Изменить владельца может только владелец файла или суперпользователь.

В случае, если задано только имя пользователя (или его числовой идентификатор), то данный пользователь становится владельцем каждого из указанных файлов, а группа этих файлов не изменяется. При этом, если за именем пользователя через двоеточие следует имя группы (или числовой идентификатор

группы) без пробелов между ними, то изменяется также и группа файлов. Также, если за именем пользователя следует двоеточие или точка, но группа не задана, то данный пользователь становится владельцем указанных файлов, а группа указанных файлов изменяется на основную группу пользователя. Также, если опущено имя пользователя, а двоеточие или точка вместе с группой заданы, то будет изменена только группа указанных файлов; в этом случае `chown` выполняет ту же функцию, что и `chgrp`.

### Примеры:

1. Поменять владельца каталог `/u` на пользователя `test`:

```
chown test /u
```

2. Поменять владельца и группу каталога `/u`:

```
chown test:staff /u
```

3. Поменять владельца каталога `/u` и вложенных файлов на `test`:

```
chown -hR test /u
```

### 3.3.3. Команда `chgrp`

Команда `chgrp` изменяет группу, владеющую каждым из указанных файлов `FILE`, на группу `GROUP`, которая может быть задана именем группы или числовым идентификатором группы.

#### Синтаксис:

```
chgrp [ОПЦИЯ]... ГРУППА ФАЙЛ ...
chgrp [ОПЦИЯ]... --reference=ИФАЙЛ ФАЙЛ...
```

#### Опции:

- 1) `-c`, `--changes` – то же, что и `--verbose` – будет сообщено только о выполненных изменениях;
- 2) `-f`, `--silent`, `--quiet` – не показывать большинство сообщений об ошибках;
- 3) `-v`, `--verbose` – выводить диагностическую информацию для каждого файла;
- 4) `--no-preserve-root` – не обрабатывать «/» специальным образом (по умолчанию);

- 5) `--preserve-root` – не выполнять рекурсивные операции с «/»;
- 6) `--reference=ИФАЙЛ` – использовать владельца и группу файла ИФАЙЛ;
- 7) `-R, --recursive` – рекурсивно изменять файлы и каталоги.

### 3.3.4. Команда `umask`

Команда `umask` задает маску режима создания файла в текущей среде командного интерпретатора равной значению, задаваемому операндом режим. Эта маска влияет на начальное значение битов прав доступа всех создаваемых далее файлов.

Синтаксис:

```
umask [-p] [-S] [режим]
```

Пользовательской маске режима создания файлов присваивается указанное восьмеричное значение. Три восьмеричные цифры соответствуют правам на чтение/запись/выполнение для владельца, членов группы и прочих пользователей соответственно. Значение каждой заданной в маске цифры вычитается из соответствующей «цифры», определенной системой при создании файла. Например, `umask 022` удаляет права на запись для членов группы и прочих пользователей (у файлов, создававшихся с режимом `777`, он оказывается равным `755`; а режим `666` преобразуется в `644`).

Если маска не указана, выдается ее текущее значение.

Команда `umask` распознается и выполняется командным интерпретатором `bash`.

### 3.3.5. Команда `chattr`

Команда `chattr` – изменяет атрибуты файлов файловой системы `ext2fs`.

Синтаксис:

```
chattr [ -Rvf ] [ +=aAcCdDeijsSTtu ] [ -v версия ] файлы...
```

Оператор '+' означает добавление выбранных атрибутов к существующим атрибутам; '-' означает их снятие; '=' означает определение только этих указанных атрибутов для файлов.

Символы 'ASacDdijsTtu' указывают на новые атрибуты для файлов: не обновлять время последнего доступа (atime) к файлу (A), синхронное обновление (s), только добавление к файлу (a), сжатый (c), синхронное обновление каталогов (D), не архивировать (d), неизменяемый (i), журналирование данных (j), безопасное удаление (s), вершина иерархии каталогов (T), нет tail-merging (t), неудаляемый (u).

Опции:

1) -R – рекурсивно изменять атрибуты каталогов и их содержимого.

Символические ссылки игнорируются;

2) -v – выводит расширенную информацию и версию программы;

3) -f – подавлять сообщения об ошибках;

4) -v версия – установить номер версии/генерации файла.

При изменении файла с атрибутом (A) время последнего доступа к нему не изменяется.

Файл с атрибутом (a) можно открыть для записи только в режиме добавления. Только суперпользователь или процесс с возможностью CAP\_LINUX\_IMMUTABLE может устанавливать и снимать этот атрибут.

Файл с атрибутом (c) автоматически сжимается на диске ядром. Чтение из такого файла возвращает несжатые данные. При записи в такой файл данные перед записью на диск сжимаются.

При изменении каталога с атрибутом (D) изменения синхронно записываются на диск. Это эквивалентно опции монтирования `dirsync` примененной к подмножеству файлов.

Файл с атрибутом (d) не является кандидатом на архивирование при использовании команды `dump`.

Атрибут (E) используется экспериментальными сжимающими патчами для того, чтобы показать, что сжатый файл содержит ошибки сжатия. Он не может быть установлен или сброшен с помощью `chattr`, хотя его можно просмотреть с помощью `lsattr`.

Атрибут (I) используется кодом `htree` для того, чтобы показать, что каталог индексируется с использованием хэширующих деревьев. Он не может быть установлен или сброшен с помощью `chattr`, хотя его можно просмотреть с помощью `lsattr`.

Файл с атрибутом (i) не может быть изменен: он не может быть удален или переименован, к этому файлу не могут быть созданы ссылки, и никакие данные не могут быть записаны в этот файл. Только суперпользователь или процесс с возможностью `CAP_LINUX_IMMUTABLE` может устанавливать и снимать этот атрибут.

При записи в файл с атрибутом (j) все данные, записываемые в такой файл, записываются в журнале `ext3`, прежде чем они будут записаны непосредственно в файл, если файловая система смонтирована с опциями `data=ordered` или `data=writeback`. Если файловая система смонтирована с опцией `data=journalled`, то все данные журналируются, и этот атрибут не дает никакого эффекта. Только суперпользователь или процесс с возможностью `CAP_SYS_RESOURCE` может устанавливать или снимать этот атрибут.

Когда удаляется файл с атрибутом (s), все его блоки заполняются нулями.

При изменении файла с атрибутом (S) все изменения синхронно записываются на диск; это эквивалентно опции монтирования `'sync'`, примененной к подмножеству файлов.

Каталог с атрибутом (T) будет поднят на вершину иерархии каталогов для целей `Orlov block allocator`.

У файла с атрибутом (t) в конце не будет `partial block fragment`, соединенного с другими файлами (для тех файловых систем, которые поддерживают `tail-merging`). Это необходимо для приложений, таких как `LILO`, которые читают файловую систему напрямую, и которые не понимают `tail-merged` файлы.

При удалении файла с атрибутом (u) его содержимое сохраняется. Это позволяет пользователю восстановить файл.

Атрибут (x) используется экспериментальными сжимающими патчами, чтобы показать, что исходное содержимое сжатых файлов доступно напрямую. В данное

время он не может быть установлен или переустановлен с помощью `chattr(1)`, но может быть показан с помощью `lsattr`.

Атрибут (z) используется экспериментальными сжимающими патчами, чтобы показать, что сжатый файл не сохранен. Он не может быть установлен или переустановлен с помощью `chattr`, но может быть показан с помощью `lsattr`.

### 3.3.6. Команда `lsattr`

Команда `lsattr` – выдает список атрибутов файлов на Linux `ext2fs`.

Синтаксис:

```
lsattr [ -RVadv ] [ файлы... ]
```

Опции:

- 1) `-R` – рекурсивно изменять атрибуты каталогов и их содержимого. Символические ссылки игнорируются;
- 2) `-v` – выводит расширенную информацию и версию программы;
- 3) `-a` – просматривает все файлы в каталоге, включая те, имена которых начинаются с '!';
- 4) `-d` – отображает каталоги так же, как и файлы, вместо того чтобы просматривать их содержимое;
- 5) `-v` – просматривает номера версий/генераций файлов.

### 3.3.7. Команда `mksock`

Команда `mksock` создает сокет домена UNIX (IPC-сокет) – сокет межпроцессного взаимодействия.

Команда `mksock` обладает следующим синтаксисом:

```
mksock [ОПЦИИ] ИМЯ...
```

Опции:

- 1) `-m, --mode=РЕЖИМ` – назначить права доступа на сокет.

### 3.3.8. Команда `mkfifo`

Команда `mkfifo` создает именованный канал.

Синтаксис:

```
mkfifo [OPTION]...NAME...
```

Опции:

- 1) `-m, --mode=РЕЖИМ` – назначить права доступа на сокет.

### 3.3.9. Команда `getfacl`

Команда `getfacl` выводит для каждого файла его характеристики: имя файла, владельца, группу-владельца и ACL. В случае, если каталог имеет ACL по умолчанию, то `getfacl` выводит также ACL по умолчанию. Файлы не могут иметь ACL по умолчанию.

Синтаксис:

```
getfacl [-dRLP] файл
```

Опции:

- 1) `--access` – вывести только ACL файла;
- 2) `-d, --default` – вывести только ACL по умолчанию;
- 3) `--omit-header` – не показывать заголовков (имя файла);
- 4) `--all-effective` – показывать все эффективные права;
- 5) `--no-effective` – не показывать эффективные права;
- 6) `--skip-base` – пропускать файлы, имеющие только основные записи;
- 7) `-R, --recursive` – для подкаталогов рекурсивно;
- 8) `-L, --logical` – следовать по символическим ссылкам, даже если они не указаны в командной строке;
- 9) `-P, --physical` – не следовать по символическим ссылкам, даже если они указаны в командной строке;
- 10) `--tabular` – использовать табулированный формат вывода;
- 11) `--numeric` – показывать числовые значения пользователя/группы;
- 12) `--absolute-names` – не удалять ведущие «/» из пути файла.

Формат вывода:

```
1: # file: somedir/
2: # owner: lisa
3: # group: staff
4: user::rwx
5: user:joe:rwx           #effective:r-x
6: group::rwx           #effective:r-x
7: group:cool:r-x
8: mask:r-x
```



```

9: other:r-x
10: default:user::rwx
11: default:user:joe:rwx #effective:r-x
12: default:group::r-x
13: default:mask:r-x
14: default:oter:---

```

Строки четыре, шесть и девять относятся к традиционным битам прав доступа к файлу, соответственно, для владельца, группы-владельца и всех остальных. Эти три элемента являются базовыми. Строки пять и семь являются элементами для отдельных пользователя и группы. Строка восемь – маска эффективных прав. Этот элемент ограничивает эффективные права, предоставляемые всем группам и отдельным пользователям. Маска не влияет на права для владельца файла и всех других. Строки с десятой по четырнадцатую показывают ACL по умолчанию, ассоциированный с данным каталогом.

### 3.3.10. Команда `setfacl`

Команда `setfacl` изменяет ACL к файлам или каталогам. В командной строке за последовательностью команд идет последовательность файлов (за которой, в свою очередь, также может идти последовательность команд и так далее).

Синтаксис:

```
setfacl [-bkndRLP] {-m| -M| -x| -X ... } файл ...
```

Опции:

- 1) `-m, --modify=acl` – изменить текущий ACL для файла;
- 2) `-M, --modify-file=file` – прочитать записи ACL для модификации из файла;
- 3) `-x, --remove=acl` – удалить записи из ACL файла;
- 4) `-X, --remove-file=file` – прочитать записи ACL для удаления из файла;
- 5) `-b, --remove-all` – удалить все разрешенные записи ACL;
- 6) `-k, --remove-default` – удалить ACL по умолчанию;
- 7) `--set=acl` – установить ACL для файла, заменив текущий ACL;
- 8) `--set-file=file` – прочитать записи ACL для установления из файла;
- 9) `--mask` – пересчитать маску эффективных прав;

- 10) `-n`, `--no-mask` – не пересчитывать маску эффективных прав, обычно `setfacl` пересчитывает маску (кроме случая явного задания маски) для того, чтобы включить ее в максимальный набор прав доступа элементов, на которые воздействует маска (для всех групп и отдельных пользователей);
- 11) `-d`, `--default` – применить ACL по умолчанию;
- 12) `-R`, `--recursive` – для подкаталогов рекурсивно;
- 13) `--restore=file` – восстановить резервную копию прав доступа, созданную командой `getfacl -R` или ей подобной. Все права доступа дерева каталогов восстанавливаются, используя этот механизм. В случае, если вводимые данные содержат элементы для владельца или группы-владельца и команда `setfacl` выполняется пользователем с именем `root`, то владелец и группа-владелец всех файлов также восстанавливаются. Эта опция не может использоваться совместно с другими опциями, за исключением опции `--test`;
- 14) `--test` – режим тестирования (ACL не изменяются).

При использовании опций `--set`, `-m` и `-x` должны быть перечислены записи ACL в командной строке. Элементы ACL разделяются одинарными кавычками.

При чтении ACL из файла при помощи опций `--set-file`, `-M` и `-X` команда `setfacl` принимает множество элементов в формате вывода команды `getfacl`. В строке обычно содержится не больше одного элемента ACL.

### 3.3.11. Элементы ACL

Команда `setfacl` использует следующие форматы элементов ACL:

- 1) права доступа отдельного пользователя (если не задан UID, то права доступа владельца файла):  

```
[d[efault]:] [u[ser]:]UID [:[+|^]perms]
```
- 2) права доступа отдельной группы (если не задан GID, то права доступа группы-владельца):

```
[d[efault]:] g[roup]:GID [:[+|^]perms]
```

3) маска эффективных прав:

```
[d[efault]:] m[ask]:[+|^] perms
```

4) права доступа всех остальных:

```
[d[efault]:] o[ther]:[+|^] perms
```

Элемент ACL является абсолютным, если он содержит поле `perms` и является относительным, если он включает один из модификаторов: «+» или «^». Абсолютные элементы могут использоваться в операциях установки или модификации ACL. Относительные элементы могут использоваться только в операции модификации ACL. Права доступа для отдельных пользователей, группы, не содержащие никаких полей после значений UID, GID (поле `perms` при этом отсутствует), используются только для удаления элементов.

Значения UID и GID задаются именем или числом. Поле `perms` может быть представлено комбинацией символов `r`, `w`, `x`, `-` или цифр (0 – 7).

### 3.3.12. Автоматически созданные права доступа

Изначально файлы и каталоги содержат только три базовых элемента ACL: для владельца, группы-владельца и всех остальных пользователей. Существует ряд правил, которые следует учитывать при установке прав доступа:

- 1) не могут быть удалены сразу три базовых элемента, должен присутствовать хотя бы один;
- 2) если ACL содержит права доступа для отдельного пользователя или группы, то ACL также должен содержать маску эффективных прав;
- 3) если ACL содержит какие-либо элементы ACL по умолчанию, то в последнем должны также присутствовать три базовых элемента (т. е. права доступа по умолчанию для владельца, группы-владельца и всех остальных);
- 4) если ACL по умолчанию содержит права доступа для всех отдельных пользователей или групп, то в ACL также должна присутствовать маска эффективных прав.

Для того чтобы помочь пользователю выполнять эти правила, команда `setfacl` создает права доступа, используя уже существующие, согласно следующим условиям:

- 1) если права доступа для отдельного пользователя или группы добавлены в ACL, а маски прав не существует, то создается маска с правами доступа группы-владельца;
- 2) если создан элемент ACL по умолчанию, а трех базовых элементов не было, тогда делается их копия и они добавляются в ACL по умолчанию;
- 3) если ACL по умолчанию содержит какие-либо права доступа для конкретного пользователя или группы и не содержит маску прав доступа по умолчанию, то при создании эта маска будет иметь те же права, что и группа по умолчанию.

### 3.4. Средства управления и очистки памяти

#### 3.4.1. Управление механизмом очистки оперативной памяти

При первоначальном назначении или при перераспределении внешней памяти КСЗ затрудняет субъекту доступ к остаточной информации. При перераспределении оперативной памяти КСЗ осуществляет ее очистку.

Внешняя память, используемая ОС, располагается на отдельном разделе диска, представленном в файловой системе специальным файлом, доступ к которому непосредственно из программы контролируется правами доступа Linux.

По умолчанию доступ к разделам диска имеет только доверенный субъект или член группы «disk». При любом выделении процессу оперативной памяти она предварительно очищается.

ОС Альт СП содержит механизм очистки освобождаемых областей оперативной памяти. Этот механизм не включен по умолчанию по соображениям производительности и в связи с тем, что защита памяти, осуществляемая ядром по умолчанию, для многих моделей угроз является достаточной.

Очистка освобождаемых областей оперативной памяти происходит в процессе перевода ядром ОС каждой страницы памяти в разряд «неиспользуемых» (free).

Это означает, в числе прочего, что ни одна страница из числа неиспользуемых не будет содержать данных, которые там размещала ОС или приложения в процессе работы системы.

В работающей системе информация об очистке освобождаемых областей памяти доступна в каталоге виртуальной служебной файловой системы `/sys/kernel/mm/sanitize_memory/`. Здесь файл `level` содержит значение параметра `smem`, а файл `count` – количество памяти в байтах, обработанных средствами очистки. Если ядро загружено без поддержки очистки освобождаемых областей памяти, указанный каталог не создается.

**Примечание.** Очистка освобождаемых областей памяти не распространяется на `swap`. Если требуется использование функций очистки памяти, `swap` не должен использоваться (не должен быть подключен или должен вообще отсутствовать).

Для удаления данных в памяти используется параметр ядра `smem` (устанавливается при установке системы).

Для изменения параметров механизма очистки освобождаемой оперативной памяти необходимо в файле `/boot/boot.conf` в значении переменной `CMDLINE` изменить параметр `smem`.

**Примечание.** Для применения настройки необходимо перезагрузить систему.

Далее представлен пример, в котором в первой строке одно значение переменной, во второй другое значение:

```
' quiet=1 panic=30 splash smem=1 '  
' quiet=1 panic=30 splash smem=2 '
```

Доступны следующие значения переменной `smem`, предоставляющие настройки очистки памяти:

- 2 – заполнение памяти случайным шаблоном;
- 1 – зануление оперативной памяти.

Для отключения `swap` необходимо закомментировать (или удалить) в файле `/etc/fstab` строку, содержащую слово `swap`. Строка может выглядеть следующим образом:

```
UUID=[...]      swap      swap      defaults      0      0
```

После внесения изменений в файлы `/boot/boot.conf` и `/etc/fstab` необходимо перезагрузить ПЭВМ.

### 3.4.2. Очистка дисковой памяти

Для обеспечения очистки дискового пространства при выходе пользователя из системы добавьте в файл `/etc/lightdm/lightdm.conf` в секцию `[Seat:*]` строку:

```
session-cleanup-script=sfill -llzf /home
```

---

⚠ Внимание, при большом объеме дисков это значительно замедлит возможность входа нового пользователя.

---

Возможной альтернативой является запуск команды `sfill` при помощи `cron`, например, для Сервера, добавить задачу: `* * * * * sfill -llzf /home`

Команда `sfill` предназначена для удаления в безопасном режиме данных, расположенных на доступном дисковом пространстве носителей (очистка от следов удаленных данных свободного места). Эти данные впоследствии не могут быть восстановлены. Безопасный процесс удаления данных `sfill` выглядит следующим образом:

- 1 проход с `0xff`;
- 5 случайных проходов. Используется `/dev/urandom` для безопасной RNG, если таковые имеются;
- 27 проходов со специальными значениями;
- 5 случайных проходов. Используется `/dev/urandom` для безопасной RNG, если таковые имеются;
- после этого, генерируется так много временных файлов, как это возможно, для того, чтобы очистить пространство индексных дескрипторов. После того, как временные файлы больше не могут быть созданы, они будут удалены и `sfill` будет завершен.

Синтаксис команды:

```
sfill [-f] [-i] [-I] [-l] [-L] [-v] [-z] directory/mountpoint
```

Опции команды:

- 1) `-f` – быстрый (и небезопасный) режим: не используется `/dev/urandom`, нет режима синхронизации;
- 2) `-i` – очистить только пространство индексных дескрипторов, не стирать места на диске;
- 3) `-I` – очистить только место на диске, не очищать пространство индексных дескрипторов;
- 4) `-l` – уменьшает безопасность. Выполняется только два прохода: проход с `0xff` и окончательный проход со случайными значениями;
- 5) `-L` – `-L` – второй раз уменьшает безопасность еще больше: выполняется проход только с записью случайных значений;
- 6) `-v` – подробный режим;
- 7) `-z` – делает последний проход с нулями вместо случайных данных;
- 8) `directory/mountpoint` – местоположение файла, созданного в вашей файловой системе. Он должен находиться в разделе, который необходимо перезаписать.

### 3.5. Средства контроля ввода-вывода

#### 3.5.1. Средства взаимодействия с устройствами ввода-вывода

В ОС Альт СП доступ к физическому устройству осуществляется с помощью специального файла устройства. При выполнении с файлом устройства операций открытия, чтения или записи осуществляется обмен данными с физическим устройством. Файлы устройств хранятся в каталоге `/dev`.

В ОС Альт СП используются стандартные имена устройств:

- `ttyN` – консоль;
- `mouse` – манипулятор типа «мышь»;
- `audio` – звуковая карта;
- `modem` – модем;

- ttySN – последовательный порт;
- lpN – параллельный порт;
- cuaN – могут обозначать последовательные порты;
- sdxN – накопитель на жестких магнитных дисках;
- fd0 – первый дисковод для гибких дисков;
- stN – стример с интерфейсом SCSI;
- nrtfN – запоминающее устройство на принципе магнитной записи на ленточном носителе, с последовательным доступом к данным с интерфейсом FDC;
- mdN – массив RAID;
- ethN – сетевая плата;
- null – пустое устройство.

Примечание. N – номер устройства (например, tty1 – первая консоль).

### 3.5.2. Средства контроля использования интерфейсов ввода (вывода) информации на машинные носители данных

В ОС Альт СП поддерживаются ограничение или запрет использования внешних носителей при помощи правил API библиотеки polkit и (или) udev.

### 3.5.3. Ограничения при помощи правил API библиотеки polkit

Устройства хранения информации делятся на системные, которые не считаются извлекаемыми, и несистемные, к которым относятся подключаемые накопители типа USB, Flash медиа и оптические приводы. Для каждой из групп устройств, системных и несистемных (извлекаемых, removable), для одной операции может понадобиться настройка двух polkit actions – по одному на каждую группу устройств.

Чтобы узнать, является ли устройство системным и распространяется ли на него действие `org.freedesktop.udisks2.filesystem-mount-system`, необходимо выполнить команду, которая выведет все подключенные накопители:

```
udisksctl status
```



Далее нужно выполнить команду с именем устройства (например, /dev/sdb). Статус true для HintSystem, в выводе команды говорит, что это системное устройство:

```
udisksctl info -b /dev/sdb|grep ' Device:||HintSystem'
Device:                               /dev/sdb
HintSystem:                           true
```

Для несистемных устройств, на которые распространяется действие org.freedesktop.udisks2.filesystem-mount-other-seat, для HintSystem статус будет false.

За разрешения ввода-вывода на машинные носители данных отвечают:

- org.freedesktop.udisks2.filesystem-mount-system – разрешение на монтирование файловых систем системных устройств;
- org.freedesktop.udisks2.filesystem-mount-other-seat – разрешение на монтирование файловых систем с устройств, подключенных в другое место;
- org.freedesktop.udisks2.eject-media-other-seat – разрешение на извлечение лотка оптического привода;
- org.freedesktop.udisks2.power-off-drive-other-seat – разрешение на извлечение USB-накопителя.

Далее приводится пример создания правила, разрешающего пользователю из системной группы xgrp выполнять монтирование и извлечение устройств – с запросом пароля. При этом для остальных пользователей действия по монтированию и извлечению будут запрещены.

Для разрешения пользователю монтирования и извлечения устройств необходимо выполнить следующие действия (все действия выполняются от администратора):

1) создать файл правила 99-udisk2\_mount.rules:

```
# touch /etc/polkit-1/rules.d/99-udisk2_mount.rules
```

2) наполнить 99-udisk2\_mount.rules следующим содержанием:

```
polkit.addRule(function(action, subject) {
  if (action.id=="org.freedesktop.udisks2.filesystem-mount-system") {
```

```

    if (subject.isInGroup("xgrp")) {
        return polkit.Result.AUTH_ADMIN;
    } else {
        return polkit.Result.NO;
    }
};
if(action.id=="org.freedesktop.udisks2.filesystem-mount-other-seat") {
    if (subject.isInGroup("xgrp")) {
        return polkit.Result.AUTH_ADMIN;
    } else {
        return polkit.Result.NO;
    }
};
if(action.id=="org.freedesktop.udisks2.eject-media-other-seat") {
    if (subject.isInGroup("xgrp")) {
        return polkit.Result.AUTH_ADMIN;
    } else {
        return polkit.Result.NO;
    }
};
if(action.id=="org.freedesktop.udisks2.power-off-drive-other-seat") {
    if (subject.isInGroup("xgrp")) {
        return polkit.Result.AUTH_ADMIN;
    } else {
        return polkit.Result.NO;
    }
}
});

```

3) создать системную группу xgrp (если ее еще нет):

```
# groupadd -r xgrp
```

4) добавить пользователя в группу xgrp:

```
gpasswd -a имя_пользователя xgrp
```

### 3.5.4. Контроль при помощи правил polkit

Правила polkit также позволяют формировать правила для добавления записей в системном журнале. Например, при подключении съемного устройства можно записывать в журнал, какое устройство было подключено и каким пользователем.

Для этого необходимо выполнить следующие действия:

1) создать файл правила 70-udisk2\_mount.rules:

```
touch /etc/polkit-1/rules.d/70-udisk2_mount.rules
```

2) наполнить его содержимым:

```

polkit.addRule(function(action, subject){
    if(action.id=="org.freedesktop.udisks2.filesystem-mount-system"){
        polkit.log("mount action=" + action);
        polkit.log("mount subject=" + subject);
    }
});

```

```

    return polkit.Result.YES;
};
if(action.id=="org.freedesktop.udisks2.filesystem-mount-other-seat"){
    polkit.log("mount action=" + action);
    polkit.log("mount subject=" + subject);
    return polkit.Result.YES;
};
});

```

Теперь при монтировании USB-диска в /var/log/secure появятся записи:

```

# tail -2 /var/log/secure
Nov 22 12:57:12 host-15 polkitd[9879]: /etc/polkit-1/rules.d/99-
udisk2_mount.rules:4: action [Action
id='org.freedesktop.udisks2.filesystem-mount-system'
id.version='FAT32' id.usage='filesystem'
drive.serial='11101094E6BA1A00A4A5200A' id.label='ALT p8
xfce/e2k' partition.flags='0x00000000'
polkit.gettext_domain='udisks2' drive='UFD 2.0 Silicon-Power4G
(/dev/sdb1)' partition.number='1' id.uuid='F076-C625'
drive.vendor='UFD 2.0' device='/dev/sdb1' id.type='vfat'
partition.type='0x0b' polkit.message='Authentication is required
to mount $(drive)' drive.revision='PMAP' drive.model='Silicon-
Power4G']
Nov 22 12:57:13 host-15 polkitd[9879]: /etc/polkit-1/rules.d/99-
udisk2_mount.rules:5: subject [Subject pid=4673 user='test'
groups=
uucp,proc,cdrom,floppy,cdwriter,audio,radio,users,scanner,xgrp,
vmusers,audit_group,audit1,test, seat='seat0' session='5'
local=true active=true]

```

Таким образом, в системном журнале зарегистрировано, что USB-диск с серийным номером 11101094E6BA1A00A4A5200A был подключен пользователем test.

### 3.5.5. Ограничения при помощи правил Udev

Udev – сервис, который подхватывает и конфигурирует внешние устройства, получая уведомления от ядра ОС.

**Примечание.** При разграничении доступа к устройствам типа видеокарт, либо сетевых карт, названный метод не используется.

Для решения задачи разграничения доступа к устройствам на основе генерации правил менеджера устройств udev в ОС реализованы:

- средства разграничения доступа к устройствам на основе правил udev;
- средства регистрации устройств.

Средства разграничения доступа к устройствам на основе генерации правил `udev` обеспечивают дискреционное разграничение доступа пользователей к подключаемым, в первую очередь, через интерфейс USB, устройствам (сканерам, съемным накопителям, видеокамерам).

Средства регистрации устройств обеспечивают учет подключаемых устройств и съемных носителей в системе, установку дискреционных атрибутов доступа пользователей к устройствам и создание дополнительных правил доступа к устройству (например, ограничение на подключение устройства только к определенному USB-порту).

`Udev` гибко настраивается под оборудование и задачи с помощью специальных правил.

Правила `Udev` хранятся в папке `/etc/udev/rules.d`. Файл правил обязательно должен иметь расширение `.rules`.

Пример правила для съемного USB-носителя:

```
ENV{ID_SERIAL}=="JetFlash_TS256MJF120_OYLIXNA6", OWNER="user",  
GROUP="users"
```

В приведенном примере для съемного USB-носителя с серийным номером `JetFlash_TS256MJF120_OYLIXNA6` разрешено его использование владельцу устройства: пользователю `user` и пользователям, входящим в группу `users`.

Типовое правило `Udev` состоит из нескольких пар «ключ – значение», разделенных запятой.

Одни ключи используются для проверки соответствия устройства определенному правилу, в таких ключах используется знак «`==`» для разделения пары. Следующий пример отражает применение правила только для случая, если значения ключа `SUBSYSTEM` для этого устройства равно `block`:

```
SUBSYSTEM=="block"
```

Другие ключи используются для указания действия, если все условия соответствия выполняются. Для разделения пар в таких ключах используется знак равно «=». Например, в случае с `NAME="mydisk"` правило будет выглядеть следующим образом:

```
SUBSYSTEM=="block", ATTR(size)=="1343153213", NAME="mydisk"
```

Это правило выполнится только если указанные выше условия соответствия выполняются и размер устройства 1343153213 байт.

Для правил Udev существуют следующие ключи соответствия:

- `SUBSYSTEM` – подсистема устройства;
- `KERNEL` – имя, выдаваемое устройству ядром;
- `DRIVER` – драйвер обслуживающий устройство;
- `ATTR` – `sysfs` атрибут устройства;
- `SUBSYSTEMS` – подсистема родительского устройства.

Для действий используются ключи:

- `NAME` – установить имя файла устройства;
- `SYMLINK` – альтернативное имя устройства;
- `RUN` – выполнить скрипт при подключении устройства;
- `GROUP` – группа, у которой есть доступ к файлу;
- `OWNER` – владелец файла устройства;
- `MODE` – маска прав доступа.

Ключ `ATTR` позволяет получить информацию об устройстве. Посмотреть все возможные Udev параметры для устройства можно с помощью команды `udevadm`.

Например, для диска `/dev/sda` команда просмотра параметров будет выглядеть следующим образом:

```
# udevadm info -a -n sda1
```

Для создания файла с правилами нужно выполнить следующую команду:

```
touch /etc/udev/rules.d/usb.rules
```

Правило отключения ручного монтирования, для всех пользователей не из группы `plugdev`, которое необходимо добавить в файл `usb.rules`, будет выглядеть следующим образом:

```
BUS=="usb", SUBSYSTEM=="block", KERNEL=="sd*", ACTION=="add",
GROUP="plugdev", MODE="660"
```

Правило, которое при подключении USB-устройства запускает скрипт `/etc/udev/usb_on.sh`, и сделает необходимые действия (например, запишет в `log`-файл необходимую информацию), будет выглядеть следующим образом:

```
ACTION=="add", SUBSYSTEM=="block",
ENV{ID_BUS}=="usb|mmc|memstick|ieee1394", RUN+="/bin/bash
/etc/udev/usb_on.sh %E{ID_SERIAL_SHORT} %E{ID_MODEL} %E{ID_VENDOR
}"
```

где:

- ACTION – отслеживаемое действие, `add` – подключение устройств, `remove` – отключение;
- ENV – перечень отслеживаемых устройств по типу;
- RUN – исполняемое действие.

Скрипту `usb_on.sh udev` передает следующие данные:

- `%E{ID_SERIAL_SHORT}` – серийный номер USB-устройства;
- `%E{ID_MODEL}` – модель USB-устройства;
- `%E{ID_VENDOR}` – производитель USB-устройства.

Использование скрипта позволяет выполнять более гибкую настройку правил: можно не только монтировать устройства, но и выполнять другие действия (копировать, менять владельца и так далее). Так как `udev` выполняется от имени учетной записи суперпользователя `root`, то и скрипт будет выполняться от имени соответствующей учетной записи. Также допускается задавать тип доступа к информации на носителе, например, «только для чтения».

Далее приводятся примеры оформления других возможных правил для `udev`:

- отключить все USB-порты:

```
SUBSYSTEM=="usb", ACTION=="add", ATTR{authorized}="0"
```

- назначить постоянное имя файлу устройства второго IDE-диска:

```
KERNEL=="sdb", NAME="my_spare"
```

- игнорировать второй USB SCSI/IDE-диск, подключенный по USB:

```
SUBSYSTEM=="usb", KERNEL=="hdb", ATTR{authorized}="0"
```

### 3.5.6. Управление монтированием блочных устройств

При монтировании блочных устройств используется утилита `mount`, модифицированная для монтирования устройства владельцем или пользователем. В процессе монтирования от имени пользователя ожидается два параметра: конкретное наименование файла устройства и конкретное наименование точки монтирования.

Для предоставления локальным пользователям возможности монтирования ФС съемных накопителей нужно наличие в файле `/etc/fstab` следующей записи:

```
/dev/sdb1 /media/usb vfat rw,noauto,user 0 0
```

### 3.5.7. Настройка ограничений на использование внешних носителей

USBGuard предназначен для настройки ограничений на использование USB-устройств.

При подключении USB-устройства к системе служба последовательно сканирует существующие правила. Если соответствующее правило найдено, оно либо разрешает, блокирует или отклоняет устройство в зависимости от цели правила. Если соответствующее правило не найдено, решение принимается на основе неявной цели по умолчанию. Это неявное значение по умолчанию заключается в блокировке устройства до тех пор, пока пользователь не примет решение.

Модуль предоставляет следующие возможности:

- сканирование подключенных устройств;
- выбор и добавление устройств в набор правил из списка подключенных устройств;
- создание предустановленных правил для распространенных сценариев;
- создание правил по дескрипторам интерфейса: CC:SS:PP;

- создание правил по свойствам USB-устройства: PID, VID;
- создание правил по хэшу устройства по PID+VID+SN;
- создание сложных правил с дополнительными условиями;
- загрузка правил из csv-файла;
- редактирование значений в созданных правилах;
- просмотр журнала событий подключения/отключения USB-устройств.

**Примечание.** Не рекомендуется использовать USB-устройства с не читаемым или коротким серийным номером.

### 3.5.7.1. Настройка USBGuard в консольном интерфейсе

#### 3.5.7.1.1. Установка

Установить пакет:

```
# apt-get install usbguard
```

Создать исходный набор правил на основе подключенных в данный момент USB-устройств (в правила с действием «allow» будут добавлены все подключенные устройства):

```
# usbguard generate-policy > /etc/usbguard/rules.conf
```

**Примечание.** Создать исходный набор правил, не генерируя атрибуты хэш-функции для устройств:

```
# usbguard generate-policy --no-hashes > /etc/usbguard/rules.conf
```

Запустить службу usbguard и добавить ее в автозагрузку:

```
# systemctl enable --now usbguard
```

#### 3.5.7.1.2. Синтаксис

Синтаксис команды usbguard:

```
usbguard [ОПЦИИ] <команда> [ОПЦИИ КОМАНДЫ]
```

Где команда может принимать одно из следующих значений:

- get-parameter <name> – получить значение параметра времени выполнения;
- set-parameter <name> <value> – установить значение параметра времени выполнения;



- `list-devices` – вывести список всех USB-устройств, распознаваемых демоном USBGuard;
- `allow-device <id|rule|p-rule>` – разрешить устройству взаимодействовать с системой;
- `block-device <id|rule|p-rule>` – деавторизовать устройство;
- `reject-device <id|rule|p-rule>` – деавторизовать и удалить устройство из системы;
- `list-rules` – вывести набор правил (политику), используемый демоном USBGuard;
- `append-rule <rule>` – добавить правило в набор правил;
- `remove-rule <id>` – удалить правило из набора правил;
- `generate-policy` – сгенерировать набор правил (политику) на основе подключенных USB-устройств;
- `watch` – следить за событиями интерфейса IPC и выводить их на стандартный вывод;
- `read-descriptor` – прочитать дескриптор USB из файла и вывести его в удобочитаемой форме;
- `add-user <name>` – добавить пользователя/группу USBGuard IPC (требуется права root);
- `remove-user <name>` – удалить пользователя/группу USBGuard IPC (требуется права root).

Дополнительную информацию смотрите на соответствующих страницах встроенного руководства, например:

```
$ usbguard add-user -h
```

Примеры:

- вывести список подключенных устройств:

```
$ usbguard list-devices
```

## ЛКНВ.11102-01 99 01

```
1: allow id 1d6b:0002 serial "0000:00:0b.0" name "EHCI Host Controller"
hash "SEiVqUWwefEKDMN90JUyXkFIvvFPJmvPTRKIlVCvlvE=" parent-hash
"BfFg9THiKJIvTnHGcJHfrWc00WcrIzhayJ9C3BiPYho=" via-port "usb1" with-interface
09:00:00 with-connect-type ""
```

```
2: allow id 1d6b:0001 serial "0000:00:06.0" name "OHCI PCI host
controller" hash "lUN32sIeMBBlD8Pd82mxu95iCTw8oKlT8izDeg628/o=" parent-hash
"XokStAV3JXWqQkW016YD7ZPFcHse1OtwuGmVNBCe46E=" via-port "usb2" with-interface
09:00:00 with-connect-type ""
```

```
3: allow id 13fe:3e00 serial "11101094E6BA1A00A4A5200A" name "Silicon-
Power4G" hash "0Dcf+ZVj7NkORO+RN1kxcP0rgDl6CqvqkcYkcf+0X/k=" parent-hash
"SEiVqUWwefEKDMN90JUyXkFIvvFPJmvPTRKIlVCvlvE=" via-port "1-1" with-interface
08:06:50 with-connect-type "unknown"
```

**- временно заблокировать устройство с id 3:**

```
$ usbguard block-device 3
```

**- для постоянной блокировки или авторизации USB-устройство, следует использовать опцию -p. В этом случае в текущую политику будет добавлено правило для конкретного устройства:**

```
$ usbguard block-device 3 -p
```

**- вывести список правил:**

```
$ usbguard list-rules
```

```
4: allow id 1d6b:0002 serial "0000:00:0b.0" name "EHCI Host
Controller" hash "SEiVqUWwefEKDMN90JUyXkFIvvFPJmvPTRKIlVCvlvE=" parent-hash
"BfFg9THiKJIvTnHGcJHfrWc00WcrIzhayJ9C3BiPYho=" via-port "usb1" with-interface
09:00:00 with-connect-type ""
```

```
5: allow id 1d6b:0001 serial "0000:00:06.0" name "OHCI PCI host
controller" hash "lUN32sIeMBBlD8Pd82mxu95iCTw8oKlT8izDeg628/o=" parent-hash
"XokStAV3JXWqQkW016YD7ZPFcHse1OtwuGmVNBCe46E=" via-port "usb2" with-interface
09:00:00 with-connect-type ""
```

```
6: block id 090c:1000 serial "2010121200000186" name "USB DISK" hash
"2dfdMHZxF5olAaNbsh68G4fpzD3iQLPL3+M7KHnSRjE=" parent-hash
"SEiVqUWwefEKDMN90JUyXkFIvvFPJmvPTRKIlVCvlvE=" via-port "1-1" with-interface
08:06:50 with-connect-type "unknown"
```

**- дать пользователю user полный доступ к разделам «devices» и «exceptions», также, user будет иметь возможность просматривать и изменять текущую политику:**

```
# usbguard add-user -u user --devices ALL --policy modify,list --exceptions ALL
- удалить права у пользователя user:
# usbguard remove-user -u user
```

### 3.5.7.1.3. Конфигурация

Основной файл конфигурации USBGuard:

```
/etc/usbguard/usbguard-daemon.conf
```

В параметре `PresentDevicePolicy` определяется как обращаться с устройствами, которые уже подключены, при запуске службы. По умолчанию для `PresentDevicePolicy` установлено значение `apply-policy`, т.е. USBGuard оценивает набор правил для каждого подключенного устройства. Это наиболее безопасный параметр, который обеспечивает безопасность даже при перезагрузке службы.

В параметре `ImplicitPolicyTarget` настраивается политика обработки устройств по умолчанию, если ни одно правило не соответствует. Самый безопасный вариант – `block`.

По умолчанию служба `usbguard` записывает события в файл `/var/log/usbguard/usbguard-audit.log` (параметр `AuditFilePath`).

В параметре `IPCAccessControlFiles` указывается каталог, файлы в котором будут интерпретироваться службой как файлы определения управления доступом. Каждый файл в этом каталоге обрабатывается следующим образом:

- базовое имя файла интерпретируется как имя пользователя, UID, имя группы или GID;
- если имя файла начинается с `:` (двоеточие), предполагается, что оставшая часть имени представляет собой идентификатор группы (имя группы или GID в случае, если строка содержит только числовые значения);
- если имя файла не начинается с `:`, то имя файла интерпретируется как идентификатор пользователя (имя пользователя или UID в случае, если строка содержит только числовые значения).

Содержимое файла анализируется построчно. Каждая строка определяет привилегии раздела в формате:

```
Раздел=привилегия [привилегия ...]
```

Если раздел опущен, предполагается, что для этого раздела привилегии не предоставлены.

Например:

```
IPCAccessControlFiles=/etc/usbguard/IPCAccessControl.d/
```

Пример ACL пользователя user (содержимое файла /etc/usbguard/IPCAccessControl.d/user):

```
Devices=list,modify,listen
```

```
Policy=list,modify
```

```
Exceptions=listen
```

Доступные следующие разделы и привилегии:

1) Devices:

- modify: изменение состояния авторизации устройств, включая постоянные изменения (т. е. изменение правил для конкретных устройств в политике);
- list: возможность получить список распознанных устройств и их атрибутов;
- listen: прослушивание присутствия устройства и изменение политики устройства;

2) Policy:

- modify: возможность добавления/удаления правил в политике;
- list: возможность просмотра текущей политики;

3) Exceptions:

- listen: возможность получения сообщений об исключениях;

4) Parameters:

- modify: возможность установить значения параметров времени выполнения;

`-list`: возможность получить значения параметров времени выполнения.

**Примечание.** Для управления списком управления доступом (ACL), следует использовать команды `usbguard add-user` и `usbguard remove-user`.

#### 3.5.7.1.4. Правила

Набор правил USBGuard по умолчанию хранится в файле `/etc/usbguard/rules.conf`. Но ручное редактирование правил обычно не требуется.

Синтаксис правила:

```
rule ::= target attributes.
target ::= "allow" | "block" | "reject".
attributes ::= attributes | attribute.
attributes ::= .
```

Например, правило для принтера Canon, подключенного через USB:

```
allow id 04a9:177a name "Canon E400" serial "F572EC" via-port "1-2" hash "eql9yA8m+5VVMmhXOvbUzwNPDGCAPq+fxIQHvbptlsY="
```

Первое слово в правиле определяет политику (цель):

- `allow` – добавляет устройство в белый список;
- `block` – останавливает обработку устройства;
- `reject` – удаляет устройство из системы.

**Примечание.** USBGuard использует термины `block` и `reject` в следующих значениях:

- `block` – пока не взаимодействовать с этим устройством;
- `reject` – игнорировать это устройство, как будто его не существует.

Затем следует набор атрибутов с их параметрами. Атрибуты устройства – это определенные значения, считываемые с USB-устройства после его подключения к системе.

Все атрибуты поддерживают две формы:

- однозначная:

```
name value
```

- многозначная:

```
name [operator] { value1 value2 ... }
```

Возможные атрибуты приведены в таблице 1.

Таблица 1

Атрибут	Описание
id usb-device-id	Соответствует идентификатору USB-устройства. Идентификатор USB-устройства представляет собой пару, разделенную двоеточием, в форме vendor_id:product_id. И vendor_id и product_id – это 16-битные числа в шестнадцатеричной системе счисления. Символ звездочки можно использовать для соответствия любому идентификатору устройства *: * или любому идентификатору продукта от конкретного поставщика, например, 1234: *.
id [оператор] {usb-device-id ... }	Соответствует набору идентификаторов USB-устройств
hash "value"	Соответствует хэшу, вычисленному на основе значений атрибутов устройства и данных дескриптора USB. Хэш вычисляется USBGuard для каждого устройства
hash [оператор] { "value" ... }	Соответствует набору хэшей устройств
parent-hash "value"	Соответствует хэшу родительского устройства
parent-hash [оператор] { "value" ... }	Соответствует набору хэшей родительского устройства
name "device-name"	Соответствует имени USB-устройства
name [оператор] { "device-name" ... }	Соответствует набору имен USB-устройств
serial "serial-number"	Соответствует серийному номеру USB-устройства
serial [оператор] { "serial-number" ... }	Соответствует набору серийных номеров USB-устройств
via-port "port-id"	Соответствует USB-порту, через который подключено устройство. Значение port-id – это идентификатор USB-порта, специфичный для платформы. В Linux он имеет форму «usbN» в случае USB-контроллера (точнее, «корневого концентратора») или «bus-port[.port[.port ...]]» (например, 1-2, 1-2.1, ...) в случае USB-устройства. Некоторые системы имеют нестабильную нумерацию портов, которая меняется после перезагрузки системы или перезагрузки определенных модулей ядра. Используйте атрибут parent-hash, чтобы убедиться, что устройство подключено через определенное родительское устройство

## Окончание таблицы 1

Атрибут	Описание
via-port [оператор] { "port-id" ... }	Соответствует набору портов USB
with-interface interface-type	Соответствует типу интерфейса, предоставляемому USB-устройством. Тип интерфейса указывается как три 8-битных числа в шестнадцатеричном формате, разделенных двоеточием, т. е. cc:ss:pp. Числа обозначают класс интерфейса (cc), подкласс (ss) и протокол (pp), назначенные USB-IF. Вместо номера подкласса и протокола можно использовать символ *, чтобы соответствовать всем подклассам или протоколам. Сопоставление определенного класса и определенного протокола не допускается, т.е. если в качестве номера подкласса используется «*», то для протокола также необходимо использовать «*»
with-interface [оператор] { interface-type interface-type ... }	Сопоставляет набор типов интерфейсов с набором интерфейсов, предоставляемых USB-устройством
with-connect-type "connect-type"	Соответствует атрибуту устройства USB-порт/connect_type
with-connect-type [оператор] { "connect-type" ... }	Соответствует набору атрибутов устройства USB-порт/connect_type

Необязательный параметр оператор является одним из:

- all-of – набор атрибутов устройства должен содержать все указанные значения;
- one-of – набор атрибутов устройства должен содержать хотя бы одно из указанных значений;
- none-of – набор атрибутов устройства не должен содержать ни одного из указанных значений;
- equals – набор атрибутов устройства должен содержать данный набор значений;
- equals-ordered – набор атрибутов устройства должен содержать данный набор значений в том же порядке;
- match-all – набор атрибутов устройства должен быть подмножеством указанных значений.



Если оператор не указан, ему присваивается значение `equals`.

За исключением цели, все остальные поля правила являются необязательными. Правило, в котором указана только цель, будет соответствовать любому устройству. Это позволяет администратору политики указать явную цель по умолчанию. Если к устройству не применимо ни одно правило из политики, будет использоваться неявная цель, настроенная в `/etc/usbguard/usbguard-daemon.conf`.

Примеры политик:

- разрешить USB-накопители (карты-памяти) и заблокировать все остальное (например, мыши, клавиатуры):

```
allow with-interface 08:*:*
```

- разрешить подключение принтера Canon, подключенного через USB, через определенный порт:

```
allow id 04a9:177a name "Canon E400" serial "F572EC" via-port "1-2" hash "eql9yA8m+5VVMmhXOvbUzwNPDGCAPq+fxIQHvbptlsY="
```

- отклонять устройства с подозрительной комбинацией интерфейсов (USB-накопитель, оснащенный клавиатурой или сетевым интерфейсом):

```
allow with-interface equals { 08:*:* }
reject with-interface all-of { 08:*:* 03:00:* }
reject with-interface all-of { 08:*:* 03:01:* }
reject with-interface all-of { 08:*:* e0:*:* }
reject with-interface all-of { 08:*:* 02:*:* }
```

### 3.5.7.2. Настройка в веб-интерфейсе ЦУС (alterator-usbguard)

Настройка ограничений на использование внешних носителей с использованием USBGuard может производиться в веб-интерфейсе ЦУС (пакет `alterator-fbi`) в меню «Система» → пункт «USBGuard» (пакет `alterator-usbguard`).

Должны быть установлены:

```
# apt-get install alterator-fbi
# apt-get install alterator-usbguard
```

Перезапустить службу:

```
# systemctl restart ahttpd
```

Далее откройте в веб-браузере адрес ЦУС (пакет alterator-fbi) `https://localhost:8080` или `https://ip_address:8080` и введите пароль администратора.

В подразделе «Система»→ «USBGuard» отображается (рис. 11):

- текущее состояние службы usbguard;
- список пользователей и групп, которые могут редактировать правила, сообщения об ошибках и предупреждения.

**Примечание.** Добавить/удалить пользователя/группу, которые могут редактировать правила, можно в командной строке, например:

- дать пользователю user полный доступ к разделам «devices» и «exceptions», также, user будет иметь возможность просматривать и изменять текущую политику:

```
# usbguard add-user -u user --devices ALL --policy modify,list --exceptions ALL
```

- удалить права у пользователя user:

```
# usbguard remove-user -u user
```

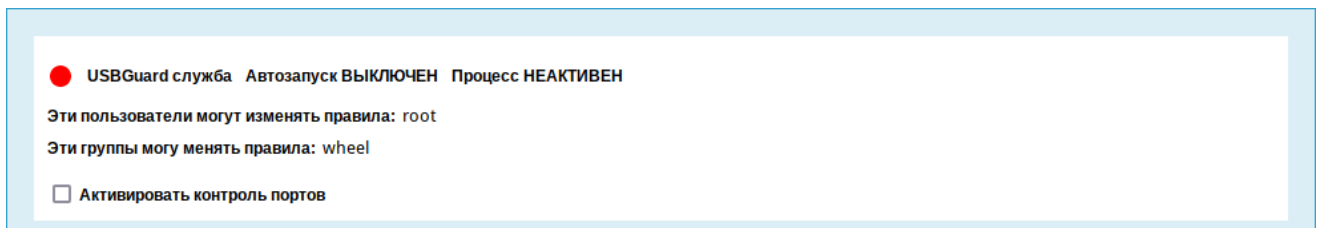


Рис. 11 – Информационное поле USBGuard

Для включения контроля за USB-устройствами:

- установить отметку в пункте «Активировать контроль портов»;
- нажать кнопку «Проверить», а затем кнопку «Применить».

Служба usbguard будет запущена и добавлена в автозагрузку (рис. 12).

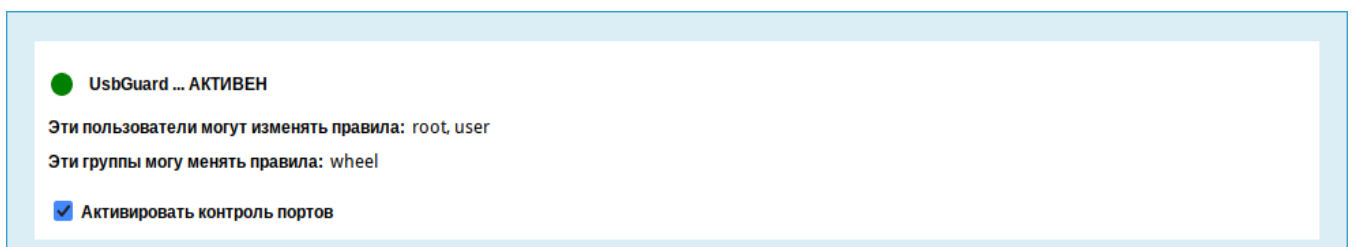


Рис. 12 – Служба usbguard запущена и добавлена в автозагрузку

### 3.5.7.2.1. Список USB-устройств

Если служба `usbguard` запущена (рис. 12), в веб-интерфейсе будет отображен список текущих подключенных USB-устройств (рис. 13).

Список устройств									
N	Порт	CC:SS:PP	VID	Вендор	PID	Название	Серийный номер	Хэш	Статус
1	usb1	09:00:00	1d6b	Linux Foundation	0002	xHCI Host Controller	0000:00:14.0	jEP/6WzviqdJ5VSeTUY8PatCNBKeaREvo2OqdpIND/o=	allow
2	usb2	09:00:00	1d6b	Linux Foundation	0003	xHCI Host Controller	0000:00:14.0	prM+Jby/bFHCn2INjQdAMbgc6tse3xVx+hZwjOPHSdQ=	allow
3	1-2	08:06:50	090c	Silicon Motion, Inc. - Taiwan (formerly Feiya Technology Corp.)	1000	USB DISK	2010121200000186	2dfdMHZxF5oIAaNbsh68G4fpzD3iQLPL3+M7KHnSRJE=	block
4	1-3	0e:*:*	04ca	Lite-On Technology Corp.	707f	HP Wide Vision HD Camera	200901010001	FD0U/H7cT78kTsmXKgrG/Z2Q2O7cu+JpQgCs24460d8=	block
5	1-10	e0:*:*	8087	Intel Corp.	0aaa			BpLyFNeilMugqZSYbuMBAIxEHNoXynuJ0UMg83HPZkdU=	block

Рис. 13 – Таблица «Список устройств»

В столбце «Статус» отображается текущее состояние USB-устройства:

- «allow» – разрешенное устройство;
- «block» – заблокированное устройство.

Для редактирования состояния USB-устройства, выделите строку с нужным USB-устройством и нажмите кнопку «Разблокировать»/«Заблокировать». При этом будет добавлено соответствующее правило в таблицу «Хэш» (рис. 15).

**Примечание.** Если активен «Белый список», то для устройства со статусом «block» будет активна кнопка «Разблокировать», если активен «Черный список», то для устройства со статусом «allow» будет активна кнопка «Заблокировать».

Кнопка «Сканировать устройства» позволяет обновить список подключенных USB-устройств.

### 3.5.7.2.2. Предустановки

Правила могут работать в режиме белого или черного списка, который выбирается в разделе «Предустановки» (рис. 14). После установки режима «Черный список», будут заблокированы только перечисленные в данном списке USB-устройства. После установки режима «Белый список», будут заблокированы все USB-устройства, кроме перечисленных в данном списке.

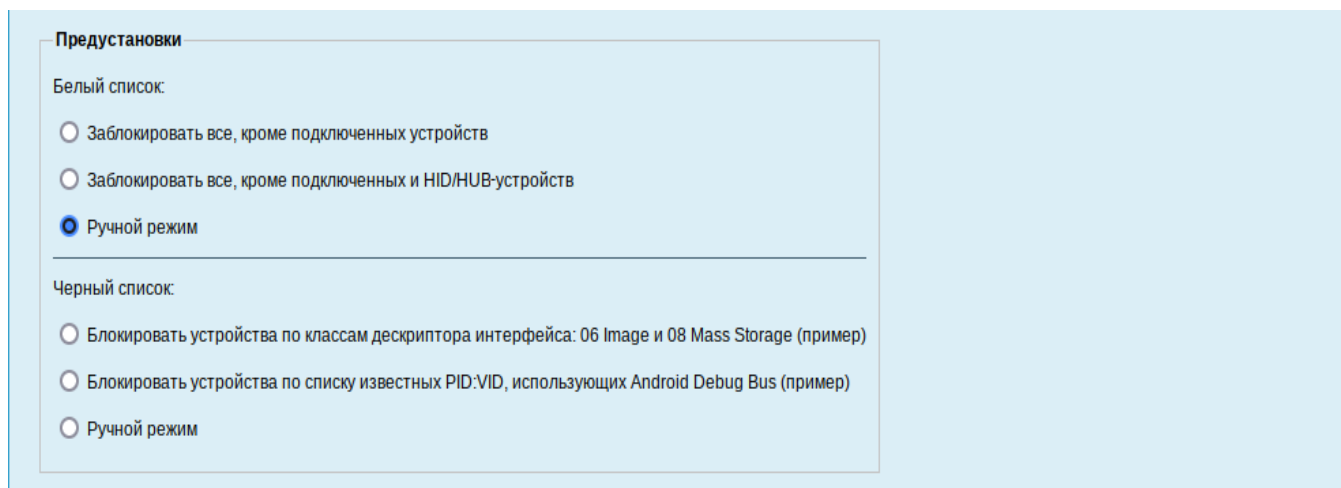


Рис. 14 – Предустановки

Существует возможность предварительной настройки правил списков, для этого:

1) выберите соответствующий режим:

- «Белый список»:

- а) «Заблокировать все, кроме подключенных устройств» – в правила (таблица «Хэш» рис. 15) с действием «allow» будут добавлены все подключенные устройства. Все новые USB-устройства будут заблокированы – будут отображаться в таблице «Список устройств» (рис. 13) со статусом «block»;
- б) «Заблокировать все, кроме подключенных и HID/HUB устройств» – в правила с действием «allow» будут добавлены все подключенные устройства (таблица «Хэш» рис. 15) и все устройства с интерфейсами 03:\*:\* и 09:\*:\* (таблица «Маски CC:SS:PP» рис. 16). Все новые USB-устройства кроме HID/HUB-устройств (клавиатуры, мыши, джойстики, USB-концентраторы) будут заблокированы – отображаться в таблице «Список устройств» (рис. 13) со статусом «block»; Подробнее см. п. 3.5.7.2.3.1;
- в) «Ручной режим» – позволяет установить свои правила;

- «Черный список»:

а) «Блокировать устройства по классам дескриптора интерфейса: 06 Image и 08 Mass Storage» – в правила (таблица «Маски CC:SS:PP» рис. 16) с действием «block» будут добавлены все устройства с интерфейсами 08:\*:\* и 06:\*:\*. Все USB-устройства Mass Storage Device (USB-накопитель, карта памяти, кардридер, цифровая фотокамера) и Image (веб-камера, сканер) будут заблокированы – будут отображаться в таблице «Список устройств» со статусом «block» Подробнее см. п. 3.5.7.2.3.1 Правила по классу интерфейса;

б) «Блокировать устройства по списку известных PID:VID, использующих Android Debug Bus» – в правила (таблица «Маски VID:PID» рис. 19) с действием «block» будут добавлены известные Android-устройства. Все Android-устройства будут заблокированы – будут отображаться в таблице «Список устройств» со статусом «block» Подробнее см. п. Правила по VID&PID;

в) «Ручной режим» – позволяет установить свои правила;

2) нажмите кнопку «Проверить». Будут показаны планируемые изменения (рис. 15);

3) если изменения корректные, нажмите кнопку «Применить»;

4) для отмены изменений, до нажатия кнопки «Применить», следует выбрать пункт «Ручной режим» и нажать кнопку «Проверить», а затем «Применить».

Другие правила							
<input type="checkbox"/>	N	Правило	Действие				
<input type="button" value="Добавить"/> <input type="button" value="Удалить"/>							
Маски CC:SS:PP							
<input type="checkbox"/>	N	CC:SS:PP	Действие				
<input type="checkbox"/>	5	03:??	allow				
<input type="button" value="Добавить"/> <input type="button" value="Удалить"/>							
Маски VID:PID							
<input type="checkbox"/>	N	VID	Вендор	PID	Название	Порт	Действие
<input type="button" value="Добавить"/> <input type="button" value="Удалить"/>							
Хэш							
<input type="checkbox"/>	N	Хэш	Описание	Действие			
<input type="checkbox"/>	0	"jEPi6WzviqdJ5VSeTUY8PatCNBKeaREvo2OqdpIND/o="	"xHCI Host Controller"	allow			
<input type="checkbox"/>	4	"prM+Jby/bFHCn2INjQdAMbge6tse3xVx+hZwjOPHSdQ="	"xHCI Host Controller"	allow			
<input type="checkbox"/>	2	"2dfdMHZxF6oIAaNbeh68G4fpzD3iQLPL3+M7KHnSRJE="	"USB-DISK"	allow			
<input type="checkbox"/>	3	"FD0U/H7cT78kTSmXKgrG/Z2Q2O7cu+JpQgCe24460d8="	"HP Wide Vision HD Camera"	allow			
<input type="checkbox"/>	4	"BpLyFNeiMugqZSYbuMBAIxEhNoXynuJ0UMg83HPZkdU="	""	allow			
<input type="checkbox"/>	6	"an+hPjkSqC/s+AnuuG9Ke1ycUY5865rBmC3TWH/SXso="	"USB2.0 HUB"	allow			
<input type="checkbox"/>	--	jEPi6WzviqdJ5VSeTUY8PatCNBKeaREvo2OqdpIND/o=	xHCI Host Controller	allow			
<input type="checkbox"/>	--	prM+Jby/bFHCn2INjQdAMbge6tse3xVx+hZwjOPHSdQ=	xHCI Host Controller	allow			
<input type="checkbox"/>	--	2dfdMHZxF6oIAaNbeh68G4fpzD3iQLPL3+M7KHnSRJE=	USB DISK	allow			
<input type="checkbox"/>	--	FD0U/H7cT78kTSmXKgrG/Z2Q2O7cu+JpQgCs24460d8=	HP Wide Vision HD Camera	allow			
<input type="checkbox"/>	--	BpLyFNeiMugqZSYbuMBAIxEhNoXynuJ0UMg83HPZkdU=		allow			
<input type="checkbox"/>	--	an+hPjkSqC/s+AnuuG9Ke1ycUY5865rBmC3TWH/SXso=	USB2.0 HUB	allow			
<input type="button" value="Загрузить из файла"/> <input type="button" value="Обзор..."/> <span>Файл не выбран.</span>				<input type="button" value="Добавить"/> <input type="button" value="Удалить"/>			
<input type="button" value="Проверить"/> <input type="button" value="Применить"/>							

Рис. 15 – Планируемые изменения

### 3.5.7.2.3. Добавление/удаление правил

Для добавления нового правила должен быть выбран пункт «Ручной режим» в белом или черном списках. Если «Ручной режим» выбран в белом списке, правило будет добавлено с действием «allow», если в черном – с действием «block».

#### 3.5.7.2.3.1. Правила по классу интерфейса

Назначение USB-устройств может определяться кодами классов, которые сообщаются USB-узлу для загрузки необходимых драйверов. Коды классов позволяют унифицировать работу с однотипными устройствами разных производителей. Устройство может поддерживать один или несколько классов, максимальное количество которых определяется количеством доступных endpoints. Например, широко известны устройства класса Human Interface Device (HID) – мыши, клавиатуры, игровые манипуляторы и т. д. или устройства Mass Storage – USB-накопители, карты памяти и т. д.

Примечание. Класс интерфейса указывается как три 8-битных числа в шестнадцатеричном формате, разделенных двоеточием (CC:SS:PP). Числа обозначают класс интерфейса (CC), подкласс (SS) и протокол (PP). Вместо номера подкласса и протокола можно использовать символ \*, чтобы соответствовать всем подклассам или протоколам. Сопоставление определенного класса и определенного протокола не допускается, то есть если в качестве номера подкласса используется \*, то для протокола также необходимо использовать \*.

Добавление правила по маске:

- под таблицей «Маски CC:SS:PP» нажать кнопку «Добавить»;
- в поле «CC:SS:PP» вписать маску, например, правило для всех устройств с интерфейсами 09:\*:\* (рис. 16);
- нажать кнопку «Проверить». Корректное правило будет выделено зеленым цветом (рис. 16), некорректное – красным;
- исправить или удалить некорректное правило и повторно нажать кнопку «Проверить»;
- нажать кнопку «Применить» для активации правила. Правило для всех устройств с интерфейсами 09:\*:\* будет добавлено.

Маски CC:SS:PP						
<input type="checkbox"/>	N	CC:SS:PP	Описание	Порт	Действие	
<input type="checkbox"/>	5	03:*:			allow	
<input type="checkbox"/>	--	09:*:	--	--	allow	

Рис. 16 – Добавление правила для всех устройств с интерфейсами 09:\*:\*

Маски CC:SS:PP						
<input type="checkbox"/>	N	CC:SS:PP	Описание	Порт	Действие	
<input type="checkbox"/>	5	03:*:			allow	
<input type="checkbox"/>	--	09:*:	--	--	allow	

Рис. 17 – Проверка правила

Удаление правила по маске:

- в таблице «Маски CC:SS:PP» установить отметку в поле с соответствующим правилом;
- нажать кнопку «Удалить». Правило будет готово к удалению (рис. 18);
- нажать кнопку «Проверить»;
- нажать кнопку «Применить» для удаления правила.

Маски CC:SS:PP						
<input type="checkbox"/>	N	CC:SS:PP	Описание	Порт	Действие	
<input checked="" type="checkbox"/>	5	03:*.*			allow	
<input type="checkbox"/>	6	09:*.*			allow	

Рис. 18 – Удаление правила для всех устройств с интерфейсами 03:\*.\*

### 3.5.7.2.3.2. Правила по VID&PID

Каждое USB-устройство содержит атрибуты, куда входит идентификатор разработчика устройства (VID) и идентификатор изделия (PID). На основании этих идентификаторов узел (компьютер) ищет методы работы с этим устройством (обычно это выражается в требовании установить драйверы, поставляемые разработчиком устройства).

**Примечание.** И VID и PID – это 16-битные числа в шестнадцатеричной системе счисления. В правиле можно также использовать символ \*:

- для соответствия любому идентификатору устройства \*:\*;
- для соответствия любому идентификатору продукта от конкретного поставщика, например, 090c:\*

Добавление правила по VID&PID:

- под таблицей «Маски VID:PID» нажать кнопку «Добавить»;
- в поле «VID» вписать идентификатор разработчика устройства (VID), а в поле «PID» идентификатор изделия (PID) (рис. 19);
- нажать кнопку «Проверить». Корректное правило будет выделено зеленым цветом, некорректное – красным;
- исправить или удалить некорректное правило и повторно нажать кнопку «Проверить»;



- нажать кнопку «Применить» для активации правила.

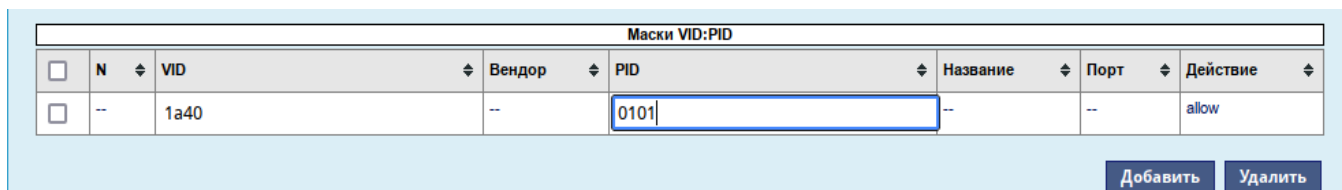


Рис. 19 – Добавление правила по VID&PID

Удаление правил(а) по VID&PID:

- в таблице «Маски VID:PID» установить отметку в поле с соответствующим правилами (рис. 20);
- нажать кнопку «Удалить». Правила будут готовы к удалению (отображаются зачеркнутым);
- нажать кнопку «Проверить»;
- нажать кнопку «Применить» для удаления правил.

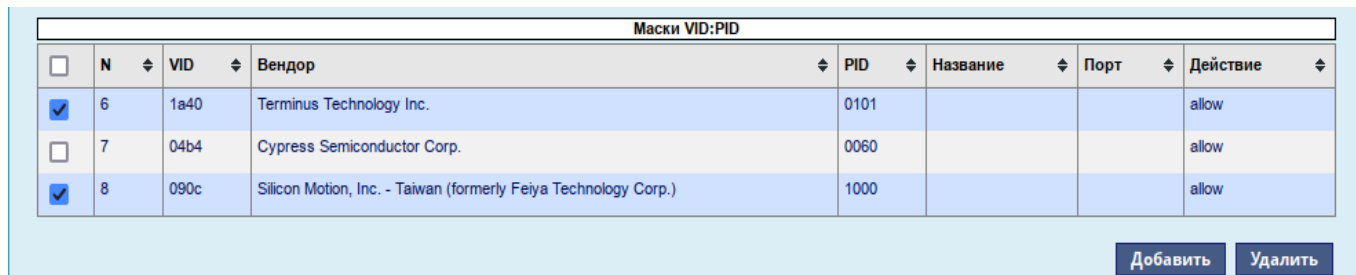


Рис. 20 – Выделение правил

### 3.5.7.2.3.3. Правила по хэшу

Для каждого USB-устройства USBGuard вычисляет хэш на основе значений атрибутов устройства и данных дескриптора USB (PID+VID+SN).

Добавление правила по хэшу:

- под таблицей «Хэш» (см. рис. 15) нажать кнопку «Добавить»;
- в поле «Хэш» вписать хэш устройства (рис. 21);
- нажать кнопку «Проверить». Корректное правило будет выделено зеленым цветом, некорректное – красным;

- исправить или удалить некорректное правило и повторно нажать кнопку «Проверить»;
- нажать кнопку «Применить» для активации правила.

Хэш				
<input type="checkbox"/>	N	Хэш	Описание	Действие
<input type="checkbox"/>	0	"jEP/6WzviqdJ5VSeTUy8PatCNBKeaREvo2OqdpIND/o="	"xHCI Host Controller"	allow
<input type="checkbox"/>	1	"prM+Jby/bFHCn2INjQdAMbgc6tse3xVx+hZwjOPHSdQ="	"xHCI Host Controller"	allow
<input type="checkbox"/>	2	"FD0U/H7cT78kTsmXKgrG/Z2Q2O7cu+JpQgCs24460d8="	"HP Wide Vision HD Camera"	allow
<input type="checkbox"/>	3	"BpLyFNeiMugqZSYbuMBAixEhNoXynuj0UMg83HPZkdU="	"	allow
<input type="checkbox"/>	--	2dfdMHZxF5olAaNbsh68G4fpzD3iQLPL3+M7KHnSRjE=	--	allow

Файл не выбран.

Рис. 21 – Добавление правила по хэшу

Удаление правил(а) по хэшу:

- в таблице «Хэш» установить отметку в поле с соответствующим правилом;
- нажать кнопку «Удалить». Правила будут готовы к удалению (отображаются зачеркнутым);
- нажать кнопку «Проверить»;
- нажать кнопку «Применить» для удаления правил(а).

#### 3.5.7.2.3.4. Другие правила

Модуль позволяет создать сложные правила с дополнительными условиями.

Добавление правила:

- под таблицей «Другие правила» (рис. 22) нажать кнопку «Добавить»;
- в поле «Правило» вписать правило (рис. 22). Например, правило, разрешающее подключение принтера только через определенный порт:

```
allow id 04a9:177a name "Canon E400" serial "F572EC" via-port "1-2" hash "eql9yA8m+5VVMmhXOvbUzwNPDGCApQ+f×IQHvbptlsY="
```

- нажать кнопку «Проверить». Корректное правило будет выделено зеленым цветом, некорректное – красным;
- исправить или удалить некорректное правило и повторно нажать кнопку «Проверить»;

- нажать кнопку «Применить» для активации правила.

Другие правила			
<input type="checkbox"/>	N	Правило	Действие
<input type="checkbox"/>	9	allow id 5cBE:fc52 serial "=5G-xMpl" hash "6CJS2jC_" parent-hash "ij~dJ" via-port "A?DYST+" with-interface { 00:c1:A6 2E:* 1e:* c6:ae:0F 9F:* 3C:* }	allow
<input type="checkbox"/>	--	allow id 04a9:177a name "Canon E400" serial "F572EC" via-port "1-2" hash "eqI9yA8m+5VVMmhXOvbUzwNPDGCAPq+fxIQHvbptIsY="	allow

Рис. 22 – Добавление сложного правила

Удаление правила:

- в таблице «Другие правила» установить отметку в поле с соответствующим правилом;
- нажать кнопку «Удалить». Правила будут готовы к удалению (отображаются зачеркнутым);
- нажать кнопку «Проверить»;
- нажать кнопку «Применить» для удаления правила.

#### 3.5.7.2.3.5. Загрузка правил из файла

Для использования правил из файла, они должны быть добавлены в csv-файл, по одному правилу в каждой строке. Строка должна иметь вид:

```
allow/block, Interface, PID:VID, Hash
```

Например:

```
allow, , 090c:1000, "2dfdMHZxF5olAaNbsh68G4fpzD3iQLPL3+M7KHnSRjE="
allow, 00:00:* , ,
allow, , 1000:* ,
allow, , , "eqI9yA8m+5VVMmhXOvbUzwNPDGCAPq+fxIQHvbptIsY="
```

**Примечание.** Файл не должен содержать конфликтные правила – должны быть либо все allow, либо все block.

Для загрузки правил из файла:

- нажать кнопку «Обзор» (под таблицей «Хэш» см. рис. 15) и выбрать файл с правилами (рис. 23);
- нажать кнопку «Загрузить из файла»;

- нажать кнопку «Проверить». Корректные правила будут выделены зеленым цветом (рис. 24), некорректные – красным;
- нажать кнопку «Применить».

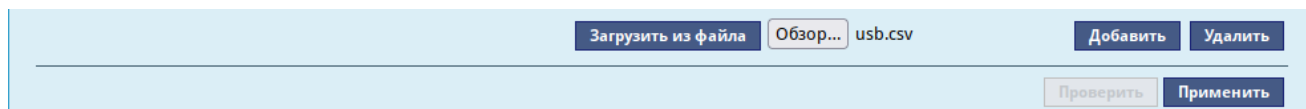


Рис. 23 – Выбор файла с правилами

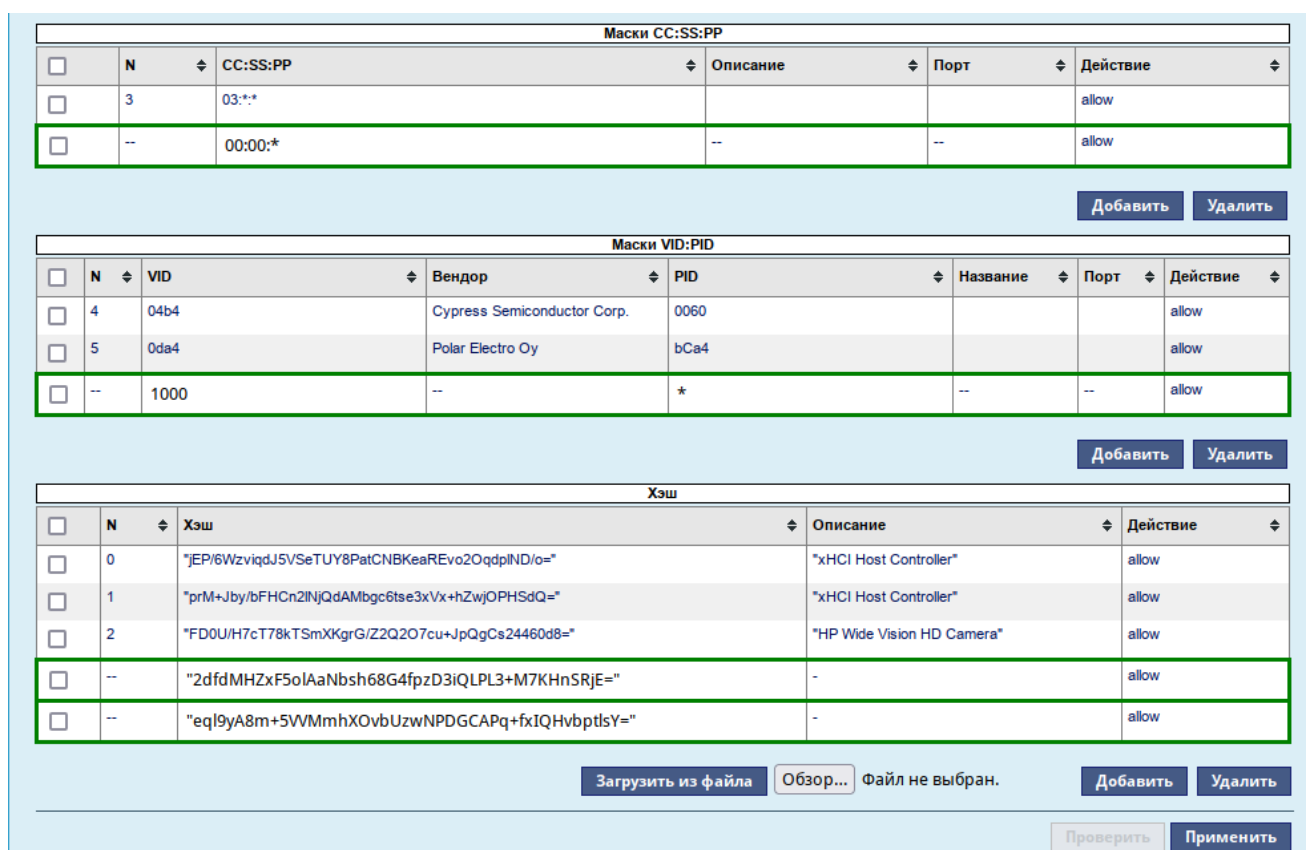


Рис. 24 – Проверка правил, загруженных из файла

⚠ При загрузке правил из файла политика тоже будет выбрана из файла. Если в файле указана политика противоположная текущей, все существующие правила будут удалены.

#### 3.5.7.2.4. Просмотр журнала аудита

Для просмотра журнала событий подключения/отключения USB-устройств (журнала аудита) нажмите кнопку «Журнал», расположенную в левом нижнем углу интерфейса модуля, раскроется журнал аудита (рис. 25).

Передвигаться по журналу к более старым или более новым записям можно с помощью кнопок «Назад» и «Вперед».

Есть возможность фильтровать данные с помощью специального поля.

**Примечание.** Фильтрация по логу USBGuard – строгая, регистрозависимая.

Для того чтобы скрыть панель журнала, нажмите кнопку «Свернуть».



Рис. 25 – Журнал аудита подключения/отключения USB-устройств

### 3.5.7.2.5. Модуль уведомлений для USBGuard

Модуль уведомлений обрабатывает сообщения от USBGuard при подключении/отключении USB-устройств, выводит сообщения в трее для пользователя, если устройство заблокировано (рис. 26).

Для возможности оповещения пользователя о блокировке USB-устройств установите пакет `usbguard-notifier`:

```
# apt-get install usbguard-notifier
```

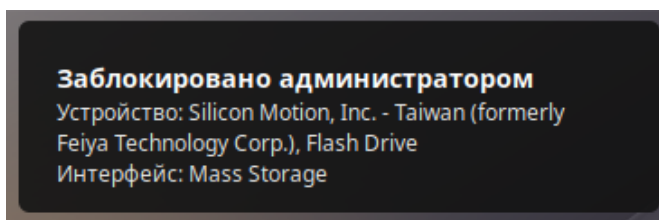


Рис. 26 – Уведомление о заблокированном устройстве

### 3.5.8. Настройка ограничения доступа к файловой системе USB-устройства

Модуль «USBMount» (пакет `alterator-usbmount`) позволяет ограничить доступ к файловой системе (ФС) USB-устройства по UID/GID. В модуле также предусмотрена возможность просмотра журнала событий подключения/отключения USB-устройств.

Особенности работы модуля:

- если для устройства не создано правило, то служба не вмешивается в логику монтирования USB-устройства;
- если для устройства создано правило, то служба монтирует блочные устройства на назначенном USB-устройстве в каталог `/media/alt-usb-mount/$user_$group` или `/media/alt-usb-mount/root_$group`, если пользователь не указан;
- служба назначает ACL для указанного пользователя и группы на каталог, в котором будет создана точка монтирования блочного устройства;
- в правилах можно указать только существующего локального пользователя и пользовательскую группу;
- доступ к USB-устройству, назначенному пользователю и группе, предоставляется полностью (`rw`);
- любой пользователь может отмонтировать устройство через стандартные средства ОС;
- служба не вмешивается в права самих ФС блочных устройств;
- рекомендуемая ФС для переносных носителей – `exFAT`.

**Примечание.** Особенности работы модуля с ФС:

- `EXT2/3/4`, `XFS`, `BTRFS` поддерживают права. Доступ к ФС будет также определяться назначенными правами самой ФС;
- `FAT16/FAT32/exFAT` не поддерживают права. Доступ будет полностью определяться через точку монтирования назначенную в `USBMount`;
- `ISO9660/UDF` поддерживает только `readonly`. Доступ будет предоставлен только на чтение;

- NTFS поддерживает права. Не рекомендуется использовать. В случае если ранее NTFS носитель был извлечен небезопасно, то носитель будет смонтирован только для чтения.

**Примечание.** В MATE автоматическое монтирование USB-устройств включено по умолчанию. Для отключения автоматического монтирования следует создать правило udev, например, файл `/etc/udev/rules.d/60-usb.rules`:

```
SUBSYSTEM=="block",ACTION=="add",ENV{ID_BUS}=="usb",ENV{UDISKS_AUTO}="0"
```

### 3.5.8.1. Установка

Должны быть установлены:

```
# apt-get install alterator-fbi
# apt-get install alterator-usbmount
```

Перезапустить службу:

```
# systemctl restart ahttpd
```

Далее откройте в веб-браузере адрес ЦУС (пакет `alterator-fbi`) `https://localhost:8080` или `https://ip_address:8080` и введите пароль администратора, перейдите в раздел «Система»→«USBMount».

### 3.5.8.2. Использование модуля

#### 3.5.8.2.1. Запуск/останов службы

В подразделе раздел «Система» → «USBMount» отображается текущее состояние службы USBMount (рис. 27).

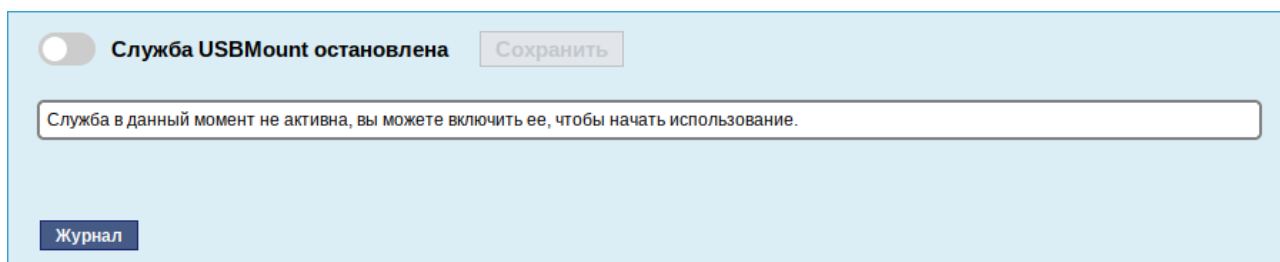


Рис. 27 – Служба USBMount остановлена

Для включения передвиньте переключатель «Служба USBMount остановлена» и нажмите кнопку «Сохранить». Служба USBMount будет запущена и добавлена в автозагрузку – состояние «Служба USBMount активна».

### 3.5.8.2.2. Список устройств

Если служба USBMount запущена, в веб-интерфейсе подраздела будет отображен список текущих подключенных устройств (рис. 28).

The screenshot shows the USBMount web interface. At the top, there is a toggle switch for 'Служба USBMount активна' (USBMount service active) and a 'Сохранить' (Save) button. Below this is the 'Список устройств' (List of devices) table with the following data:

N	Устройство	Файловая система	VID	PID	Серийный номер	Статус	Точка монтирования
1	/dev/sdb		090c	1000	2010121200000186	free	
2	/dev/sdb1	vfat	090c	1000	2010121200000186	free	

Below the table are buttons for 'Обновить список блочных устройств' (Refresh list of block devices) and 'Назначить владельца' (Assign owner). Underneath is the 'Список владельцев' (List of owners) table with the following structure:

<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа

At the bottom of the interface are buttons for 'Изменить' (Change), 'Сбросить' (Reset), 'Удалить' (Delete), 'Добавить' (Add), and 'Сохранить' (Save). A 'Журнал' (Log) button is also present in the bottom left corner.

Рис. 28 – Таблица «Список устройств»

В столбце «Статус» отображается текущее состояние устройства:

- «free» – владелец для устройства не назначен;
- «owned» – устройству назначен владелец.

Если устройству назначен владелец и устройство примонтировано, то текущая точка монтирования отображается в столбце «Точка монтирования».

Кнопка «Обновить список блочных устройств» позволяет обновить список подключенных устройств.

### 3.5.8.2.3. Добавление/удаление правил

Для того чтобы назначить права для подключенного блочного устройства, выполните следующие действия:

- выделите строку с нужным устройством в таблице «Список устройств» и нажмите кнопку «Назначить владельца» (или выполните двойной щелчок мыши по строке с устройством) (рис. 29). Правило будет добавлено в таблицу «Список владельцев» (рис. 30);



- в столбце «Пользователь» выбрать пользователя, в столбце «Группа» – группу владельца блочного устройства (рис. 31);
- нажмите кнопку «Сохранить». Статус устройства в таблице «Список устройств» изменится на «owned» (рис. 32).

Список устройств							
N	Устройство	Файловая система	VID	PID	Серийный номер	Статус	Точка монтирования
1	/dev/sdb		090c	1000	2010121200000186	free	
2	/dev/sdb1	vfat	090c	1000	2010121200000186	free	

Рис. 29 – Назначение владельца для подключенного устройства

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>		090c	1000	2010121200000186		

Введены некорректные значения, обратите внимание на ячейки, отмеченные красным

Рис. 30 – Правило добавлено в таблицу «Список владельцев»

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>		090c	1000	2010121200000186	user	user

Рис. 31 – Указание пользователя и группы для блочного устройства

Список устройств							
N	Устройство	Файловая система	VID	PID	Серийный номер	Статус	Точка монтирования
1	/dev/sdb		090c	1000	2010121200000186	owned	
2	/dev/sdb1	vfat	090c	1000	2010121200000186	owned	

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>	0	090c	1000	2010121200000186	user	user

Рис. 32 – Статус «owned» для устройства в таблице «Список устройств»

**Примечание.** При создании/редактировании правила некорректные значения будут выделены красным цветом, корректные – зеленым.

Чтобы назначить права для произвольного блочного устройства:

- нажмите кнопку «Добавить», расположенную под таблицей «Список владельцев». В таблицу будет добавлена пустая строка (рис. 33);
- в соответствующих столбцах укажите «VID», «PID» и «Серийный номер устройства» (рис. 34);
- в столбце «Пользователь» выберите пользователя, в столбце «Группа» – группу владельца блочного устройства (рис. 35);
- нажмите кнопку «Сохранить».

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>	0	090c	1000	2010121200000186	user	user
<input type="checkbox"/>						

Рис. 33 – Создание нового правила USBMount

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>	0	090c	1000	2010121200000186	user	user
<input type="checkbox"/>		346d	5678	FC0950FA3ADE4		

Изменить Сбросить Удалить **Добавить**

Сохранить

Введены некорректные значения, обратите внимание на ячейки, отмеченные красным

Рис. 34 – Параметры устройства

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>	0	090c	1000	2010121200000186	user	user
<input type="checkbox"/>		346d	5678	FC0950FA3ADE4	--	

Изменить Сбросить У

user  
test  
usbmount

Рис. 35 – Указание группы для блочного устройства

Для назначения прав для определенной группы пользователей, в столбце «Пользователь» следует выбрать прочерк (рис. 35).

Редактирование правила:

- двойным щелчком мыши по строке с правилом в таблице «Список владельцев» или выделите строку в таблице «Список владельцев» и нажмите кнопку «Изменить»;
- внесите изменения;
- нажмите кнопку «Сохранить».

Удаление правила:

- выделите строку(и) с правилом в таблице «Список владельцев»;
- нажмите кнопку «Удалить» (рис. 36);
- нажмите кнопку «Сохранить».

Список владельцев						
<input type="checkbox"/>	N	VID	PID	Серийный номер	Пользователь	Группа
<input type="checkbox"/>	0	090c	1000	2010121200000186	user	user
<input type="checkbox"/>	4	346d	6678	EC0950FA3ADE3	test2	usbmount

Рис. 36 – Удаление правила

**Примечание.** Для отмены внесенных изменений, до нажатия кнопки «Сохранить», следует нажать кнопку «Сбросить».

#### 3.5.8.2.4. Просмотр журнала аудита

Для просмотра журнала событий подключения/отключения USB-устройств нажмите кнопку «Журнал», расположенную в левом нижнем углу модуля. По нажатии на эту кнопку раскрывается журнал аудита (рис. 37).

Передвигаться по журналу к более старым или более новым записям можно с помощью кнопок «Назад» и «Вперед». Для того чтобы скрыть панель журнала, нажмите кнопку «Свернуть».

Фильтровать

```

[7] [2024-05-21 09:12:39.764] [usb-automount] [info] Polkit request for device (CanUserMount)/dev/sdb1
[8] [2024-05-21 09:12:39.773] [usb-automount] [info] daemon reply to polkit = NO
[9] [2024-05-21 09:13:52.941] [usb-automount] [info] Polkit request for device (CanUserMount)/dev/sdc2
[10] [2024-05-21 09:13:52.951] [usb-automount] [info] daemon reply to polkit = YES
[11] [2024-05-21 09:14:17.251] [usb-automount] [info] Polkit request for device (CanUserMount)/dev/sdb1
[12] [2024-05-21 09:14:17.260] [usb-automount] [info] daemon reply to polkit = NO
[13] [2024-05-21 09:14:48.676] [usb-automount] [info] Mount /dev/sdb1
[14] [2024-05-21 09:14:48.679] [usb-automount] [info] Created mount endpoint /media/alt-usb-mount/user_user/AC3F-3DC7
[15] [2024-05-21 09:14:48.679] [usb-automount] [info] Mount data =
uid=501,gid=501,fsmask=0007,dmask=0007,allow_utime=0020,codepage=866,shortname=mixed,utf8,flush,errors=remount-ro
[16] [2024-05-21 09:14:48.683] [usb-automount] [info] Mounted /dev/sdb1 to /media/alt-usb-mount/user_user/AC3F-3DC7
[17] [2024-05-21 09:14:48.683] [usb-automount] [info] Created mountpoint for /dev/sdb1 in the db
[18] [2024-05-21 09:33:45.405] [usb-automount] [info] Polkit request for device (CanUserUnMount)/dev/sdb1
[19] [2024-05-21 09:33:45.405] [usb-automount] [info] Daemon response to polkit = YES
[20] [2024-05-21 09:33:45.484] [usb-automount] [info] Removing empty /media/alt-usb-mount/user_user/AC3F-3DC7
[21] [2024-05-21 09:34:13.879] [usb-automount] [info] Mount /dev/sdb1
[22] [2024-05-21 09:34:13.881] [usb-automount] [info] Created mount endpoint /media/alt-usb-mount/user_user/AC3F-3DC7
[23] [2024-05-21 09:34:13.881] [usb-automount] [info] Mount data =
uid=501,gid=501,fsmask=0007,dmask=0007,allow_utime=0020,codepage=866,shortname=mixed,utf8,flush,errors=remount-ro
[24] [2024-05-21 09:34:13.887] [usb-automount] [info] Mounted /dev/sdb1 to /media/alt-usb-mount/user_user/AC3F-3DC7
[25] [2024-05-21 09:34:13.888] [usb-automount] [info] Created mountpoint for /dev/sdb1 in the db
[26] [2024-05-21 09:34:37.316] [usb-automount] [info] Polkit request for device (CanUserUnMount)/dev/sdb1
[27] [2024-05-21 09:34:37.316] [usb-automount] [info] Daemon response to polkit = YES
[28] [2024-05-21 09:34:37.403] [usb-automount] [info] Removing empty /media/alt-usb-mount/user_user/AC3F-3DC7

```

< Назад
|
Вперед >

Страница 1 из 2

Рис. 37 – Журнал аудита USBMount

### 3.5.9. Контроль доступа к портам

Модуль ЦУС «Контроль доступа к портам» (`alterator-port-access`) позволяет настроить ограничения на использование внешних носителей.

Должны быть установлены пакеты `alterator-fbi` и `alterator-ports-access`:

```
# apt-get install alterator-fbi  
# apt-get install alterator-ports-access
```

Запустить службу `ahttpd`:

```
# systemctl start ahttpd
```

Далее откройте в веб-браузере адрес ЦУС (пакет `alterator-fbi`) `https://localhost:8080` или `https://ip_address:8080` и введите пароль администратора. Для настройки ограничений на использование внешних носителей в меню «Система» выберите пункт «Контроль доступа к портам» (пакет `alterator-ports-access`) (рис. 38).

Для задания доступа к последовательным портам выполните следующие шаги:

- выбрать порт;
- разрешить или запретить доступ к выбранному порту;
- нажать кнопку «Сохранить настройки последовательного порта»;
- просмотреть раздел «Статус», чтобы убедиться в корректной работе настроек.

## КОНТРОЛЬ ДОСТУПА К ПОРТАМ

Настройка

Справка

Выйти

## Контроль последовательных портов

Порт	Включен	Режим доступа
------	---------	---------------

Настройки последовательного порта	Значение
Последовательный порт:	пожалуйста выберите из списка выше
Разрешен:	Да
Владелец:	
Группа:	
Режим доступа:	По умолчанию

Сохранить настройки последовательного порта

## Контроль USB-портов

## СПИСОК РАЗРЕШЕННЫХ УСТРОЙСТВ

Производитель	Устройство	Серийный №	Class	Subclass	Protocol	Режим доступа
---------------	------------	------------	-------	----------	----------	---------------

Удалить выбранное правило

Часть правила	Значение
ID производителя:	
ID продукта:	
Серийный номер:	
Device/interface class:	
Device/interface subclass:	
Device/interface protocol:	
Владелец:	
Группа:	
Режим доступа:	По умолчанию

Сохранить параметры USB устройства

Сброс

## ПОДКЛЮЧЁННЫЕ USB-УСТРОЙСТВА

 Show interfaces

Производитель	Устройство	Серийный №
---------------	------------	------------

Сканировать подключённые USB устройства

Разрешить выбранное устройство

## Статус

Контроль последовательных портов выключен **Включить контроль последовательных портов**Контроль USB-портов выключен **Включить контроль USB-портов**

Рис. 38 – Контроль доступа к портам

Для того чтобы отключить поддержку всех USB-устройств кроме заданных, нажмите на кнопку «Включить контроль USB-портов».

Для того чтобы добавить USB-устройство в список разрешенных, нужно указать один (или все) из идентификаторов: ID производителя, ID продукта, Серийный номер в соответствующее поле и нажать кнопку «Добавить правило».

Для определения подключенных USB-устройств нужно нажать кнопку «Сканировать USB-устройства», выделить устройство, которое необходимо разрешить, и нажать кнопку «Разрешить выбранное устройство» (рис. 39).

Для исключения устройства из списка разрешенных, нужно выделить правило, разрешающее данное устройство в таблице «Список разрешенных устройств» и нажать кнопку «Удалить выбранное правило».

Чтобы отключить поддержку всех USB-устройств кроме заданных, нужно нажать на кнопку «Включить контроль USB-портов».

**Примечание.** Перед активацией ограничений предварительно разрешите использование USB-портов для клавиатуры и мыши.

**Контроль USB-портов**  
СПИСОК РАЗРЕШЕННЫХ УСТРОЙСТВ

ID производителя	ID продукта	Серийный №	Информация	Режим доступа
13fe	4200	070879C4B9943F00		По умолчанию

Удалить выбранное правило

Часть правила	Значение
ID производителя:	
ID продукта:	
Серийный номер:	
Полезная информация об устройстве:	
Владелец:	
Группа:	
Режим доступа:	По умолчанию

Сохранить параметры USB устройства **Сброс**

**ПОДКЛЮЧЁННЫЕ USB-УСТРОЙСТВА**

Производитель	Устройство	Серийный №	ID производителя	ID продукта
		None	0458	0186
		070879C4B9943F00	13fe	4200

**Сканировать подключённые USB устройства** Разрешить выбранное устройство

Рис. 39 – Добавление USB-устройства в список разрешенных устройств

### 3.6. Средства контроля целостности

#### 3.6.1. Утилита `integalert`

`integalert` – это сканер проверки целостности файловой системы на основе `ossec` с оповещениями `syslog`, триггерами, отчетами о файлах журналов и ротацией, который легко настраивается и расширяется с помощью динамических подпрофилей.

Синтаксис команды `integalert`:

```
integalert [опции] [main|vm|container] [fix|check|configure]
```

Первый необязательный аргумент – подпрофиль. Если подпрофиль не указан, то неявно подразумевается подпрофиль `main`.

Примечание. Пакет `integalert-postgresql` предоставляет еще один профиль `integalert: postgresql`.

Второй необязательный аргумент – это действие, которое будет выполнено. Если этот аргумент не указан, то неявно предполагается проверка (`check`).

##### 3.6.1.1. Инициализация и конфигурация

Для инициализации подпрофиля используется команда:

```
integalert [подпрофиль] configure
```

Примечание. Если используется подпрофиль `main`, к именам файлов конфигурации и журналов не добавляется суффикс `_<SUBPROFILE>`.

После завершения настройки будут созданы следующие файлы конфигурации в каталоге `/etc/ossec/integalert[_<SUBPROFILE>]/`:

- `dirs.conf` – список каталогов, содержимое которых будет сканироваться, если используется действие `fix`. Содержимое этих каталогов также будет сканироваться и сравниваться с фиксированным состоянием, если используется действие `check`. Список каталогов, которые сканируются по умолчанию: `/bin, /sbin, /usr/bin, /usr/sbin`;
- `exclude.conf` – список шаблонов имен файлов, которые будут исключены из сканирования и проверки;



- `sender.conf` – конфигурация `sender` скрипта подпрофиля, в основном связанная с ведением журнала (функция ведения журнала и приоритет);
- `pipe.conf` – основной файл конфигурации подпрофиля (не рекомендуется вносить изменения в этот файл);
- `files.list` – альтернатива `dirs.conf` (см. п. 3.6.1.2).

В файлах `/etc/logrotate.d/integralert[_<SUBPROFILE>].conf` для каждого подпрофиля содержится начальный набор параметров ротации журналов.

Все файлы конфигурации, перечисленные выше, не перезаписываются.

Все начальные параметры конфигурации зависят, в свою очередь, от базовых параметров, указанных в файлах конфигурации `/etc/osec/integralert.conf` и `/etc/sysconfig/integralert` (считывается последним). Следует обратить внимание, что данные файлы конфигурации также прочитываются `sender`-скриптом (`/usr/lib/integralert/sender`).

#### 3.6.1.2. Параметр `FILELIST`

В качестве альтернативы указанию списка каталогов в файле `dirs.conf` можно определить список каталогов и отдельных файлов для проверки, используя файл конфигурации `files.list`. Список файлов может содержать шаблоны имен файлов, например:

```
/etc/lo*
```

Следует обратить внимание, что, если существует файл `files.list`, файлы `dirs.conf` и `exclude.conf` игнорируются.

#### 3.6.1.3. Фиксация состояния файловой системы

Чтобы исправить (и, при необходимости, инициализировать) подпрофиль, используется команда:

```
integralert [SUBPROFILE] fix
```

При выполнении этой команды все файлы из каталогов, перечисленных в `dirs.conf`, с исключениями, определенными в `exclude.conf`, сканируются, и их состояние фиксируется. Отчет записывается в лог-файл.

### 3.6.1.4. Проверка состояния файловой системы

Для проверки текущего состояния файлов, определенных подпрофилем, используется команда:

```
integalert [SUBPROFILE] [check]
```

При выполнении этой команды все файлы из каталогов, перечисленных в `dirs.conf`, с исключениями, определенными в `exclude.conf`, сканируются, и их состояние сравнивается с ранее зафиксированным состоянием.

Изменения, если таковые имеются, указываются в отчете (`lastlog[_<SUB-PROFILE>]`, и в `integalert[_<SUBPROFILE>].log`). Если состояние оказывается отличным от фиксированного, то предупреждающее сообщение записывается в стандартную ошибку и код возврата процесса не равен нулю.

Пример проверки:

- целостность не нарушена:

```
# integalert check
Integrity check OK.
```

- целостность нарушена:

```
# integalert check
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Integrity check failure!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Результат последней проверки в файле `/var/log/integalert/lastlog`:

```
===== Cp 13 ноя 2024 18:35:11 MSK =====
```

```
Processing /run/integalert/main/files.list ...
```

```
This is a report generated by osec at 'Wed Nov 13 18:35:11 MSK 2024'
```

```
New files added to control:
```

```
- /etc/logrotate.conf~
```

- /etc/logwatch/conf/logwatch.conf
- /etc/sysconfig/logwatch

Changed controlled files:

- /etc/logrotate.conf

inode: 262752 -> 264556

mtime: Wed May 24 19:12:21 2023 -> Wed Nov 13 18:35:06 2024

System integrity check (chg=1,add=3,del=0)

**Содержимое файла /var/log/integralert/lastlog после выполнения команды integralert fix:**

```
===== Cp 13 ноя 2024 18:35:43 MSK =====
```

```
Processing /run/integralert/main/files.list ...
```

```
This is a report generated by osec at 'Wed Nov 13 18:35:43 MSK 2024'
```

```
New files added to control:
```

- /etc/logrotate.conf~
- /etc/logwatch/conf/logwatch.conf
- /etc/sysconfig/logwatch

```
Changed controlled files:
```

- /etc/logrotate.conf

```
inode: 262752 -> 264556
```

```
mtime: Wed May 24 19:12:21 2023 -> Wed Nov 13 18:35:06 2024
```

```
System integrity fix (chg=1,add=3,del=0)
```

### 3.6.1.5. Триггеры

Если триггеры включены и состояние файловой системы отличается от фиксированного, `integralert` выполняет файлы сценариев, найденные в `/etc/osec/integralert[_<SUBPROFILE>]/trigger.d/`, передавая отчет на стандартный ввод каждого триггера. По умолчанию триггеры отключены. Для включения триггеров необходимо указать `ENABLE_TRIGGERS=1` в файле конфигурации `/etc/osec/integralert.conf` или `/etc/sysconfig/integralert`. Триггеры для определенного подпрофиля можно отключить с помощью файла `trigger.d/disable`.

### 3.6.2. Активация средств контроля целостности

Все действия производятся от имени администратора.

Для активации средств контроля целостности необходимо:

- 1) проверить, что сервис контроля целостности при загрузке активирован (enabled), выполнив команду:

```
# systemctl is-enabled integalert
```

Если сервис не активирован, активировать его, выполнив команду:

```
# systemctl enable integalert
```

- 2) определить каталоги для проверки целостности см. п. 3.6.2.1;

- 3) инициализировать подпрофиль, выполнив команду:

```
# integalert fix
```

Проверка целостности системы выполняется командой `integalert`.

Порядок восстановления целостности системы приведен в п. 3.6.2.2.

#### 3.6.2.1. Определение/изменение каталогов для проверки целостности

Для того чтобы определить или изменить список каталогов для проверки целостности, необходимо отредактировать список каталогов в файле `/etc/osec/integalert[_<SUBPROFILE>]/dirs.conf` (строки, начинающиеся со знака «#», игнорируются) или создать файл `/etc/osec/integalert[_<SUBPROFILE>]/files.list` со списком каталогов и отдельных файлов.

Например, в профиле `main` изменения в каталогах `/bin`, `/sbin`, `/usr/bin`, `/usr/sbin` отслеживаются по умолчанию. Если необходимо добавить или удалить каталог из списка отслеживаемых, необходимо проверить и отредактировать список каталогов, который хранится в файле `/etc/osec/integalert/dirs.conf`.

После внесения изменений в файл `dirs.conf` следует выполнить инициализацию профиля (см. п. 3.6.2 шаг 3).

#### 3.6.2.2. Порядок восстановления целостности системы

Нарушение целостности системы может быть выявлено:

- 1) при проверке целостности системы командой `integalert`;

2) при загрузке ОС – вывод визуального оповещения с обязательным подтверждением дальнейшей загрузки ОС. Система загружается в однопользовательском режиме (если активирован однопользовательский режим см. п. 3.6.2.3) с запросом пароля суперпользователя:

```
login [root]:
```

```
Password:
```

Порядок восстановления:

- от администратора (root) выполнить просмотр лога аудита и найти нарушение целостности (см. п. 3.6.2.4);
- восстановить объект с нарушенной целостностью – заменить на эталон/переустановить.

Порядок фиксирования нового состояния:

- обновить контролируемые файлы;
- запустить процедуру фиксирования нового состояния системы (если система контроля целостности IMA/EVM не инициализирована, иначе см. п. 3.6.3.4), выполнив команду:

```
# integalert fix
```

При этом будут сформированы и занесены в базу данных контрольные суммы для измененных/новых объектов.

Если система была загружена в однопользовательском режиме можно продолжить загрузку в обычном режиме, завершив однопользовательский сеанс (например, выполнив команду: `systemctl default`).

### 3.6.2.3. Автоматические действия при нарушении целостности при загрузке системы

Если необходимо автоматически выполнять определенные действия – включать и выключать службы, переводить в систему в определенный режим, если при загрузке обнаруживается нарушение целостности, следует связать требуемые действия с целью `integ-check-failed.target`. Для этого следует использовать тег

WantedBy= integ-check-failed.target либо использовать символические ссылки (подробнее см. systemd.unit(5)).

Например, для того, чтобы ОС при обнаружении нарушения целостности системы переходила в однопользовательский режим (режим emergency), следует:

- 1) создать каталог /etc/systemd/system/integ-check-failed.target.wants:
 

```
# mkdir /etc/systemd/system/integ-check-failed.target.wants
```
- 2) связать integ-check-failed.target с emergency.target используя символическую ссылку:
 

```
# ln -s /lib/systemd/system/emergency.target \
/etc/systemd/system/integ-check-failed.target.wants/
```
- 3) перезагрузить конфигурацию systemd:
 

```
# systemctl daemon-reload
```

С этими настройками при обнаружении нарушения целостности система будет загружена в однопользовательском режиме с запросом пароля root:

```
login [root]:
Password:
```

После входа от администратора необходимо выполнить просмотр журнала аудита для получения информации о нарушении целостности (см. п. 3.6.2.4), восстановить целостность системы (см. п. 3.6.2.2). После завершения однопользовательского сеанса, система продолжит загрузку в обычном режиме.

#### 3.6.2.4. Отчеты integalert

Просмотреть записи integalert в системном журнале можно, выполнив команду:

```
# journalctl | grep -E 'integalert'
```

При запусках системы, в случае, если в системе после предыдущего запуска integalert ничего не менялось, в отчете будет указано, что нет никаких изменений, например:

```
ноя 13 18:57:08 host-193 osec:integalert[2400]: Started
ноя 13 18:57:09 host-193 integalert[2420]: System integrity check
(chg=0,add=0,del=0)
ноя 13 18:57:09 host-193 osec:integalert[2423]: Finished
ноя 13 18:57:09 host-193 integalert[2428]: Integrity check OK
System integrity check (chg=0,add=0,del=0)
```

В случае, если в системе произошли изменения, то в журнале, содержится отчет об измененных файлах:

```
ноя 13 18:55:10 host-193 osec:integralert[2534]: Started
ноя 13 18:55:11 host-193 integralert[2557]: System integrity check
(chg=1,add=1,del=0)
ноя 13 18:55:11 host-193 osec:integralert[2560]: Finished
ноя 13 18:55:11 host-193 integralert[2565]: Integrity check
failure!
ноя 13 18:55:11 host-193 systemd[1]: integralert.service: Main
process exited, code=exited, status=1/FAILURE
```

Для просмотра отчета об изменениях и их количестве, найденных при последней проверке, можно воспользоваться командой:

```
# cat /var/log/integralert/lastlog
```

### 3.6.3. Подсистема IMA/EVM

#### 3.6.3.1. Утилиты integrity-applier и integrity-remover

Утилита integrity-applier используется для включения подсистем IMA и EVM, а также для добавления ключей в набор ключей ядра.

Синтаксис команды integrity-applier:

```
integrity-applier [ --init ] [ --auto ]
integrity-applier [ --sign ]
integrity-applier [ -v | --verbose ] [ --hash=HASH ] [ -E | --with-evm ]
[ --without-evm ] [ -R | --no-reboot ]
```

**# этап I:**

```
integrity-applier -i | --init [ -G | --disable-graphics ] [ -A | --auto ]
```

**# этап II:**

```
integrity-applier -s | --sign [ -I | --skip-initrd ] [ --log FILE ]
```

**# Other:**

```
integrity-applier --initrd-only
integrity-applier -h | --help
integrity-applier -V | --version
```

Параметры --init и --sign используются, для того чтобы явно указать, что следует выбрать этап I или этап II инициализации IMA.

При запуске команды `integrity-applier` с параметром `--init` (или без параметров, в случае если в параметрах ядра нет значения `ima_appraise=fix`) будет выполнен этап I инициализации подсистемы IMA/EVM. По окончании этого этапа система будет перезагружена на так называемый этап II с опцией ядра `ima_appraise=fix`.

При запуске команды `integrity-applier` с параметром `--sign` (или без параметров, в случае если в опциях ядра есть значение `ima_appraise=fix`) будут подписаны файлы. Затем система будет перезагружена в режим принудительного использования IMA (т. е. с параметром ядра `ima_appraise=enforce`).

**Примечание.** Для проверки параметров ядра можно выполнить команду:  
`# cat /proc/cmdline`

Параметры:

- 1) `-R` | `--no-reboot` – отключить автоматическую перезагрузку системы;
- 2) `-a HASH` | `--hash=HASH` – указать тип криптографической хеш-функции (SHA или ГОСТ) для подписи файлов. Алгоритм хеширования также можно настроить в файле `/etc/integrity/config` и (или) `/etc/sysconfig/integrity`;
- 3) `-E` | `--with-evm` – включить EVM;
- 4) `--without-evm` – не включать EVM;
- 5) `-G` | `--disable-graphics` – не запускать графическую подсистему на этапе II;
- 6) `-A` | `--auto` – автоматически запустить этап II после перезагрузки с помощью специального `ima-signing.target`;
- 7) `--log=FILE` – выводить журнал подписи файлов в указанный файл на этапе II. По умолчанию журнал подписи файлов записывается в `/var/log/integrity-sign.log`;
- 8) `--log-stderr` – выводить журнал подписи файлов в текущий поток ошибок (если используется опция `--log-stderr`, то опция `--log` игнорируется);



9) `-I | --skip-initrd` – не пересоздавать `initrd`.

**Примечание.** Если использовалась опция `--skip-initrd` может потребоваться повторно создать образ `initrd`:

```
# integrity-applier --initrd-only
```

В файлах `/etc/integrity/config` и `/etc/sysconfig/integrity`, настраиваются каталоги с файлами для подписи и другие параметры, например, EVM. Содержимое `/etc/sysconfig/integrity` переопределяет значения, заданные в `/etc/integrity/config`.

Для проверки политики IMA при загрузке системы необходимо включить службу `ima-check.service`.

В `ima-check-failed.target` можно добавить службы, для запуска действий, если проверка не удалась.

Чтобы вообще не загружать систему, если загрузка политики IMA не удалась на этапе `initrd`, следует создать файл `/etc/integrity/reboot-on-initrd-error`. Скрипт `/etc/integrity/on-initrd-error` можно использовать, чтобы запустить его вместо перезагрузки.

---

⚠ **Файлы** `/etc/integrity/reboot-on-initrd-error` и `/etc/integrity/on-initrd-error` нужно создать до запуска `integrity-applier` так как они должны попасть в `initrd`.

---

Переопределить политику целостности по умолчанию можно с помощью `/etc/integrity/policy`.

Утилита `integrity-remover` используется для отключения подсистем IMA и EVM.

Синтаксис команды `integrity-remover`:

```
integrity-remover
```

```
integrity-remover -h | --help
```

```
integrity-remover -V | --version
```

Для отключения IMA следует запустить `integrity-remover`.

После завершения работы скрипта система будет перезагружена.

### 3.6.3.2. Служба оповещения – integrity-notifier

Служба integrity-notifier – оповещает пользователя в том случае, если была запущена поврежденная программа.

**Примечание.** Служба integrity-notifier должна запускаться при загрузке системы.

Сообщения из системного журнала с пометкой INTEGRITY\_DATA фильтруются службой integrity-scanner и складываются в каталог /var/log/integrityd/. В файле /var/log/integrityd/current поддерживается окно сообщений заданного размера (по умолчанию 300 Кбайт) за установленный интервал времени (по умолчанию 5 минут). Старые записи складываются там же, с именами, начинающимися на "@" (по умолчанию 50 файлов максимум). Настройки по умолчанию можно изменить в файле /var/log/integrityd/config.

Минимальный UID пользователя для отправки оповещений по умолчанию равен 500.

В файле /etc/integrity/notify.conf можно изменить настройки по умолчанию:

- 1) PATTERN – пометка, сообщения с которой будут фильтроваться службой integrity-scanner в системном журнале;
- 2) LOGDIR – каталог для файлов журнала;
- 3) LOGCONF – определяет конфигурацию для управления размером и количеством файлов журнала. По умолчанию «s307200 n50», где:
  - s – размер журнала (307200 байт);
  - n – количество файлов журнала, которые будут сохранены (50);
  - MIN\_UID – минимальный UID пользователя для отправки оповещений.

Служба оповещения (integrity-notifier) читает свежие сообщения из окна и оповещает соответствующих пользователей посредством команды write.

Сообщение, которое отправляется пользователю, настраивается в файле /etc/integrity/message. Это должен быть шаблон printf с количеством строковых аргументов не более двух штук. По умолчанию используется строка:

```
You have attempted to run a damaged file: %s (%s)
```

Конфигурационный файл `/etc/integrity/also` состоит из набора имен тех пользователей, которых нужно оповещать о каждом событии `INTEGRITY_DATA`. По умолчанию этот файл пустой (в нем записан администратор (`root`)).

Предусмотрена программа для отправки оповещений на рабочий стол – `/usr/bin/integrity-notifier`. Она использует команду `notify-send`.

Текст заголовка и сообщения для оповещений на рабочий стол задается в файле `/etc/integrity/desktop_message` (рис. 40).

Могут быть и другие службы оповещения пользовательского уровня, также читающие сообщения из файла `/var/log/integrityd/current`.

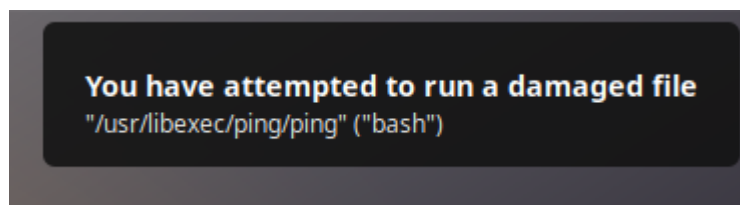


Рис. 40 – Оповещение пользователя

### 3.6.3.3. Настройка контроля целостности

Для настройки контроля целостности необходимо выполнить следующие действия:

- 1) изменить параметры монтирования файловой системы. Для этого следует выставить параметр `iversion` на всех записях в файле `/etc/fstab` относящихся к местам, где могут быть исполняемые файлы:

```
# vim /etc/fstab
UUID=c7834d14-d0f0-4d70-94f5-f1ce09fda00c / ext4 relatime,iversion 1 1
UUID=16b090bf-8b7a-4e69-8df6-6a4374f3d550 /home ext4 noexec,nosuid,relatime,iversion 1 2
```

- 2) настроить контроль целостности. Желаемый алгоритм хеширования (SHA или ГОСТ) можно выбрать в файле `/etc/integrity/config`. Там же можно включить EVM в дополнение к IMA. Для изменения алгоритма хеширования необходимо изменить значение переменной `HASH_ALGO` в файле `/etc/integrity/config`, например, заменить `HASH_ALGO="sha512"` на `HASH_ALGO="streebog512"`.

Если необходимо выполнить активацию EVM, то следует раскомментировать строку: `#WITH_EVM=yes`

- 3) запустить инициализацию подсистем контроля целостности в автоматическом режиме:

```
# integrity-applier --auto
```

Будет запущен первый этап инициализации подсистемы IMA/EVM, затем система будет перезагружена и будет запущен второй этап инициализации IMA/EVM, после завершения которого система будет перезапущена еще раз.

**Примечание.** Если есть необходимость между этапами инициализации системы контроля целостности выполнить какие-либо действия, можно запустить инициализацию в два этапа. Для запуска первого этапа выполните команду:

```
# integrity-applier -i
```

После выполнения данной команды система будет перезагружена.

После перезагрузки системы следует выполнить авторизацию в терминале и получить права суперпользователя (root) и запустить второй этап инициализации подсистемы IMA/EVM, выполнив команду:

```
# integrity-applier -sv
```

В результате выполнения данной команды, система будет перезагружена еще раз;

- 4) после перезагрузки системы войти в систему и проверить параметры командной строки ядра:

```
# cat /proc/cmdline
```

```
BOOT_IMAGE=/boot/vmlinuz root=UUID=19124d73-ad8b-4b83-ba28-350312a1b7f7 ro panic=30 init_on_free=1 loglevel=3 splash  
lsm=integrity quiet ima_hash=streebog512 ima_appraise=enforce
```

в командной строке ядра должен присутствовать параметр `ima_appraise=enforce` – система находится в режиме оценки контроля целостности;

- 5) добавить в автозагрузку и включить службы:

```
# systemctl enable --now integrity-scanner.service integrity-  
notifier.service ima-check.service
```

Система будет работать с включенным контролем исполняемых файлов.

Перед запуском программы система проверяет хеш-образ файла с сохраненным значением. Если образы не совпадают, то любой доступ к этому файлу будет отклонен с ошибкой «Отказано в доступе».

**Примечание.** Добавление опции монтирования `iversion` позволяет оптимизировать ИМА: хеш файла будет пересчитываться только тогда, когда он фактически изменяется, а не каждый раз, когда файл открывается.

**Примечание.** Проверить наличие маркировки у файла можно, выполнив команду:

```
# getfattr -m - -d /usr/bin/ping
getfattr: Removing leading '/' from absolute path names
# file: usr/bin/ping
security.ima=0sAwITlfpL+wCANOWZZTZnDBbNJDysg7B3R5j1orvmFyk2Fpfw0F
YPBiTZRwUuAARPLD+pOJouUTUbKI f0JGppJr8y7uw6EpFbkkkgBy4HzdFYgPFI1TeD
n/jJtuXFoNnxxwllLm5d2Ag6UthfdIlrHIWCYLy+/oDCo/b5I3SammkhBuZKO4Zj1
s/k=
```

Строка `security.ima=...` свидетельствует о том, что файл маркирован.

В журнале записи, связанные с нарушением целостности можно просмотреть командой:

```
# journalctl -r | grep INTEGRITY_DATA
```

**Пример вывода:**

```
ноя 05 13:51:40 sp.test.alt audit[3823]: INTEGRITY_DATA pid=3823
uid=0 auid=500 ses=3 op=appraise_data cause=invalid-signature
comm="bash" name="/usr/libexec/ping/ping" dev="sda2" ino=1748147
res=0 errno=0
```

В случае выявления нарушения целостности при загрузке ОС выводится визуальное оповещение с обязательным подтверждением дальнейшей загрузки ОС.

Если активирован однопользовательский режим (см. п. 3.6.2.3) система будет загружаться с запросом пароля суперпользователя:

```
login [root]:
Password:
```

Порядок восстановления см. в п. 3.6.3.4.

### 3.6.3.4. Восстановление работы целостности системы с IMA/EVM

Для восстановления работы системы необходимо:

1) загрузить систему с `ima_appraise=fix` – в файле `/boot/boot.conf` в значении переменной `CMDLINE` изменить значение параметра загрузки ядра `ima_appraise`. Перезагрузить ОС;

2) войти в ОС и от администратора (`root`) выполнить команду просмотра лога аудита и найти нарушение целостности, например:

```
# cat /var/log/integrityd/current
@400000006735d1f72c026d14 ноя 14 13:33:17 host-191 audit[3447]:
INTEGRITY_DATA pid=3447 uid=0 auid=500 ses=1 op=appraise_data
cause=invalid-signature comm="bash"
name="/usr/libexec/ping/ping" dev="sda2" ino=546186 res=0
errno=0
```


3) восстановить файл с нарушенной целостностью – заменить на эталон/переустановить;

4) после выполнения настроек и восстановления повреждения запустить инициализацию системы контроля целостности IMA/EVM – выполнить шаг 4) п. 3.6.3.3.

**Примечание.** При активации однопользовательского режима будет осуществляться четыре перезагрузки, ничего не вводить при запросе:

```
login [root]:
```

---

 Необходимо дождаться завершения работы команды, выполнение может занять довольно продолжительное время (подписываются все файлы системы).

---

### 3.6.3.5. Обновление системы с IMA/EVM

Для обновления системы необходимо:

1) загрузить систему с `ima_appraise=fix` – в файле `/boot/boot.conf` в значении переменной `CMDLINE` изменить значение параметра загрузки ядра `ima_appraise`. Перезагрузить ОС;

2) обновить систему до актуального состояния в соответствии документом «Руководство администратора. ЛКНВ.11102-01 90 01»;

- 3) запустить инициализацию подсистем контроля целостности (переподписать) автоматически или в два этапа (см. п. 3.6.3.3).

По завершении указанных настроек система будет перезапущена.

### 3.6.4. Проверка хешей установленных пакетов rpm

#### 3.6.4.1. Утилита verify-checksums

Утилита verify-checksums предназначена для проверки установленных пакетов rpm.

Синтаксис:

```
verify-checksums [-h] [-f] [--dir <каталог>] [--alg <алгоритм>] [--R <каталог>]
```

Опции:

- 1) -h, --help – показать справочное сообщение и выйти;
- 2) -f, --files – отображать файлы с поврежденным хешем;
- 3) --dir <path/to/catalog> – каталог, содержащий файлы .chksum. Опция может быть указана несколько раз, для указания нескольких каталогов;
- 4) --alg <алгоритм> – алгоритм для хеша. Возможные значения: sha256sum, gost12sum (по умолчанию), md5sum;
- 5) --R <path/to/rescue> – смонтированный путь восстановления с базой данных RPM и файлами .chksum (каталог восстановления для обработки файлов). Опция --R не может использоваться одновременно с опцией --dir. В опции --R нельзя использовать больше чем один аргумент. Опция --R может быть указана только один раз.

Для запуска утилиты ей необходимо указать путь к каталогу (/chksum), содержащему файлы в формате .chksum. Эти файлы должны содержать информацию из исходных пакетов rpm.

Пофайловые отчеты подсчета контрольных сумм неизменяемых файлов дистрибутива, представлены на установочном компакт-диске каждого варианта исполнения соответствующей процессорной архитектуры в корневой директории /chksum.

Для успешной проверки файлов программа должна запускаться с правами суперпользователя.

Во время выполнения программы создаются:

- checksum-verification – каталог, содержащий результаты выполнения;
- report-rpm-found – файл, содержащий список найденных пакетов;
- report-rpm-not-found – файл, содержащий список не найденных пакетов;
- report-files-check-ok – файл, содержащий список успешно проверенных ELF-файлов;
- report-files-check-error – файл, содержащий список непроверенных ELF-файлов;
- log – журнал работы утилиты.

Примеры:

- выполнить проверку подлинности файлов ОС:

```
# verify-checksums --dir /media/ALTlinux/chksum/c10f2/ --dir
/media/ALTlinux/chksum/c10f2_e2k/noarch/
```

Задаем директории для проверки:

```
'/media/ALTlinux/chksum/c10f2_e2k/e2kv4/'
'/media/ALTlinux/chksum/c10f2_e2k/noarch/'
```

Создаем директорию: /root/checksum-verification

Processing:

```
/media/ALTlinux/chksum/c10f2_e2k/e2kv4/chksum/M4PX37Y.e2kv4.chksum (PACKAGES:
32|FILES: 0):
```

Processing:

```
/media/ALTlinux/chksum/c10f2_e2k/e2kv4/chksum/M4VDE7Y.e2kv4.chksum (PACKAGES:
11|FILES: 1):
```

...

Processing:

```
/media/ALTlinux/chksum/c10f2_e2k/noarch/chksum/M4CS7KQ.noarch.chksum
(PACKAGES: 1018|FILES: 30):
```

Processing:

```
/media/ALTlinux/chksum/c10f2_e2k/noarch/chksum/M4P46FI.noarch.chksum
```

Processing:

```
/media/ALTlinux/chksum/c10f2_e2k/noarch/chksum/M4J7JYY.noarch.chksum
(PACKAGES: 25|FILES: 0):
```

\*\*Количество проверенных пакетов: 1850\*\*

Отчет: /root/checksum-verification/report-rpm-found

\*\*Количество отсутствующих пакетов: 2131\*\*



```

Отчет: /root/checksum-verification/report-rpm-not-found
**Общее количество проверенных неизменяемых файлов: 12985**
Отчет: /root/checksum-verification/report-files-check-ok
**Количество неизменяемых файлов с нарушенной целостностью: 1**
Отчет: /root/checksum-verification/report-files-check-error

```

Найдены ошибки: проверьте '/root/checksum-verification/report-files-check-error'

**- выполнить проверку и отобразить файлы с поврежденным хешем:**

```
# verify-checksums --dir c10f2_e2k/e2kv4/ -f
```

```

**Количество проверенных пакетов: 1442**
Отчет: /root/checksum-verification/report-rpm-found
**Количество отсутствующих пакетов: 1387**
Отчет: /root/checksum-verification/report-rpm-not-found
**Общее количество проверенных неизменяемых файлов: 12955**
Отчет: /root/checksum-verification/report-files-check-ok
**Количество неизменяемых файлов с нарушенной целостностью: 1**
Отчет: /root/checksum-verification/report-files-check-error

```

Найдены ошибки: проверьте '/root/checksum-verification/report-files-check-error'

```

Файлы с нарушенной целостностью:
/usr/lib/nagios/plugins/check_ping

```

**- выполнить проверку, используя алгоритм md5sum:**

```
# verify-checksums --dir /media/ALTLinux/chksum/c10f2_e2k/e2kv4/ --alg md5sum
```

### 3.6.4.2. Утилита alt-chksum

Утилита alt-chksum проверяет контрольные суммы файлов, вызывая verify-checksums.

Синтаксис:

```
alt-chksum <command> ...
```

Возможные команды утилиты alt-chksum представлены в таблице 2.

**Т а б л и ц а 2 – Команды утилиты alt-chksum**

Команда	Параметры
alt-chksum url [?\>   <n>   https://...]	<ul style="list-style-type: none"> <li>- \? – отобразить предварительно настроенный пронумерованный набор URL-адресов;</li> <li>- &lt;n&gt; – выбрать URL-адрес с указанным номером;</li> <li>- https:// – выбрать указанный</li> </ul>

Команда	Параметры
	(пользовательский) URL-адрес
<code>alt-chksum add \?   &lt;branch&gt;</code>	- \? – отобразить список веток, доступных для выбранного URL; - <branch> – загрузить и извлечь ветку с указанным именем
<code>alt-chksum show \?   &lt;branch&gt;</code>	- \? – вывести список веток, которые в данный момент проверены; - <branch> – отобразить подробную информацию о ветке
<code>alt-chksum log \?   &lt;branch&gt;</code>	- \? – вывести список текущих извлеченных веток; - <branch> – отобразить журнал указанной ветки
<code>alt-chksum del \?   &lt;branch&gt;</code>	- \? – перечислить текущие извлеченные ветки; - <branch> – удалить локальную копию ветки
<code>alt-chksum update [\?   &lt;branch&gt;]</code>	- \? – перечислить текущие извлеченные ветки; - <branch> – загрузить обновления для указанной ветки; - без параметров – загрузить обновления для всех текущих извлеченных веток
<code>alt-chksum validate [ \?   &lt;branch&gt;]</code>	- \? – перечислить текущие извлеченные ветки; - <branch> – проверить подпись данной ветки; - без параметров – проверить подпись для всех текущих извлеченных ветвей
<code>alt-chksum verify [параметры]</code>	- проверить контрольные суммы. Все параметры передаются в скрипт <code>verify-checksum(1)</code>

*Окончание таблицы 2*

Команда	Параметры
<code>alt-chksum keys [\?]</code>	- \? – отобразить метаданные доступных открытых ключей; - без параметров – вывести данные ключей как есть
<code>alt-chksum init [path/to/dir...]</code>	- <code>path/to/dir...</code> – инициализирует указанный каталог для данных контрольной суммы. Если каталог не указан, инициализирует текущий каталог
<code>alt-chksum help [&lt;команда&gt;]</code>	Позволяет вывести справку по команде
<code>alt-chksum version</code>	Позволяет вывести версию утилиты

## Примеры:

- отобразить предварительно настроенный пронумерованный набор URL-адресов:

```
# alt-chksum url \?
1) https://checksum.altsp.su/alt-checksum/checksums.git
```

- отобразить список веток для выбранного URL:

```
# alt-chksum add \?
base
c10f2/aarch64
c10f2/noarch
c10f2/x86_64
c10f2/x86_64-i586
c10f2_e2k/e2kv4
c10f2_e2k/e2kv6
c10f2_e2k/noarch
```

- добавить контрольные суммы, например, c10f2\_e2k/e2kv4:

```
# alt-chksum add c10f2_e2k/e2kv4
```

- посмотреть список веток, которые в данный момент проверены:

```
# alt-chksum show \?
c10f2_e2k/e2kv4
```

- проверить контрольные суммы:

```
# alt-chksum verify
...
**Количество проверенных пакетов: 1443**
Отчет: /root/checksum-verification/report-rpm-found
**Количество отсутствующих пакетов: 1386**
Отчет: /root/checksum-verification/report-rpm-not-found
**Общее количество проверенных неизменяемых файлов: 12958**
Отчет: /root/checksum-verification/report-files-check-ok
**Количество неизменяемых файлов с нарушенной целостностью: 0**
Отчет: /root/checksum-verification/report-files-check-error

Проверка завершилась успешно
```

### 3.6.5. Интеграция Icinga

Нарушение контроля целостности системы фиксируется nagwad.service (см. подраздел «Настройка системы сигнализации на основе icinga» в документе «Руководство администратора. ЛКНВ.11102-01 90 01»).

### 3.7. Средства резервного копирования

#### 3.7.1. Резервное копирование при помощи утилиты rsync

##### 3.7.1.1. Общие сведения

Rsync (англ. Remote Synchronization) – программа, которая выполняет синхронизацию файлов и каталогов в двух местах с минимизированием трафика, используя кодирование данных при необходимости. Важным отличием rsync от многих других программ/протоколов является то, что зеркалирование осуществляется одним потоком в каждом направлении (а не по одному или несколько потоков на каждый файл).

Rsync может копировать или отображать содержимое каталога и копировать файлы, опционально используя сжатие и рекурсию.

Утилита rsync использует протокол удаленного обновления (remote-update protocol) для значительного ускорения передачи файлов, которые уже существуют в месте назначения. Благодаря этому протоколу rsync передает только различия между двумя наборами файлов через сетевое соединение, используя эффективный алгоритм поиска контрольных сумм.

Дополнительные особенности rsync:

- поддержка копирования ссылок, файлов устройств, разрешений и атрибутов владельца и группы;
- параметры исключения путей exclude и exclude-from;
- может прозрачно использовать любую оболочку удаленного доступа, включая rsh или ssh;
- не нуждается в привилегиях суперпользователя root;
- используется конвейеризация передачи файлов для уменьшения задержек;
- поддержка анонимного сервера rsync или сервера rsync с аутентификацией (идеально для зеркалирования).

Существует восемь способов использования rsync:

- локальное копирование файлов;

- копирование локальных файлов на удаленный хост, используя программу удаленной оболочки в качестве транспорта (например, rsh или ssh);
- копирование с удаленного хоста в локальные файлы, используя программу удаленной оболочки;
- копирование с удаленного rsync-сервера в локальные файлы;
- копирование локальных файлов на удаленный rsync-сервер;
- копирование с удаленной машины с использованием удаленной оболочки как транспорта и удаленного rsync-сервера;
- копирование с локальной машины на удаленную с использованием удаленной оболочки как транспорта и удаленного rsync-сервера;
- получение списка файлов на удаленной машине.

Во всех случаях (кроме запроса списка файлов) как минимум один путь (либо исходный SRC, либо конечный DEST) должен быть локальным.

Rsync должна быть установлена на обоих хостах, которые вовлечены в операцию копирования.

Синтаксис:

```
rsync [OPTION]... SRC [SRC]... DEST
```

Основные опции:

- 1) `-n` не создавать дочерний процесс. Для запуска из `inittab`;
- 2) `-v`, `--verbose` увеличить уровень подробностей;
- 3) `-q`, `--quiet` уменьшить уровень подробностей;
- 4) `-c`, `--checksum` проверять контрольные суммы;
- 5) `-a`, `--archive` архивный режим, эквивалент для `-rlptgD`;
- 6) `-r`, `--recursive` рекурсивно входить в подкаталоги;
- 7) `-R`, `--relative` использовать относительные пути;
- 8) `-b`, `--backup` создавать резервную копию;
- 9) `--backup-dir` создавать резервную копию в этом каталоге;
- 10) `--suffix=SUFFIX` суффикс для резервной копии (по умолчанию `~` в отсутствие `--backup-dir`);

- 11) `-u, --update` только обновление (не переписывает более новые файлы);
- 12) `-l, --links` копировать символичные ссылки как символичные ссылки;
- 13) `-H, --hard-links` сохранять жесткие ссылки;
- 14) `-p, --perms` сохранять разрешения;
- 15) `-o, --owner` сохранять владельца (только root);
- 16) `-g, --group` сохранять группу;
- 17) `-D, --devices` сохранять файлы устройств (только root);
- 18) `-t, --times` сохранять время;
- 19) `-R, --relative` использовать относительные пути;
- 20) `-X, --xattrs` сохраняют расширенные атрибуты;
- 21) `--existing` обновить только те файлы, которые уже существуют;
- 22) `--ignore-existing` пропускать те файлы, которые уже существуют на приемной стороне;
- 23) `--delete` удалять файлы, которых нет на передающей стороне;
- 24) `--delete-excluded` также удалять те файлы, которые исключены из списка копирования;
- 25) `-z, --compress` сжимать поток передачи данных;
- 26) `--exclude=PATTERN` исключить файлы, соответствующие шаблону PATTERN;
- 27) `--exclude-from=FILE` шаблоны исключения брать из файла FILE;
- 28) `-h, --help` показать помощь.

Настроить резервное копирование при помощи `rsync` (с сохранением атрибутов файлов) можно для файловых систем типа `ext3-4`, `afs`.

#### Примеры:

1. Копирование всех файлы рекурсивно из каталога `src/bar` в каталог `/data/tmp/bar`. Передача файлов происходит в «архивном» режиме, который гарантирует сохранение символических ссылок, файлов-устройств, атрибутов, разрешений и т. д. Кроме того, используется сжатие для уменьшения объема непосредственно передаваемых данных:

```
rsync -avz src/bar /data/tmp
```

2. Копирование всех файлов по шаблону \*.c из текущего каталога в каталог src, если какой-либо из файлов уже существует, rsync использует протокол удаленного обновления для передачи только различий:

```
rsync *.c src/
```

### 3.7.2. Пример настройки системы резервного копирования данных

В качестве примера настроим локальное резервное копирование содержимого папки /home, исключая временные файлы. Резервное копирование должно быть инкрементальным, копии старше 30 дней при этом должны удаляться.

Для настройки системы резервного копирования данных необходимо выполнить следующие действия:

- 1) создать файл /etc/backup.lst со списком каталогов для резервного копирования (один каталог в одной строке):

```
# touch /etc/backup.lst
# echo "/home">> /etc/backup.lst
```

- 2) создать файл /etc/backup-exclude.lst с масками пропускаемых файлов:

```
# touch /etc/backup-exclude.lst
# echo "*.tmp">> /etc/backup-exclude.lst
# echo "~*">> /etc/backup-exclude.lst
```

- 3) создать каталог для резервных копий /var/backup:

```
# mkdir /var/backup
```

- 4) создать скрипт для инкрементального архивирования

/usr/share/backup.sh со следующим содержимым:

```
#!/bin/sh
# Настройки переменных скрипта
BACKUPLIST=/etc/backup.lst
EXCLUDES=/etc/backup-exclude.lst
ARCHIVEROOT=/var/backup
# Каталог, в котором хранится текущая копия файлов
CURRENT=latest
# Каталоги, в которых хранятся резервные копии по дате архива
DATEDIR=Date-`date +%F--%H-%M`
# Опции, которые передаются rsync
OPTIONS="--force \
--ignore-errors \
--delete \
--delete-excluded \
--exclude-from=$EXCLUDES \
```

```

--backup \
-aqRSX"

export PATH=$PATH:/bin:/usr/bin:/usr/local/bin
export LANG=ru_RU.KOI8-R
# функция архивирования
do_rsync()
{
cat ${BACKUPLIST} | while read BACKUPDIR; do
rsync $OPTIONS $BACKUPDIR $ARCHIVEROOT/$CURRENT
done
}
# Удаление старых резервных копий
do_clear()
{
find $ARCHIVEROOT/ -maxdepth 1 -name "Date-*" -mtime +30 -exec
rm -Rf {} \;
}

# Сохранение старых резервных копий
do_link()
{
cp --archive --link $ARCHIVEROOT/$CURRENT
$ARCHIVEROOT/$DATEDIR
}

if [ -f $EXCLUDES ]; then
if [ -d $BACKUPDIR ]; then
do_rsync && do_clear && do_link
else
echo "не найден $BACKUPDIR"; exit
fi
else
echo "не найден $EXCLUDES"; exit
fi

```

5) добавить права на запуск файлу /usr/share/backup.sh:

```
# chmod +x /usr/share/backup.sh
```

Теперь, в случае необходимости, можно в любой момент создать резервную копию, запустив скрипт /usr/share/backup.sh:

```
# /usr/share/backup.sh
```

В папке /var/backup будет создан новый каталог Date-2024-02-14--16-50 (текущая дата – текущее время) с резервной копией.



Для создания резервной копии в автоматическом режиме необходимо выполнить следующие действия:

- 1) создать в `/etc/systemd/system` файл `timebackup.service`, со следующим содержимым:

```
[Unit]
Description=Timed run backup
[Service]
ExecStart=/usr/share/backup.sh
```

- 2) создать в `/etc/systemd/system` файл `timebackup.timer`, со следующим содержимым:

```
[Unit]
Description= run every day in 16:50:40

[Timer]
OnCalendar=16:50:40

[Install]
WantedBy=multi-user.target
```

- 3) перечитать конфигурацию Systemd:

```
# systemctl daemon-reload
```

- 4) включить таймер в автозагрузку:

```
# systemctl enable timebackup.timer
```

- 5) запустить таймер:

```
# systemctl start timebackup.timer
```

Резервное копирование будет выполняться ежедневно в 16:50. Будет создан новый каталог `Date-2024-02-14--16-50` с резервной копией.

Просмотреть список таймеров можно командой:

```
# systemctl list-timers
```

Для восстановления данных по состоянию на определенную дату необходимо выполнить следующие действия:

- 1) создать каталог для восстановления, например, `/var/restore`:

```
# mkdir /var/restore
```

- 2) выполнить восстановление данных

```
# rsync -rlptgoX /var/backup/Date-2024-02-14--16-50 /var/restore
```

Для выборочного восстановления отдельных файлов и каталогов (в примере файл `new.txt` из домашнего каталога пользователя `test1`) необходимо в каталоге `/var/backup` найти резервную копию файла `new.txt` за нужную дату и выполнить копирование данных в каталог пользователя `test1`:

```
# rsync -rlptgoX /var/backup/Date-2024-02-14--16-50/home/test1/  
/new.txt /home/test1/new.txt
```

### 3.7.3. Восстановление программного обеспечения при возникновении нештатных ситуаций

В случае нештатного выключения системы при последующей загрузке производится проверка файловых систем, результаты которой доступны в системных журналах.

Также можно провести проверку средствами контроля целостности (см. п. 3.6), в случае, если проверка показала нарушения целостности, произвести восстановление программного обеспечения (включая программное обеспечение средств защиты) при помощи средств резервного копирования.

## 3.8. Средства управления протоколированием событий

### 3.8.1. Управление журналированием

#### 3.8.1.1. Системная служба `syslogd`

В ОС Альт СП функция записи информации о системных событиях и событиях безопасности обеспечивается с помощью системной службы `syslogd` (пакет `rsyslogd`).

Средства журналирования в ОС Альт СП функционируют в соответствии со следующим алгоритмом:

- программы (источники регистрируемых данных) формируют простые текстовые сообщения о происходящих в них событиях и передают их на обработку в ядро, инициализируя при этом системный вызов;

- системная служба `syslogd` сравнивает каждую поступившую запись с правилами, которые находятся в файле конфигурации `/etc/syslog.conf`: в случае обнаружения соответствия служба `syslogd` обрабатывает запись описанным в конфигурационном файле `syslog.conf` способом.

**Примечание.** Файл `/etc/syslog.conf` создается после запуска служб `rsyslogd` и `rsyslog`.

Основные положения средств журналирования:

- формирование сообщений о событиях и их передача осуществляется по определенным правилам (протокол Syslog);
- передача текстовых сообщений осуществляется с использованием сетевых или доменных сокетов;
- источники сообщений могут располагаться на разных машинах.

Все регистрируемые сообщения по умолчанию записываются в каталог системного журнала `/var/log`, при необходимости могут быть указаны и другие хранилища (для каждой службы может быть установлено собственное хранилище или несколько хранилищ).

**Примечание.** Для очистки системных журналов от сообщений об устаревших событиях в ОС Альт СП используется служба `logrotate`.

### 3.8.1.2. Системная служба `systemd-journald`

В ОС Альт СП функция записи информации о системных событиях и событиях безопасности обеспечивается также с помощью системной службы `systemd-journald`. Она создает и поддерживает структурированные, индексированные журналы, на основе регистрируемой информации, полученной от ядра, от пользовательских процессов через вызов `Libc syslog`, от потоков `STDOUT/STDERR` системных служб через собственный API. Журналы данного инструмента хранятся в бинарном виде в `/var/log/journal`, что исключает возможность просмотра содержимого данных файлов стандартными утилитами обработки текстовых данных. Для просмотра логов используется утилита `journalctl`.

Journald может работать совместно с syslog.

### 3.8.1.3. Просмотр журналов systemd с помощью команды journalctl

С помощью команды journalctl на рабочих станциях пользователей доступен просмотр журналов для анализа и отладки работы системных компонентов. Просмотр осуществляется с помощью следующей команды journalctl.

Синтаксис:

```
journalctl [ПАРАМЕТРЫ...] [СООТВЕТСТВИЯ...]
```

При вводе команды без аргументов, как представлено выше, на консоль выводится список всех журнальных сообщений, включая исходящие как от системных компонентов, так и от пользователей, прошедших авторизацию.

При этом:

- строки с приоритетом «error» и выше подсвечены красным;
- строки с приоритетом «notice» и «warning» выделены жирным шрифтом;
- все отметки времени сформированы с учетом часового пояса;
- для навигации по тексту используется просмотрщик («pager»), по умолчанию «less»;
- выводятся все доступные данные, включая информацию из файлов, прошедших ротацию («rotated logs»);
- загрузка системы отмечается специальной строкой, отделяющей записи, сгенерированные между (пере-)загрузками.

Файлы журнала по умолчанию принадлежат и доступны для чтения системной группе systemd-journal (но не доступны для записи). Таким образом, добавление пользователя в эту группу позволит ему читать файлы журнала.

По умолчанию каждый зарегистрированный пользователь получит свой собственный набор файлов журнала в /var/log/journal/. Однако эти файлы не будут принадлежать пользователю, чтобы избежать прямого доступа к ним. Каждый

зарегистрированный пользователь имеет доступ на чтение только собственного набора журналов.

### 3.8.1.3.1. Фильтрация записей

Фильтрация записей в `journalctl` выполняется с помощью опций-ключей.

Опция `-b` позволяет просмотреть все данные журналов, собранные с момента последней загрузки системы:

```
$ journalctl -b
```

Опции `--since` и `--until` позволяют просматривать журналы за определенные периоды времени:

```
$ journalctl --since "2023-12-22 17:15:00"
```

В случае, если с опцией `since` не будет указано никакой даты, на консоль будут выведены данные журналов, начиная с текущей даты. В случае, если дата указана, но при этом не указано время, будет применено значение времени по умолчанию «00:00:00». Также можно воспользоваться следующими командами:

```
$ journalctl --since yesterday
$ journalctl --since 09:00 --until now
$ journalctl --since 10:00 --until "1 hour ago"
```

Для просмотра логов конкретного приложения или службы используется опция `-u`:

```
$ journalctl -u <приложение_или_служба>
```

Также допускается просмотр логов какой-либо службы за определенный период времени:

```
$ journalctl -u network.service --since yesterday
```

Просмотр всех записей в журнале, сделанных определенной службой и только в текущей загрузке системы:

```
$ journalctl -b -u network.service
```

Благодаря этому можно отслеживать взаимодействие различных служб и получать информацию, которую нельзя было бы получить при отслеживании соответствующих процессов по отдельности.

Просмотреть логи для какого-либо процесса можно, указав в команде его идентификационный номер (PID):

```
$ journalctl _PID=381
```

Для просмотра логов процессов, запущенных от имени определенного пользователя или группы, используются фильтры `_UID` и `_GID` соответственно:

```
$ journalctl _UID=33
```

Вывести на консоль список пользователей, о которых имеются записи в логах, можно следующим образом:

```
$ journalctl -F _UID
```

Для просмотра аналогичного списка пользовательских групп используется следующая команда:

```
$ journalctl -F _GUID
```

Просмотр сообщений ядра:

```
$ journalctl -k
```

Просмотреть список всех доступных фильтров можно, выполнив команду:

```
$ man systemd.journal-fields
```

Кроме того, в `journalctl` предусмотрена возможность фильтрации по уровню ошибки. В `journal` используется такая же классификация уровней ошибок, как и в `syslog`:

- 0 – EMERG (система неработоспособна);
- 1 – ALERT (требуется немедленное вмешательство);
- 2 – CRIT (критическое состояние);
- 3 – ERR (ошибка);
- 4 – WARNING (предупреждение);
- 5 – NOTICE (все нормально, но следует обратить внимание);
- 6 – INFO (информационное сообщение);
- 7 – DEBUG (отложенная печать).

Коды уровней ошибок указываются после опции `-p`:

```
$ journalctl -p err -b
```

Приведенная команда покажет все сообщения об ошибках, имевших место в системе.

Также вместо имени приоритета можно указывать номер приоритета согласно указанному выше списку:

```
$ journalctl -p 3 -b
```

### 3.8.1.3.2. Запись логов в стандартный вывод

По умолчанию `journalctl` использует для вывода сообщений логов внешнюю команду `less`.

При использовании опции `--no-pager` все сообщения будут записываться в стандартный вывод, без возможности листать страницы журнала:

```
$ journalctl --no-pager
```

После этого их можно будет передать другим утилитам для дальнейшей обработки или сохранить в текстовом файле.

### 3.8.1.3.3. Выбор формата вывода

С помощью опции `-o` можно преобразовывать данные логов в различные форматы, что облегчает их парсинг и дальнейшую обработку, например:

```
$ journalctl -u network.service -o json
{ "__CURSOR" :
"s=13a21661cf4948289c63075db6c25c00;i=116f1;b=81b58db8fd9046ab9f8
47ddb82a2fa2d;m=19f0daa;t=50e33c33587ae;x=e307daadb4858635",
"__REALTIME_TIMESTAMP" : "1422990364739502",
"__MONOTONIC_TIMESTAMP" : "27200938", "_BOOT_ID" :
"81b58db8fd9046ab9f847ddb82a2fa2d", "PRIORITY" : "6", "_UID" :
"0", "_GID" : "0", "_CAP_EFFECTIVE" : "3fffffffff",
"_MACHINE_ID" : "752737531a9d1a9c1e3cb52a4ab967ee", "_HOSTNAME" :
"desktop", "SYSLOG_FACILITY" : "3", "CODE_FILE" :
"src/core/unit.c", "CODE_LINE" : "1402", "CODE_FUNCTION" :
"unit_status_log_starting_stopping_reloading",
"SYSLOG_IDENTIFIER" : "systemd", "MESSAGE_ID" :
"7d4958e842da4a758f6c1cdc7b36dcc5", "_TRANSPORT" : "journal",
"_PID" : "1", "_COMM" : "systemd", "_EXE" :
"/usr/lib/systemd/systemd", "_CMDLINE" :
"/usr/lib/systemd/systemd", "_SYSTEMD_CGROUP" : "/", "UNIT" :
"network.service ", "MESSAGE" : "Starting A high performance web
```

```
server and a reverse proxy server...",
"_SOURCE_REALTIME_TIMESTAMP" : "1422990364737973" }
```

Помимо JSON данные журналов могут быть преобразованы в следующие форматы:

- `cat` – только сообщения из журналов без служебных полей;
- `export` – бинарный формат, подходит для экспорта или резервного копирования логов;
- `short` – формат вывода `syslog`;
- `short-monotonic` – формат вывода `syslog` с метками монотонного времени (`monotonic timestamp`);
- `verbose` – максимально подробный формат представления данных (включает даже те поля, которые в других форматах не отображаются).

#### 3.8.1.3.4. Просмотр информации о недавних событиях

Опция `-n` используется для просмотра информации о недавних событиях в системе:

```
$ journalctl -n
```

По умолчанию на консоль выводится информация о последних 10 событиях. С опцией `-n` можно указать необходимое число событий:

```
$ journalctl -n 20
```

Сообщения из журналов можно просматривать не только в виде сохраненных файлов, но и в режиме реального времени. Для этого используется опция `-f`:

```
$ journalctl -f
```

#### 3.8.1.3.5. Примеры просмотра записей журнала

Просмотр всех попыток пользователей войти в систему с момента последней загрузки системы:

```
$ journalctl -b | grep USER_LOGIN
```

```
июн 01 13:16:30 host-15.localdomain audit[1622]: USER_LOGIN
pid=1622 uid=0 auid=500 ses=2 msg='op=login id=500
exe="/usr/sbin/lightdm" hostname=host-15.localdomain addr=?
terminal=/dev/tty1 res=success'
```



## ЛКНВ.11102-01 99 01

```

июн 01 13:17:10 host-15.localdomain audit[2205]: USER_LOGIN
pid=2205 uid=0 auid=500 ses=4 msg='op=login id=500
exe="/usr/sbin/sshd" hostname=192.168.3.191 addr=192.168.3.191
terminal=/dev/pts/1 res=success'
июн 01 13:17:51 host-15.localdomain audit[2259]: USER_LOGIN
pid=2259 uid=0 auid=4294967295 ses=4294967295 msg='op=login
acct="root" exe="/usr/sbin/sshd" hostname=? addr=192.168.3.191
terminal=sshd res=failed'
июн 01 13:19:00 host-15.localdomain audit[2270]: USER_LOGIN
pid=2270 uid=0 auid=4294967295 ses=4294967295 msg='op=login
acct="user" exe="/bin/login" hostname=host-15.localdomain addr=?
terminal=/dev/tty3 res=failed'
июн 01 13:19:07 host-15.localdomain audit[2270]: USER_LOGIN
pid=2270 uid=0 auid=0 ses=5 msg='op=login id=0 exe="/bin/login"
hostname=host-15.localdomain addr=? terminal=/dev/tty3
res=success'

```

Приведенная команда покажет все попытки входа в систему с момента последней загрузки системы. При этом для каждой записи будут выведены следующие параметры: имя пользователя (op=login acct="user", op=login id=500), дата и время входа, результат попытки входа (успешный/неуспешный – res=failed/success), используемый терминал для входа (terminal=/dev/tty3), в случае удаленного входа через сеть – адрес узла, с которого осуществляется вход (addr=192.168.3.191 terminal=sshd).

Вывести только неуспешные попытки входа в систему можно с помощью команды:

```

$ journalctl -b | grep "USER_LOGIN.*failed"

июн 01 13:17:56 host-15.localdomain audit[2259]: USER_LOGIN
pid=2259 uid=0 auid=4294967295 ses=4294967295 msg='op=login
acct="root" exe="/usr/sbin/sshd" hostname=? addr=192.168.3.191
terminal=sshd res=failed'
июн 01 13:19:00 host-15.localdomain audit[2270]: USER_LOGIN
pid=2270 uid=0 auid=4294967295 ses=4294967295 msg='op=login
acct="user" exe="/bin/login" hostname=host-15.localdomain addr=?
terminal=/dev/tty3 res=failed'

```

**Информация о выходе пользователей из системы:**

```

$ journalctl -b | grep USER_END

июн 01 13:43:30 host-15.localdomain audit[2270]: USER_END
pid=2270 uid=0 auid=0 ses=5 msg='op=PAM:session_close

```

```
grantors=pam_tcb,pam_mktemp,pam_limits,pam_loginuid,pam_systemd,p
am_lastlog,pam_mail,pam_console,pam_ck_connector acct="root"
exe="/bin/login" hostname=localhost addr=127.0.0.1
terminal=/dev/tty3 res=success'
июн 01 13:43:45 host-15.localdomain audit[2205]: USER_END
pid=2205 uid=0 auid=500 ses=4 msg='op=PAM:session_close
grantors=pam_tcb,pam_mktemp,pam_limits,pam_loginuid,pam_systemd
acct="user" exe="/usr/sbin/sshd" hostname=192.168.3.191
addr=192.168.3.191 terminal=ssh res=success'
```

**Просмотр попыток пользователей войти в систему после превышения количества неправильно введенных паролей:**

```
# journalctl -b | grep "pam_tally2(.*:auth): user\|too many bad
attempts"
сен 10 17:05:34 host-105.localdomain lightdm[3130]:
pam_faillock(lightdm:auth): user test (502) tally 3, deny 2
сен 10 17:05:38 host-105.localdomain lightdm[3133]:
pam_faillock(lightdm:auth): user test (502) tally 4, deny 2
сен 10 17:11:17 host-105.localdomain login[3211]:
pam_faillock(login:auth): user test (502) tally 3, deny 2
сен 10 17:12:00 host-105.localdomain login[3211]:
login_authenticate_user: Login failed - too many bad attempts
сен 10 17:12:24 host-105.localdomain lightdm[3308]:
pam_faillock(lightdm:auth): user test (501) tally 3, deny 2
```

**С помощью записей в системном журнале можно проследить все события от конкретной загрузки операционной системы до ее программного останова.**

**Просмотреть список предыдущих загрузок можно с помощью команды:**

```
$ journalctl --list-boots
-2 03b740b67cbf436d96769e632fe87a9c Wed 2024-05-16 13:10:02 MSK-
Wed 2024-05-16 20:55:05 MSK
-1 6247ec62e2b14ed6a7539de5fd955f0d Fri 2024-06-01 13:12:32 MSK-
Fri 2024-06-01 13:55:53 MSK
0 3a6dfac76afd40ab93d6c8cff91c2c76 Fri 2024-06-01 14:47:19 MSK-
Fri 2024-06-01 15:06:12 MSK
```

**Вывод состоит из четырех колонок. В первой из них указывается порядковый номер загрузки, во второй – ее ID, в третьей – дата и время. Чтобы просмотреть журнал для конкретной загрузки, можно использовать идентификаторы, как из первой, так и из второй колонки:**

```
$ journalctl -b -1
```

Будет показан полный журнал от момента загрузки системы, до момента ее выключения и остановки всех сервисов.

### 3.8.1.3.6. Управление логированием

Узнать объем имеющихся на текущий момент логов можно с помощью команды:

```
$ journalctl --disk-usage
Journals take up 16.0M on disk.
```

Ротация логов:

- для удаления с помощью указания размера (опция `--vacuum-size`) необходимо установить предельно допустимый размер для хранимых на диске журналов, как только объем журналов превысит указанную цифру, лишние файлы будут автоматически удалены:

```
$ journalctl --vacuum-size=200M
```

- для удаления старых записей по времени (опция `--vacuum-time`) необходимо установить для журналов срок хранения, по истечении которого они будут автоматически удалены:

```
# journalctl --vacuum-time=1months
```

Настройки ротации логов можно прописать в конфигурационном файле `/etc/systemd/journald.conf`, который включает в числе прочих следующие параметры:

- `SystemMaxUse` – максимальный объем, который логи могут занимать на диске;
- `SystemKeepFree` – объем свободного места, которое должно оставаться на диске после сохранения логов;
- `SystemMaxFileSize` – объем файла лога, по достижении которого он должен быть удален с диска;
- `RuntimeMaxUse` – максимальный объем, который логи могут занимать в файловой системе `/run`;

- `RuntimeKeepFree` – объем свободного места, которое должно оставаться в файловой системе `/run` после сохранения логов;
- `RuntimeMaxFileSize` – объем файла лога, по достижении которого он должен быть удален из файловой системы `/run`.

#### 3.8.1.4. Просмотр системных журналов в графической среде (alterator-logs)

Модуль ЦУС «Системные журналы» (alterator-logs) предназначен для просмотра системных журналов в графическом или веб-интерфейсе.

Для перехода в «Центр управления системой» выбрать на панели инструментов меню «Система» → «Администрирование» → «Центр управления системой».

При запуске необходимо ввести пароль администратора (root). После успешного входа откроется окно «Центра управления системой» (рис. 41), в котором нужно выбрать пункт «Системные журналы» из секции «Система». Откроется графический интерфейс модуля «Системные журналы» (рис. 42).

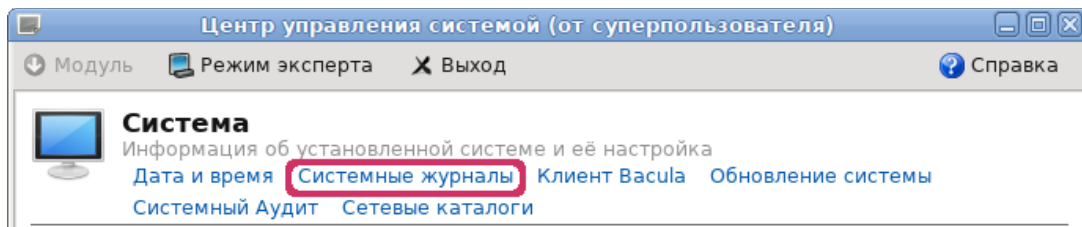


Рис. 41 – Центр управления системой

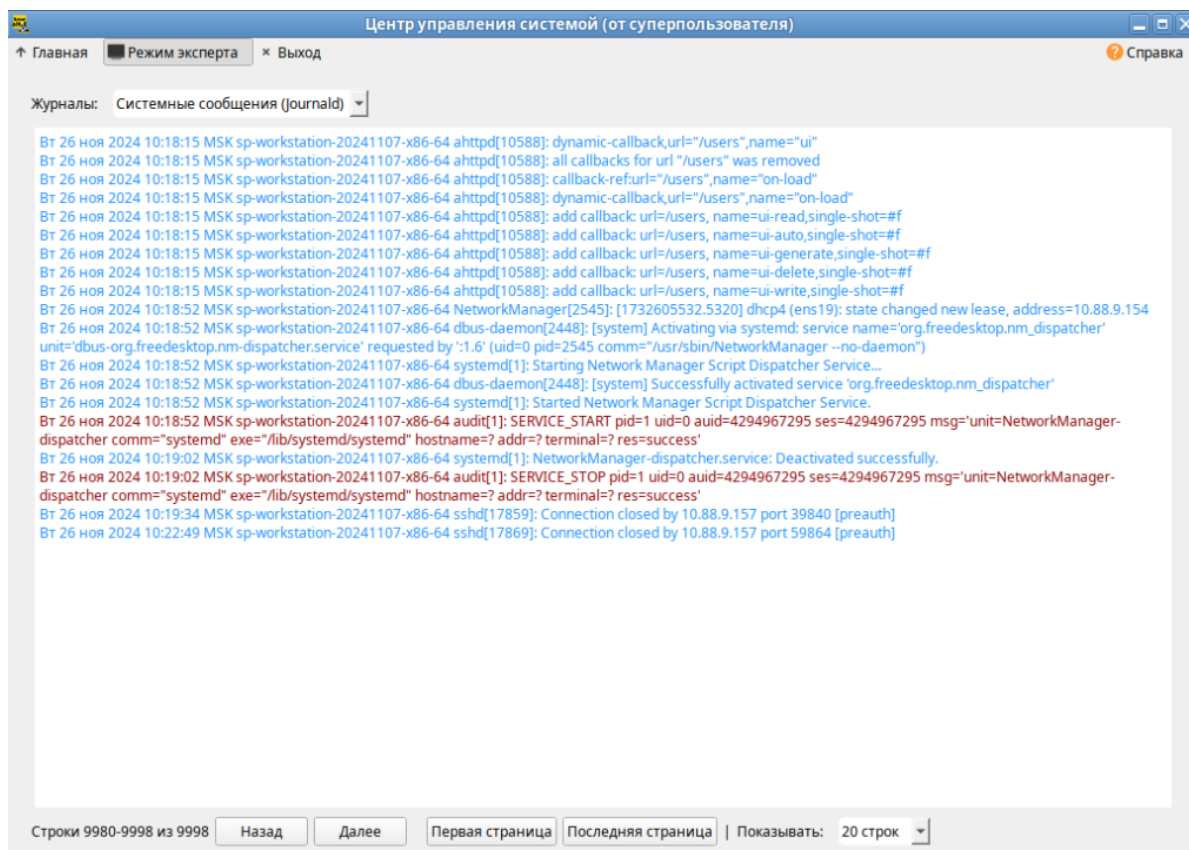


Рис. 42 – Просмотр системных журналов (alterator-logs) в графическом интерфейсе ЦУС

Также модуль «Системные журналы» доступен в веб-интерфейсе ЦУС (<https://ip-address:8080>) (рис. 43).

Системные журналы позволяют отслеживать события, происходящие с системой. Эта информация может быть полезна при диагностике разного рода проблем.

Уменьшить либо увеличить количество выводимых строк можно, выбрав нужное значение в списке «Показывать». При необходимости просмотра более старых/новых сообщений можно воспользоваться кнопками «Назад» и «Далее» соответственно. Переход к самым старым и самым новым сообщениям осуществляется кнопками «Последняя страница» и «Первая страница».

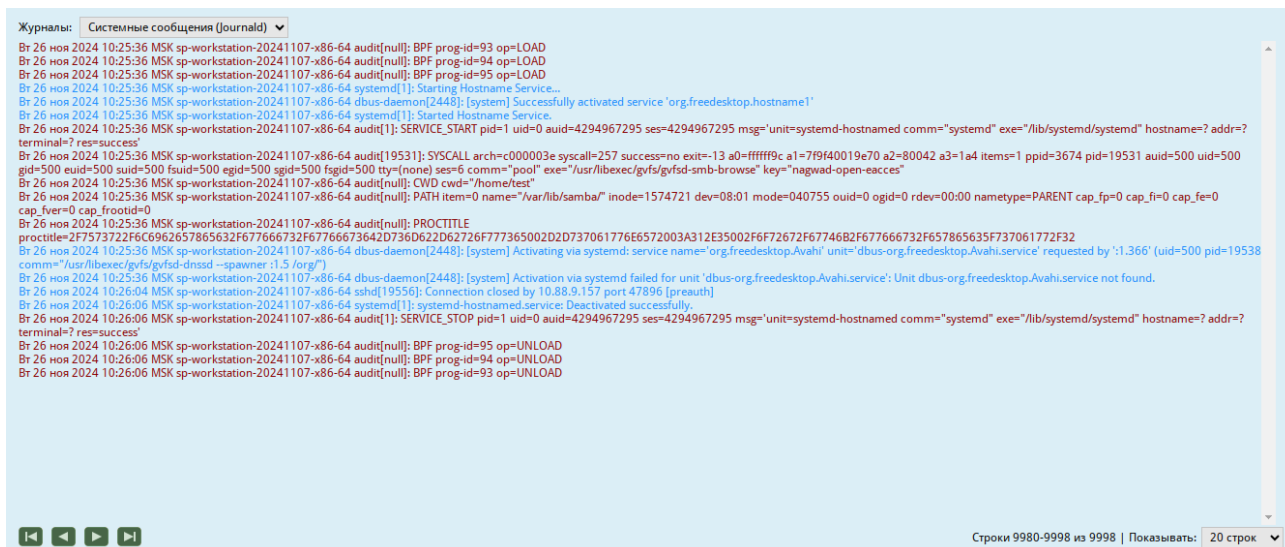


Рис. 43 – Веб-интерфейс ЦУС просмотра системных журналов

Для просмотра в выпадающем меню (см. рис. 42) могут быть доступны, например, следующие виды журналов:

- «Безопасность» – отображается информация, связанная с аутентификацией пользователей, ошибками входа в систему, изменением уровня доступа, длительностью сеанса пользователей;
- «Брандмауэр» – события безопасности, связанные с работой брандмауэра ОС;
- «Сервер резервного копирования» – отчеты и события, связанные с работой сервера резервного копирования, если таковой настроен;
- «Ядро» – сообщения от ядра ОС;
- «Электронная почта» – сообщения о получении и доставке писем (журнал обычно ведется почтовым сервером);
- «Системные сообщения» – сообщения от системных служб (сообщения с типом DAEMON).

### 3.8.2. Управление аудитом

Служба `auditd` – это прикладной компонент системы аудита ОС, который ведет протокол аудита на системном диске и формирует перечень событий, происходящих

в системе. Кроме самого факта возникновения события, система аудита представляет такую информацию, как дата и время возникновения события, ответственность пользователя за событие, тип события и его успешность.

Конфигурации аудита хранятся в файле `/etc/audit/auditd.conf`, правила аудита, загружаемые при запуске службы, хранятся в файле `/etc/audit/audit.rules`.

Для просмотра протоколов используются команды `ausearch` и `aureport`. Команда `auditctl` позволяет настраивать правила аудита. Кроме того, при загрузке загружаются правила из файла `/etc/audit/audit.rules`. Некоторые параметры самой службы можно изменить в файле `auditd.conf`.

### 3.8.2.1. Команда `auditd`

Синтаксис:

```
auditd [-f] [-l] [-n] [-s disable|enable|nochange] [-c <config_dir>]
```

Опции:

- 1) `-f` не переходить в фоновый режим (для отладки). Сообщения программы будут направляться в стандартный вывод для ошибок (`stderr`), а не в файл;
- 2) `-l` включить следование по символическим ссылкам при поиске конфигурационных файлов;
- 3) `-n` не создавать дочерний процесс. Для запуска из `inittab` или `system`;
- 4) `-s=ENABLE_STATE` указать, должен ли `auditd` при старте изменять текущее значение флага ядра — `enabled`. Допустимые значения `ENABLE_STATE`: `disable`, `enable` и `nochange`. Значение по умолчанию `enable` (`disable`, когда `auditd` остановлен). Значение флага может быть изменено во время жизненного цикла `auditd` с помощью команды: `auditctl -e`;
- 5) `-c` указать альтернативный каталог конфигурационного файла (по умолчанию: `/etc/audit/`). Этот же каталог будет передан диспетчеру.

### Сигналы:

- `SIGHUP` – перезагрузить конфигурацию – загрузить файл конфигурации с диска. Если в файле не окажется синтаксических ошибок, внесенные в него изменения вступят в силу. При этом в протокол будет добавлена запись о событии `DAEMON_CONFIG`. В противном случае действия службы будут зависеть от параметров `space_left_action`, `admin_space_left_action`, `disk_full_action`, `disk_error_action` в файле `auditd.conf`;
- `SIGTERM` – прекратить обработку событий аудита и завершить работу, о чем предварительно занести запись в протокол;
- `SIGUSR1` – произвести ротацию файлов журналов `auditd`. Создать новый файл для протокола, перенумеровав старые файлы или удалив часть из них, в зависимости от параметра `max_log_size_action`;
- `SIGUSR2` – попытаться возобновить ведение журналов `auditd` (необходимо после приостановки ведения журнала).

### Файлы:

- `/etc/audit/auditd.conf` – файл конфигурации службы аудита;
- `/etc/audit/audit.rules` – правила аудита (загружается при запуске службы);
- `/etc/audit/rules.d/` – каталог, содержащий отдельные наборы правил, которые будут скомпилированы в один файл утилитой `augenrules`.

Для того чтобы сделать возможным аудит всех процессов, запущенных до службы аудита, необходимо добавить в строку параметров ядра (в конфигурации загрузчика) параметр `audit=1`. В противном случае аудит некоторых процессов будет невозможен.

Демон аудита может получать события – сообщения от других приложений через плагин `audispd`: `audisp-remote`. Демон аудита может быть связан с `tcp_wrappers`, чтобы контролировать, какие машины могут подключаться. В этом случае можно добавить запись в `hosts.allow` и отказать в соединении.



### 3.8.2.2. Файл конфигурации `auditd.conf`

В файле `/etc/audit/auditd.conf` определяются параметры службы аудита. На одной строке может быть не больше одной директивы. Директива состоит из ключевого слова (названия параметра), знака равенства и соответствующих ему данных (значения параметра). Все названия и значения параметров чувствительны к регистру. Допустимые ключевые слова перечислены и описаны ниже. Каждая строка должна быть ограничена 160 символами, иначе она будет пропущена. К файлу можно добавить комментарии, начав строку с символа '#'.  
#

Описание ключевых слов:

- `local_events` – ключевое слово `yes/no`, указывающее, следует ли включать запись локальных событий (значение по умолчанию – `yes`). В случае если необходимо записывать только сообщения из сети, следует установить значение – `no`. Этот параметр полезен, если демон аудита работает в контейнере. Данный параметр может быть установлен только один раз при запуске аудита. Перезагрузка файла конфигурации никак на него не влияет;
- `log_file` – полное имя файла, в который следует записывать протокол;
- `write_logs` – ключевое слово `yes/no`, указывающее, следует ли записывать журналы (значение по умолчанию – `yes`);
- `log_format` – оформление данных в протоколе. Допустимы два значения: `raw` и `enriched`. При указании `RAW`, данные будут записываться в том виде, в котором они получаются от ядра. Значение `ENRICHED` разрешает информацию (вместо идентификатора, будет указано значение): идентификатор пользователя (`uid`), идентификатор группы (`gid`), системный вызов (`syscall`), архитектуру и адрес сокета перед записью события на диск. Это помогает осмыслить события, созданные в одной системе, но сообщенные/проанализированные в другой системе. Значение `NOLOG` устарело, вместо него следует установить параметр `write_logs` в значение `no`;

- `log_group` – указывает группу, на которую распространяются права на файлы журнала (по умолчанию – `root`). Можно использовать либо идентификатор, либо имя группы;
- `priority_boost` – неотрицательное число, определяющее повышение приоритета выполнения службы аудита. Значение по умолчанию – 4. Для того чтобы не изменять приоритет, следует указать – 0;
- `flush` – стратегия работы с дисковым буфером. Допустимые значения: `none`, `incremental`, `incremental_async`, `data` и `sync`. Вариант `none` отключает какие-либо дополнительные действия со стороны службы по синхронизации буфера с диском. При значении `incremental`, запросы на перенос данных из буфера на диск выполняются с частотой, задаваемой параметром `freq`. Значение `incremental_async` очень похоже на значение `incremental`, за исключением того, что перенос данных выполняется асинхронно для более высокой производительности. При значении `data` данные файла синхронизируются немедленно. Значение `sync` указывает на необходимость немедленной синхронизации, как данных, так и метаданных файла при записи на диск. Значение по умолчанию – `incremental_async`;
- `freq` – максимальное число записей протокола, которые могут храниться в буфере. При достижении этого числа производится запись буферизованных данных на диск. Данный параметр допустим только в том случае, когда `flush` имеет значение `incremental` или `incremental_async`;
- `num_logs` – максимальное число файлов с протоколами. Используется в том случае, если параметр `max_log_file_action` имеет значение `rotate`. Если указано число меньше 2, при достижении ограничения на размер файла он обнуляется. Значение параметра не должно превышать 999. Значение по умолчанию: 0 (то есть ротация файлов не происходит). При указании большого числа может потребоваться увеличить ограничение на количество ожидающих запросов (в файле `/etc/audit/audit.rules`).

- Если настроена ротация журналов, демон проверяет наличие лишних журналов и удаляет их, чтобы освободить место на диске. Проверка выполняется только при запуске и при проверке изменения конфигурации;
- `disp_qos` – разрешить ли блокирование при взаимодействии с диспетчером. Для передачи информации диспетчеру используется буфер размером 128 Кбайт. Это значение является оптимальным для большинства случаев. Если блокирование запрещено (`lossy`), то все сообщения, поступающие при полном буфере, не будут доходить до диспетчера (записи о них по-прежнему будут вноситься в файл на диске, если только `log_format` не равно `NOLOG`). В случае, если блокирование разрешено (`lossless`), служба аудита будет ожидать появления свободного места в очереди, передавать сообщение диспетчеру и только потом записывать его на диск. Допустимые значения: `lossy` и `lossless`. Значение по умолчанию – `lossy`;
  - `dispatcher` – диспетчер – программа, которой (на стандартный ввод) будут передаваться копии сообщений о событиях аудита. Она запускается (с правами администратора) службой аудита при загрузке последней;
  - `name_format` – контролирует, как имена узлов компьютеров вставляются в поток событий аудита. Допустимы следующие значения: `none`, `hostname`, `fqd`, `numeric` и `user`. При значении `none` имя компьютера не используется в записи аудита. `Hostname` – имя, возвращаемое системным вызовом `gethostname`. Значение `fqd` означает, что аудит принимает имя хоста и разрешает его с помощью DNS в полное доменное имя этой машины. Значение `numeric` схоже с `fqd`, за исключением того, что разрешается IP-адрес машины. Чтобы использовать эту опцию, нужно проверить, что команда `'hostname -i'` или `'domainname -i'` возвращает числовой адрес. Кроме того, эта опция не рекомендуется, если используется `dhcp`, поскольку у одной и той же машины в разное время могут быть разные адреса. `User` это строка, определенная администратором в параметре `name`. Значение по умолчанию – `none`;

- `name` – строка, определенная администратором, которая идентифицирует компьютер, если в параметре `name_format` указано значение `user`;
- `max_log_file` – ограничение на размер файла протокола в мегабайтах. Действие, выполняемое при достижении размера файла указанного значения, можно настроить с помощью следующего параметра;
- `max_log_file_action` – действие, предпринимаемое при достижении размером файла протокола максимального значения. Допустимые значения: `ignore`, `syslog`, `suspend`, `rotate` и `keep_logs`. Вариант `ignore` отключает контроль над размером файла. При значении `syslog` в системный протокол будет внесено соответствующее сообщение. При значении `suspend` дальнейшее ведение протокола будет прекращено. Служба по-прежнему будет работать. При значении `rotate` текущий файл будет переименован и для протокола будет создан новый файл. Имя предыдущего протокола будет дополнено числом 1, а номера других протоколов (если они имеются) будут увеличены на единицу. Таким образом, чем больше номер у протокола, тем он старше. Максимальное число файлов определяется параметром `num_logs` (соответствие ему достигается за счет удаления самых старых протоколов). Такое поведение аналогично поведению утилиты `logrotate`. Вариант `keep_logs` аналогичен предыдущему, но число файлов не ограничено, это предотвращает потерю данных аудита. Протоколы накапливаются и не удаляются, что может вызвать событие `space_left_action`, если весь объем заполнится. Это значение следует использовать в сочетании с внешним сценарием, который будет периодически архивировать журналы;
- `verify_email` – определяет, указан ли адрес электронной почты в параметре `action_mail_acct` и проверяет, можно ли разрешить имя домена. Этот параметр должен быть предоставлен до параметра `action_mail_acct`, иначе будет использовано значение по умолчанию – `yes`;

- `action_mail_acct` – адрес электронной почты. Значение по умолчанию: `root`. Если адрес не локальный по отношению к данной системе, необходимо чтобы в ней был настроен механизм отправки почты. В частности, требуется наличие программы `/usr/lib/sendmail`;
- `space_left` – минимум свободного пространства в мегабайтах, при достижении которого должно выполняться действие, определяемое следующим параметром;
- `space_left_action` – действие, предпринимаемое при достижении объемом свободного пространства на диске указанного минимума. Допустимые значения – `ignore`, `syslog`, `rotate`, `email`, `exec`, `suspend`, `single` и `halt`. При значении `ignore` никаких действий не производится. При значении `syslog` в системный протокол добавляется соответствующая запись. При значении `rotate` будет производиться ротация журналов, с удалением самых старых, чтобы освободить место на диске. При значении `email` по адресу, указанному в `action_mail_acct`, отправляется уведомление. При значении `exec` <путь к программе> запускается программа по указанному пути (передача параметров не поддерживается). При значении `suspend` служба аудита прекратит вести протокол событий на диске, но будет продолжать работать. Указание `single` приведет к переводу компьютера в однопользовательский режим (см. п. 3.6.2.3). Указание `halt` приведет к выключению компьютера;
- `admin_space_left` – критический минимум свободного пространства в мегабайтах, при достижении которого должно выполняться действие, определяемое следующим параметром. Данное действие следует рассматривать как последнюю меру, предпринимаемую перед тем, как закончится место на диске. Значение настоящего параметра должно быть меньше значения `space_left`;

- `admin_space_left_action` – действие, предпринимаемое при достижении объемом свободного пространства на диске указанного критического минимума. Допустимые значения – `ignore`, `syslog`, `rotate`, `email`, `exec`, `suspend`, `single` и `halt`. При значении `ignore` никаких действий не производится. При значении `syslog` в системный протокол добавляется соответствующая запись. При значении `rotate` будет производиться ротация журналов, с удалением самых старых, чтобы освободить место на диске. При значении `email` по адресу, указанному в `action_mail_acct` отправляется уведомление. При значении `exec` <путь к программе> запускается программа по указанному пути (передача параметров не поддерживается). При значении `suspend` служба аудита прекратит вести протокол событий на диске, но будет продолжать работать. Указание `single` приведет к переводу компьютера в однопользовательский режим (см. п. 3.6.2.3). Указание `halt` приведет к выключению компьютера;
- `disk_full_action` – действие, предпринимаемое при обнаружении отсутствия свободного пространства на диске. Допустимые значения – `ignore`, `syslog`, `rotate`, `exec`, `suspend`, `single` и `halt`. При значении `ignore` никаких действий не производится. При значении `syslog` в системный протокол добавляется соответствующая запись. При значении `rotate` будет производиться ротация журналов, с удалением самых старых, чтобы освободить место на диске. При значении `exec` <путь к программе> запускается программа по указанному пути (передача параметров не поддерживается). При значении `suspend` служба аудита прекратит вести протокол событий на диске, но будет продолжать работать. Указание `single` приведет к переводу компьютера в однопользовательский режим (см. п. 3.6.2.3). Указание `halt` приведет к выключению компьютера;

- `disk_error_action` – действие, предпринимаемое при возникновении ошибки в работе с диском. Допустимые значения – `ignore`, `syslog`, `exec`, `suspend`, `single` и `halt`.

При значении `ignore` никаких действий не производится. При значении `syslog` в системный протокол добавляется соответствующая запись. При значении `exec` <путь к программе> запускается программа по указанному пути (передача параметров не поддерживается). При значении `suspend` служба аудита прекратит вести протокол событий на диске, но будет продолжать работать. Указание `single` приведет к переводу компьютера в однопользовательский режим (см. п. 3.6.2.3). Указание `halt` приведет к выключению компьютера;

- `tcp_listen_port` – числовое значение в диапазоне 1..65535, при указании которого служба аудита будет прослушивать соответствующий TCP порт для аудита удаленных систем. Демон аудита может быть связан с `tcp_wrappers`, чтобы контролировать, какие машины могут подключаться. В этом случае можно добавить запись в `hosts.allow` и отказать в соединении. Если решение развернуто в ОС на основе `systemd` может потребоваться изменить параметр `After`;

- `tcp_listen_queue` – количество разрешенных ожидающих подключений (запрошенных, но не принятых). Значение по умолчанию – 5. Установка слишком маленького значения может привести к отклонению соединений, при одновременном запуске нескольких хостов (например, после сбоя питания);

- `tcp_max_per_addr` – количество одновременных подключений с одного IP-адреса. Значение по умолчанию – 1, максимальное значение – 1024. Установка слишком большого значения может привести к атаке типа «отказ в обслуживании» при ведении журнала сервером. Значение по умолчанию подходит в большинстве случаев;

- `use_libwrap` – следует ли использовать `tcp_wrappers` для распознавания попыток подключения с разрешенных компьютеров. Допустимые значения `yes` или `no`. Значение по умолчанию – `yes`;
- `tcp_client_ports` – порты, с которых можно принимать соединение. Значением параметра может быть либо число, либо два числа, разделенные тире (пробелы не допускаются). Если порт не указан, соединения принимаются с любого порта. Допустимые значения 1..65535. Например, для указания клиенту использовать привилегированный порт, следует указать значение 1-1023 для этого параметра, а также установить опцию `local_port` в файле `audisp-remote.conf`. Проверка того, что клиенты отправляют сообщения с привилегированного порта, это функция безопасности, предотвращающая атаки с использованием инъекций;
- `tcp_client_max_idle` – количество секунд, в течение которых клиент может бездействовать (то есть никаких данных от него нет). Используется для закрытия неактивных соединений, если на компьютере клиенте возникла проблема, из-за которой он не может завершить соединение корректно. Это глобальный параметр, его значение должно быть больше (желательно, в два раза), чем любой параметр клиента `heartbeat_timeout`. Значение по умолчанию – 0, что отключает эту проверку;
- `enable_krb5` – при значении `yes` – использовать Kerberos 5 для аутентификации и шифрования. Значение по умолчанию – `no`;
- `krb5_principal` – принципал для этого сервера. Значение по умолчанию – `auditd`. При значении по умолчанию, сервер будет искать ключ с именем типа `auditd/hostname@EXAMPLE.COM` в `/etc/audit/audit.key` для аутентификации себя, где `hostname` – имя сервера, возвращаемое запросом DNS имени по его IP-адресу;
- `krb5_key_file` – расположение ключа для принципала этого клиента. Файл ключа должен принадлежать пользователю `root` и иметь права 0400. По умолчанию – файл `/etc/audit/audit.key`;



- `distribute_network` – при значении `yes`, события, поступающие из сети будут передаваться диспетчеру аудита для обработки. Значение по умолчанию – `no`.

**Примечание.** Рекомендуется выделять для файла `/var/log/audit` специальный раздел. Кроме того, параметру `flush` необходимо присвоить значение `sync` или `data`.

Для обеспечения полного использования раздела параметрам `max_log_file` и `num_logs` необходимо присвоить соответствующие значения. Чем больше файлов создается на диске, тем больше времени будет уходить на обработку событий при достижении размером очередного файла максимума. Параметру `max_log_file_action` рекомендуется присвоить значение `keep_logs`.

Значение `space_left` должно быть таким, которое позволит администратору вовремя среагировать на предупреждение. Обычно в число действий, выполняемых администратором, входит запуск `aureport -t` и архивирование самых старых протоколов. Значение `space_left` зависит от системы, в частности от частоты поступления сообщений о событиях. Значение `space_left_action` рекомендуется установить в `email`. Если требуется отправка сообщения `snmp trap`, нужно указать вариант `exec`.

Значение `admin_space_left` должно быть установлено таким образом, чтобы хватило свободного места для сохранения записей о последующих действиях администратора. Значение параметра `admin_space_left_action` следует установить в `single`, ограничив, таким образом, способы взаимодействия с системой, консолью.

Действие, указанное в `disk_full_action`, выполняется, когда в разделе уже не осталось свободного места. Доступ к ресурсам машины должен быть полностью прекращен, так как нет возможности контролировать работу системы. Это можно сделать, указав значение `single` или `halt`.

Значение `disk_error_action` следует установить в `syslog`, `single`, либо `halt` в зависимости от соглашения относительно обработки сбоев аппаратного

обеспечения. Указание единственного разрешенного клиентского порта может затруднить перезапуск средств аудита у клиента, так как он не сможет восстановить соединение с теми же адресами и портами хоста, пока не истечет тайм-аут закрытия соединения `TIME_WAIT`.

### 3.8.2.3. Команда `auditctl`

Команда `auditctl` используется для настройки параметров ядра, связанных с аудитом, получения состояния и добавления/удаления правил аудита.

Синтаксис:

```
auditctl [опции]
```

Опции конфигурации команды `auditctl` приведены в таблице 3, опции состояния приведены в таблице 4. Опции правил приведены в таблице 5.

Т а б л и ц а 3 – Опции конфигурации команды `auditctl`

Опция	Описание
-b <количество>	Установить максимальное количество доступных для аудита буферов, ожидающих обработки (значение в ядре по умолчанию – «64»). В случае если все буферы заняты, то флаг сбоя будет выставлен ядром для его дальнейшей обработки.
--backlog_wait_time <время_ожидания>	Установить время ожидания для ядра достижения значения <code>backlog_limit</code> (значение в ядре по умолчанию – $60 * \text{HZ}$ ), прежде, чем ставить в очередь дополнительные события аудита для их передачи аудиту. Число должно быть больше или равно нулю, но меньше, чем десятикратное значение по умолчанию.
-c	Продолжать загружать правила, несмотря на ошибку. Суммирует результат загрузки правил. Код завершения будет ошибочным, если какое-либо правило не будет загружено.
-D	Удалить все правила и точки наблюдения. Может также принимать параметр (-k).
-e [0..2]	Установить флаг блокировки. «0» – отключить аудит, «1» – включить аудит, «2» – защитить конфигурацию аудита от изменений. Для использования данной возможности необходимо внести данную команду последней строкой в <code>audit.rules</code> . После ее выполнения все попытки изменить конфигурацию будут отвергнуты с уведомлением в журналах аудита (чтобы задействовать новую конфигурацию аудита, необходимо перезагрузить систему).
-f [0..2]	Установить способ обработки для флага сбоя: 0=silent 1=printk 2=panic. Позволяет определить, каким образом ядро будет обрабатывать критические ошибки. Например, флаг сбоя выставляется при следующих условиях: ошибки передачи в пространство службы аудита, превышение лимита буферов, ожидающих обработки, выход за пределы памяти ядра, превышение лимита скорости выдачи сообщений (значение, установленное по умолчанию: «1», для систем с повышенными требованиями к безопасности, значение «2» может быть более предпочтительно).

## Окончание таблицы 3

Опция	Описание
-h	Краткая помощь по аргументам командной строки.
-i	Игнорировать ошибки при чтении правил из файла. Если этот параметр передан в качестве аргумента (-s), то привести, если это возможно, числа к удобочитаемому виду.
--loginuid-immutable	Сделать login UID неизменяемым сразу после его установки. Для изменения login UID требуется CAP_AUDIT_CONTROL, поэтому непривилегированный пользователь не может его изменить. Установка этого параметра может вызвать проблемы в некоторых контейнерах.
-q точка- <монтирования, поддер ево>	При наличии точки наблюдения за каталогом и объединении или перемещении монтирования другого поддерева в наблюдаемое поддерево, необходимо указать ядру, сделать монтируемое поддерево эквивалентным просматриваемому каталогу. Если поддерево уже смонтировано во время создания точки наблюдения за каталогом, поддерево автоматически помечается для просмотра. Эти два значения разделяет запятая, отсутствие запятой приведет к ошибке.
-r <частота>	Установить ограничение скорости выдачи сообщений в секунду («0» – нет ограничения). В случае, если эта частота не нулевая и она превышает в ходе аудита, флаг сбоя выставляется ядром для выполнения соответствующего действия. Значение, установленное по умолчанию: «0».
--reset-lost	Сбросить счетчик потерянных записей, отображаемых командой статуса.
-R <файл>	Читать правила из файла. Правила должны быть организованы следующим образом: располагаться по одному в строке и в том порядке, в каком должны исполняться. Накладываются следующие ограничения: владельцем читаемого файла должен быть root, и доступ на чтение должен быть только у него. Файл может содержать комментарии, начинающиеся с символа «#». Правила, расположенные в файле, идентичны тем, что набираются в командной строке, без указания auditctl.
-t	Обрезать поддеревья после команды монтирования.

Т а б л и ц а 4 – Опции состояния

Опция	Описание
-l	Вывести список всех правил по одному правилу в строке. Этой команде могут быть предоставлены две опции: либо ключ фильтрации (-k), чтобы вывести список правил, соответствующих ключу, либо опцию (-i) интерпретирующую значения полей от a0 до a3, для корректного определения значений аргументов системных вызовов.
-m <текст>	Послать в систему аудита пользовательское сообщение. Команда может быть выполнена только из-под учетной записи root.
-s	Получить статус аудита. Будут показаны значения, которые можно установить с помощью опций (-e), (-f), (-r) и (-b). Значение pid – это номер процесса службы аудита. Значение pid 0 указывает, что служба аудита не работает. Поле lost сообщает, сколько записей событий аудита было отброшено из-за переполнения буфера аудита. Поле backlog сообщает, сколько записей событий аудита находится в очереди, ожидая, когда auditd прочтает их. С этим параметром можно использовать опцию (-i) для интерпретации значений некоторых полей.
-v	Вывести версию auditctl.

Т а б л и ц а 5 – Опции правил

Опция	Описание
-a <список, действие   действие, список>	Добавить правило с указанным действием к концу списка. Необходимо учитывать, что значения «список» и «действие» разделены запятой, и ее отсутствие вызовет ошибку. Поля могут быть указаны в любом порядке.
-A <список, действие>	Добавить правило с указанным действием в начало списка.
-C <f=f   f!=f>	Создать правило сравнения между полями: поле, операция, поле. Можно передавать несколько сравнений в одной командной строке. Каждое из них должно начинаться с (-C). Каждое правило сравнения добавляется друг к другу, а также к правилам, начинающимся с (-F) для инициирования записи аудита. Поддерживаются два оператора – равно и не равно. Допустимые поля: auid, uid, euid, suid, fsuid, obj_uid; и gid, egid, sgid, fsgid, obj_gid. Две группы uid и gid не могут быть смешаны. Внутри группы может быть сделано любое сравнение.
-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. Может также принимать параметр (-k).

## Продолжение таблицы 5

Опция	Описание
-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. В одной команде допускается задавать до шестидесяти четырех полей сравнения. Каждое новое поле должно начинаться с -F. Аудит будет генерировать запись, если произошло совпадение по всем полями сравнения. Допустимо использование одного из следующих восьми 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. То же самое можно сказать и про имя группы. Поля сравнения могут быть заданы для объектов, приведенных в таблице 6.
-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. Поиск значения ключа можно выполнить с помощью команды ausearch, чтобы независимо от того, какое правило вызвало событие, можно было найти его результаты. Ключ также можно использовать для удаления всех правил (-D), или правил с определенным ключом (-l). В правиле можно использовать несколько ключей, если необходимо иметь возможность поиска зарегистрированных событий несколькими способами, или если применяется плагин audispd, который в своем анализе использует ключ.
-p <r w x a>	Установить фильтр прав доступа для точки наблюдения. r=чтение, w=запись, x=исполнение, a=изменение атрибута, которые определяют типы системных вызовов, которые выполняют данные действия (системные вызовы «read» и «write» не включены в этот набор, поскольку логи аудита были бы перегружены информацией о работе этих вызовов).
-S <имя или номер системного вызова all>	В случае, если какой-либо процесс выполняет указанный системный вызов, то аудит генерирует соответствующую запись. В случае, 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. Следует писать по два правила: одно для 32-битной архитектуры (\fBb32\fp), другое для 64-битной (\fBb64\fp), чтобы убедиться, что ядро находит все ожидаемые события.

## Окончание таблицы 5

Опция	Описание
<code>-w &lt;путь&gt;</code>	Добавить точку наблюдения за файловым объектом, находящимся по указанному пути. Добавление точки наблюдения к каталогу верхнего уровня запрещено ядром. Групповые символы (wildcards) также не могут быть использованы, попытки их использования будут генерировать предупреждающее сообщение. Внутренние точки наблюдения реализованы как слежение за <code>inode</code> . Установка точки наблюдения за файлом аналогична использованию параметра <code>path</code> в правиле системного вызова <code>-F</code> . Установка точки наблюдения за каталогом аналогична использованию параметра <code>dir</code> в правиле системного вызова <code>-F</code> . Единственными допустимыми параметрами при использовании точек наблюдения являются <code>-p</code> и <code>-k</code> .
<code>-W &lt;путь&gt;</code>	Удалить точку наблюдения за файловым объектом, находящимся по указанному пути.

## Т а б л и ц а 6 – Объекты поля сравнения

Объект	Описание
<code>a0, a1, a2, a3</code>	Четыре первых аргумента, переданных системному вызову. Строковые аргументы не поддерживаются. Это связано с тем, что ядро должно получать указатель на строку, а проверка поля по значению адреса указателя не желательна. Таким образом, необходимо использовать только цифровые значения.
<code>arch</code>	Архитектура процессора, на котором выполняется системный вызов. Для определения архитектуры необходимо использовать команду: <code>uname -m</code> В случае, если архитектура ПЭВМ неизвестна, необходимо использовать таблицу 32-х битных системных вызовов, если она поддерживается ПЭВМ, можно использовать <code>b32</code> . Аналогичным образом применяется таблица системных вызовов <code>b64</code> . Можно написать правила, которые в некоторой степени не зависят от архитектуры, потому что тип будет определяться автоматически. Однако системные вызовы могут зависеть от архитектуры, и то, что доступно на <code>x86_64</code> , может быть недоступно на PPC. Директива <code>arch</code> должна предшествовать опции <code>-S</code> , чтобы <code>auditctl</code> знал, какую внутреннюю таблицу использовать для поиска номеров системных вызовов.
<code>auid</code>	Идентификатор пользователя, использованный для входа в систему. Можно использовать либо имя пользователя, либо идентификатор пользователя.
<code>devmajor</code>	Главный номер устройства (Device Major Number).
<code>devminor</code>	Вспомогательный номер устройства (Device Minor Number).

## Окончание таблицы 6

Объект	Описание
dir	Полный путь к каталогу для создания точки наблюдения. Помещает точку наблюдения в каталог и рекурсивно во все его поддеревья. Можно использовать только в списке exit.
egid	Действительный идентификатор группы.
euid	Действительный идентификатор пользователя.
exe	Абсолютный путь к приложению, к которому будет применяться это правило. Можно использовать только в списке exit.
exit	Значение, возвращаемое системным вызовом при выходе.
fsgid	Идентификатор группы, применяемый к файловой системе.
fsuid	Идентификатор пользователя, применяемый к файловой системе.
filetype	Тип целевого файла: файл, каталог, сокет, ссылка, символ, блок или FIFO.
gid	Идентификатор группы.
inode	Номер inode.
key	Альтернативный способ установить ключ фильтрации.
msgtype	Используется для проверки совпадения с числом, описывающим тип сообщения. Может использоваться только в списках exclude и user.
obj_uid	UID объекта.
obj_gid	GID объекта.
path	Полный путь к файлу для точки наблюдения. Может использоваться только в списке exit.
perm	Фильтр прав доступа для файловых операций. Может использоваться только в списке exit. Можно использовать без указания системного вызова, при этом ядро выберет системные вызовы, которые удовлетворяют запрашиваемым разрешениям.
pers	Персональный номер операционной системы.
pid	Идентификатор процесса.
ppid	Идентификатор родительского процесса.
sessionid	Идентификатор сеанса пользователя.
sgid	Установленный идентификатор группы.
success	Если значение, возвращаемое системным вызовом, больше либо равно 0, данный объект будет равен «true»/«yes», иначе «false»/«no». При создании правила нужно использовать 1 вместо «true»/«yes» и 0 вместо «false»/«no».
suid	Установленный идентификатор пользователя.
uid	Идентификатор пользователя.



Для добавления правил используется следующая форма записи команды `auditctl`:

```
# auditctl -a список,действие -S имя_системного_вызова -F фильтры
```

Здесь список – это список событий, в который следует добавить правило.

Далее описаны имена доступных списков:

- 1) `task` – добавить правило к списку, отвечающему за процессы. Этот список правил используется только во время создания процесса – когда родительский процесс вызывает `fork()` или `clone()`. При использовании этого списка можно использовать только те поля, которые известны во время создания процесса (`uid`, `gid` и так далее);
- 2) `exit` – добавить правило к списку, отвечающему за точки выхода из системных вызовов. Этот список применяется, когда необходимо создать событие для аудита, привязанное к точкам выхода из системных вызовов;
- 3) `user` – добавить правило, отвечающее за список фильтрации пользовательских сообщений. Этот список используется ядром, чтобы отфильтровать события, приходящие из пользовательского пространства, перед тем, как они будут переданы службе аудита. Необходимо отметить, что только следующие поля могут быть использованы: `uid`, `auid`, `gid`, `pid`, `subj_user`, `subj_role`, `subj_type`, `subj_sen`, `subj_clr` и `msgtype`. Все остальные поля будут обработаны, как если бы они не совпали;
- 4) `exclude` – добавить правило к списку, отвечающего за фильтрацию событий определенного типа. Этот список используется, чтобы отфильтровывать ненужные события. События могут быть исключены по идентификатору процесса, идентификатору пользователя, идентификатору группы, идентификатору логина пользователя, типу сообщения или контексту предмета.

Второй параметр опции `-a` – это действие, которое должно произойти в ответ на возникшее событие:

- 1) `never` – аудит не будет генерировать никаких записей. Может быть использовано для подавления генерации событий. Обычно необходимо подавлять генерацию вверху списка, а не внизу, поскольку событие инициируется на первом совпавшем правиле;
- 2) `always` – установить контекст аудита. Всегда заполнять его во время входа в системный вызов, и всегда генерировать запись во время выхода из системного вызова.

Далее указывается опция `-s`, которая задает имя системного вызова, при обращении к которому должен срабатывать триггер (например, `open`, `close`, `exit`). Вместо имени может быть использовано числовое значение.

Необязательная опция `-F` используется для указания дополнительных параметров фильтрации события.

Примеры использования:

- для ведения журнала событий, связанных с использованием системного вызова `open()`, и регистрирования при этом только обращения к файлам каталога `/etc`, используется следующее правило:

```
# auditctl -a always,exit -S open -F path=/etc/
```

- регистрировать только те события, при которых файл открывается только на запись и изменение атрибутов:

```
# auditctl -a always,exit -S open -F path=/etc/ -F perm=aw
```

- записывать все системные вызовы, используемые определенным процессом:

```
auditctl -a always,exit -S all -F pid=1005
```

- записывать все файлы, открытые определенным пользователем:

```
auditctl -a always,exit -S openat -F auid=510
```

- записывать неудачные попытки вызова системной функции 'openat':

```
auditctl -a exit,always -S openat -F success!=0
```

- записывать попытки изменения файла (два способа):

```
auditctl -w /etc/shadow -p wa
```

```
auditctl -a always,exit -F path=/etc/shadow -F perm=wa
```

Правила не обязательно задавать, используя командную строку. Во время старта служба `auditd` читает файл `/etc/audit/audit.rules`, который содержит правила аудита в формате `auditctl`.

В файл записываются правила без имени команды. Например:

```
-w /etc/passwd -p wa
```

#### 3.8.2.4. Команда `aureport`

Команда `aureport` генерирует итоговые отчеты на основе логов службы аудита, также может принимать данные со стандартного ввода (`stdin`) до тех пор, пока на входе будут необработанные данные логов. В шапке каждого отчета для каждого столбца есть заголовок. Все отчеты, кроме основного итогового отчета, содержат номера событий аудита. Используя их, можно найти полные данные о событии с помощью команды `ausearch -a <номер события>`. В случае, если в отчете слишком много данных, можно задать время начала и время окончания для уточнения временного промежутка.

Отчеты, генерируемые `aureport`, могут быть использованы как исходный материал для получения развернутых отчетов.

Синтаксис:

```
aureport [опции]
```

Опции команды `aureport` приведены в таблице 7.

Т а б л и ц а 7 – Опции команды `aureport`

Опция	Описание
<code>-au, --auth</code>	Отчет о попытках аутентификации.
<code>-a, --avc</code>	Отчет о авс сообщениях.
<code>--comm</code>	Отчет о выполнении команд.
<code>-c, --config</code>	Отчет об изменениях конфигурации.
<code>-cr, --crypto</code>	Отчет о событиях, связанных с кодированием.
<code>-e, --event</code>	Отчет о событиях.

Опция	Описание
<code>--escape &lt;опция&gt;</code>	Экранировать вывод. Возможные значения: raw, tty, shell и shell_quote. Каждый режим включает в себя символы предыдущего режима и экранирует больше символов. То есть shell включает все символы, экранируемые tty, и добавляет новые. Значение по умолчанию – tty.
<code>-f, --file</code>	Отчет о файлах и сокетах.

*Продолжение таблицы 7*

Опция	Описание
<code>--failed</code>	Для обработки в отчетах выбирать только неудачные события. По умолчанию показываются как удачные, так и неудачные события.
<code>-h, --host</code>	Отчет о хостах.
<code>-i, --interpret</code>	Транслировать числовые значения в текстовые (например, идентификатор пользователя будет транслирован в имя пользователя). Трансляция выполняется с использованием данных с той машины, где запущен «aureport».
<code>-if, --input &lt;файл&gt;   &lt;каталог&gt;</code>	Использовать указанный файл или каталог вместо логов аудита. Это может быть полезно при анализе логов с другой машины или при анализе частично сохраненных логов.
<code>--input-logs</code>	Использовать местоположение файла журнала из auditd.conf как исходные данные для анализа. Это необходимо, при использовании команды aureport в задании cron.
<code>--integrity</code>	Отчет о событиях целостности.
<code>-k, --key</code>	Отчет о ключевых словах в правилах.
<code>-l, --login</code>	Отчет о попытках входа в систему.
<code>-m, --mods</code>	Отчет об изменениях пользовательских учетных записей.
<code>-n, --anomaly</code>	Отчет об аномальных событиях. Эти события включают переход сетевой карты в беспорядочный режим и ошибки сегментации.

Опция	Описание
--node <имя узла>	Отобразить в отчете только события со строкой <имя узла>. По умолчанию включены все узлы. Допускается перечисление нескольких узлов.
-nc, --no-config	Не включать событие CONFIG_CHANGE. Это особенно полезно для ключевого отчета, поскольку правила аудита во многих случаях имеют ключевые метки. Использование этой опции избавляет от ложных срабатываний.
-p, --pid	Отчет о процессах.
-r, --response	Отчет о реакциях на аномальные события.
-s, --syscall	Отчеты о системных вызовах.

*Окончание таблицы 7*

Опция	Описание
--success	Для обработки в отчетах выбирать только удачные события. По умолчанию показываются как удачные, так и неудачные события.
--summary	Генерировать итоговый отчет, который дает информацию только о количестве элементов в том или ином отчете. Такой режим есть не у всех отчетов.
-t, --log	Генерация отчетов о временных рамках каждого отчета.
--tty	Отчеты о нажатых клавишах.
-te, --end <дата> <время>	Искать события, которые произошли раньше (или во время) указанной временной точки. Формат даты и времени зависит от региональных настроек. В случае, если дата не указана, то подразумевается текущий день («today»). В случае, если не указано время, то подразумевается текущий момент («now»).
-tm, --terminal	Отчет о терминалах.
-ts, --start <дата> <время>	Искать события, которые произошли после (или во время) указанной временной точки. Формат даты и времени зависит от региональных настроек. В случае, если дата не указана, то подразумевается текущий день (today). В случае, если не указано время, то подразумевается полночь (midnight).
-u, --user	Отчет о пользователях.

Опция	Описание
-v, --version	Вывести версию программы и выйти.
-x, --executable	Отчет об исполняемых объектах.

**Примечание.** Необходимо использовать нотацию времени в формате «24 часа», а не «AM/PM». Например, дата может быть задана как «10/24/2005», а время – как «18:00:00».

Также допускается использовать следующие ключевые слова:

- now – сейчас;
- recent – десять минут назад;
- boot – время за секунду до того, когда система загружалась в последний раз;
- today – первая секунда после полуночи текущего дня;
- yesterday – первая секунда после полуночи предыдущего дня;
- this-week – первая секунда после полуночи первого дня текущей недели, первый день недели определяется из региональных настроек;
- week-ago – первая секунда после полуночи ровно 7 дней назад;
- this-month – первая секунда после полуночи первого числа текущего месяца;
- this-year – первая секунда после полуночи первого числа первого месяца текущего года.

#### 3.8.2.5. Команда ausearch

Команда `ausearch` является инструментом поиска по журналу аудита. `ausearch` может также принимать данные со стандартного ввода (`stdin`) до тех пор, пока на входе будут необработанные данные логов. Все условия, указанные в параметрах, объединяются логическим «И».

Каждый системный вызов ядра из пользовательского пространства и возвращение данных в пользовательское пространство имеет один уникальный (для каждого системного вызова) идентификатор события.

Различные части ядра могут добавлять дополнительные записи. Например, в событие аудита для системного вызова «open» добавляется запись `PATH` с именем

файла. `ausearch` показывает все записи события вместе. Это означает, что при запросе определенных записей результат может содержать записи `SYSCALL`.

Не все типы записей содержат указанную информацию. Например, запись `PATH` не содержит имя узла или `loginuid`.

**Синтаксис:**

`ausearch [опции]`

Опции команды `ausearch` приведены в таблице 8.

Т а б л и ц а 8 – Опции команды ausearch

Опция	Описание
-a, --event <идентификатор события>	Искать события с заданным идентификатором события. В сообщении: msg=audit(1116360555.329:2401771), идентификатор события – число после «:». Все события аудита, связанные с одним системным вызовом, имеют одинаковый идентификатор.
--arch <CPU>	Искать события на основе определенной архитектуры процессора. Для определения архитектуры необходимо использовать команду: uname -m В случае, если архитектура ПЭВМ неизвестна, необходимо использовать таблицу 32-х битных системных вызовов, если она поддерживается ПЭВМ, можно использовать b32. Аналогичным образом применяется таблица системных вызовов b64.
-c, --comm <comm-name>	Искать события с заданным «comm name», именем исполняемого файла из структуры задачи.
--debug	Вывести сообщения, пропущенные stderr.
--checkpoint <файл контрольной точки>	Контрольная точка – это вывод между последовательными вызовами ausearch, так что в последующих вызовах будут выводиться только события, не попавшие в предыдущий вывод. Событие auditd состоит из одной или нескольких записей. При обработке события ausearch определяет события как завершенные и незавершенные. Завершенное событие – это одно событие записи или то, которое произошло раньше, чем за 2 секунды по сравнению с текущим обрабатываемым событием. Контрольная точка обеспечивается путем записи последнего завершенного события вывода вместе с номером устройства и индексом файла последнего завершенного события в файл контрольной точки. При следующем вызове ausearch загрузит данные контрольной точки и при обработке файлов журнала, будет отбрасывать все завершенные события, пока они не соответствуют контрольной точке, в этот момент ausearch начнет выводить события.
-e, --exit <код>	Искать события на основе кода системного вызова exit или errno.
--escape <опция>	Экранировать вывод. Возможные значения: raw, tty, shell и shell_quote. Каждый режим включает в себя символы предыдущего режима и экранирует больше символов. То есть shell включает все символы, экранируемые tty, и добавляет новые. Значение по умолчанию – tty.



Опция	Описание
--extra-keys	<p>Если параметр <code>format</code> имеет значение <code>csv</code>, вывести столбец с дополнительной информацией.</p> <p>Работает только с записями SYSCALL, которые были записаны в результате запуска правила аудита, определенного ключом.</p>

## Продолжение таблицы 8

Опция	Описание
--extra-labels	Если параметр <code>format</code> имеет значение <code>csv</code> , добавить информацию о метках субъекта и объекта (если метки существуют).
--extra-obj2	Если параметр <code>format</code> имеет значение <code>csv</code> , добавить информацию о втором объекте (если он существует). Второй объект иногда является частью записи, например, при переименовании файла или монтировании устройства.
--extra-time	Если параметр <code>format</code> имеет значение <code>csv</code> , добавить информацию о времени простоя.
-f, --file <файл>	Искать события с заданным именем файла.
--format <опции>	Отформатировать события, которые соответствуют критериям поиска. Поддерживаемые форматы: <code>raw</code> , <code>default</code> , <code>interpret</code> , <code>csv</code> и <code>text</code> . Значение <code>raw</code> описано в опции <code>raw</code> . При значении <code>default</code> строки выводятся без форматирования, в выводе используется одна строка в качестве визуального разделителя, далее указывается метка времени, а затем следуют записи события. Значение <code>interpret</code> объясняется в описании опции <code>-i</code> . При значении <code>csv</code> результат поиска выводится в формате CSV (от англ. Comma-Separated Values – значения, разделенные запятыми). Значение <code>text</code> преобразует вывод к формату предложений, что упрощает понимание вывода, но происходит это за счет потери деталей.
-ga, --gid-all all- <идентификатор группы>	Искать события с заданным эффективным или обычным идентификатором группы.
-ge, --gid-effective <эффективный идентификатор группы>	Искать события с заданным эффективным идентификатором группы или именем группы.
-gi, --gid <группа>	Искать события с заданным идентификатором группы или именем группы.
-h, --help	Справка.
-hn, --host <имя узла>	Искать события с заданным именем узла. Имя узла может быть именем узла, полным доменным именем или цифровым сетевым адресом.
-i, --interpret	Транслировать числовые значения в текстовые. Например, идентификатор пользователя будет транслирован в имя пользователя. Трансляция выполняется с использованием данных с той машины, где запущен «ausearch».
-if, --input <файл> <каталог>	Использовать указанный файл или каталог вместо логов аудита. Это может быть полезно при анализе логов с другой машины или при анализе частично сохраненных логов.

## Продолжение таблицы 8

Опция	Описание
--input-logs	Использовать местоположение файла журнала из auditd.conf как исходные данные для анализа. Применяется при использовании команды «ausearch» в задании cron.
--just-one	Остановиться после выдачи первого события, соответствующего критериям поиска.
-k, --key <ключевое слово>	Искать события с заданным ключевым словом.
-l, --line-buffered	Сбрасывать вывод после каждой строки.
-m, --message <тип>   <список типов>	Искать события с заданным типом, при этом можно указать список значений, разделенных запятыми. Можно указать несуществующий в событиях тип «ALL», который позволяет получить все сообщения системы аудита (список допустимых типов будет показан, если указать эту опцию без значения). Тип сообщения может быть строкой или числом. В списке значений этого параметра в качестве разделителя используются запятые и пробелы недопустимы.
-n, --node	Искать события с определенным узла. Допускается указание нескольких узлов (для вывода достаточно совпадение любого узла).
-p, --pid <идентификатор процесса>	Искать события с заданным идентификатором процесса.
-pp, --ppid <идентификатор процесса>	Искать события с заданным идентификатором родительского процесса.
-r, --raw	Необработанный вывод. Используется для извлечения записей для дальнейшего анализа.
-sc, --success <системный вызов>	Искать события с заданным системным вызовом. Можно указать номер или имя системного вызова. При указании имени системного вызова, оно будет проверено по таблице системных вызовов на машине, где запущен ausearch.
--session <идентификатор сеанса>	Искать события с заданным идентификатором сеанса. Этот атрибут устанавливается, когда пользователь входит в систему, и может связать любой процесс с определенным именем пользователя.
-sv, --success <флаг>	Искать события с заданным флагом успешного выполнения. Допустимые значения: «yes» (успешно) и «no» (неудачно).
-te, --end <дата> <время>	Искать события, которые произошли раньше (или во время) указанной временной точки. Формат даты и времени зависит от региональных настроек. В случае, если дата не указана, то подразумевается текущий день («today»). В случае, если не указано время, то подразумевается текущий момент («now»).

-ts, --start <дата> <время>	Искать события, которые произошли после (или во время) указанной временной точки.
--------------------------------	---

Окончание таблицы 8

Опция	Описание
-tm, --terminal <терминал>	Искать события с заданным терминалом. Некоторые службы (такие как cron и atd) используют имя службы как имя терминала.
-ua, --uid-all <идентификатор пользователя>	Искать события, у которых любой из идентификатора пользователя, эффективный идентификатор пользователя или loginuid (auid) совпадают с заданным идентификатором пользователя.
-ue, --uid- <эффективный идентификатор пользователя>	Искать события с заданным эффективным идентификатором пользователя.
-ui, --uid <идентификатор пользователя>	Искать события с заданным идентификатором пользователя.
-ul, --loginuid <идентификатор пользователя>	Искать события с заданным идентификатором пользователя. Все программы, которые его используют, должны использовать pam_loginuid.
-uu, --uuid <идентификатор гостя>	Искать события с заданным идентификатором гостя.
-v, --verbose	Показать версию и выйти.
--vm, --vm-name<имя гостя>	Искать события с заданным именем гостя.
-x, --executable <программа>	Искать события с заданным именем исполняемой программы.

**Примечание.** Необходимо использовать нотацию времени в формате «24 часа», а не «AM/PM». Например, дата может быть задана как «10/24/2005», а время – как «18:00:00».

Также допускается использовать следующие ключевые слова:

- now – сейчас;
- recent – десять минут назад;
- boot – время за секунду до того, когда система загружалась в последний раз;
- today – первая секунда после полуночи текущего дня;
- yesterday – первая секунда после полуночи предыдущего дня;
- this-week – первая секунда после полуночи первого дня текущей недели, первый день недели определяется из региональных настроек;

- week-ago – первая секунда после полуночи ровно 7 дней назад;
- this-month – первая секунда после полуночи первого числа текущего месяца;
- this-year – первая секунда после полуночи первого числа первого месяца текущего года.

### 3.8.2.6. Команда `autrace`

Команда `autrace` добавляет правила аудита для того, чтобы следить за использованием системных вызовов в указанном процессе подобно тому, как это делает `strace`.

После добавления правил, команда `autrace` запускает процесс с указанными аргументами. Результаты аудита будут либо в логах аудита (если служба аудита запущена), либо в системных логах. `autrace` устроена так, что удаляет все предыдущие правила аудита, перед тем как запустить указанный процесс и после его завершения. Поэтому, в качестве дополнительной меры предосторожности, программа не запустится, если перед ее использованием правила не будут удалены с помощью `audtictl` – об этом известит предупреждающее сообщение.

Синтаксис:

```
autrace [-r] процесс [аргументы]
```

Опция `-r` позволяет ограничить сбор информации о системных вызовах только теми, которые необходимы для анализа использования ресурсов. Это может быть полезно при моделировании внештатных ситуаций, к тому же позволяет уменьшить нагрузку на логи.

Примеры использования:

- обычное использование программы:

```
autrace /bin/ls /tmp
ausearch --start recent -p 2442 -i
```

- режим ограниченного сбора информации:

```
autrace -r /bin/ls
ausearch --start recent -p 2450 --raw | aureport --file --summary
ausearch --start recent -p 2450 --raw | aureport --host --summary
```

### 3.8.2.7. Настройка ротации логов аудита

Правила ротации логов аудита настраиваются в файле `/etc/audit/auditd.conf`.

Например, для того, чтобы при нехватке места на диске старые записи затирались новыми, необходимо отредактировать файл `/etc/audit/auditd.conf`:

```
max_log_file = 8
space_left = 100
space_left_action = ROTATE
```

где:

- `max_log_file` – максимальный размер файла журнала в Мбайт;
- `space_left` – минимум свободного пространства в Мбайт;
- `space_left_action` – действие (в данном случае старые файлы журналов будут удаляться, освобождая место для новых).

После редактирования файла, для того чтобы новые настройки вступили в силу, необходимо перезапустить `auditd`:

```
# /etc/init.d/auditd restart
```

### 3.8.2.8. Ротация логов сервиса `auditd` по расписанию с использованием `crond`

Для настройки ротации логов сервиса `auditd` по расписанию с использованием `crond` необходимо:

- 1) в файле `/etc/audit/auditd.conf` опцию `max_log_file_action` установить в значение `IGNORE` (опция `IGNORE`, отключает контроль за размером файла лога):

```
# vim /etc/audit/auditd.conf
max_log_file_action = IGNORE
```

- 2) для применения данной конфигурации необходимо выполнить следующую команду:

```
# service auditd restart
```

- 3) проверить состояния сервиса `auditd` выполнив следующую команду:

```
# systemctl status auditd | grep running
```

- 4) просмотреть содержимое файла `/etc/crontab`, оно должно быть следующим:

```
# vim /etc/crontab
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

5) для задания частоты ротации логов сервиса `auditd` необходимо создать скрипт `logauditd` в каталоге `/etc/cron.frequency`, где `frequency` – часть имени каталога, обозначающая частоту выполнения скрипта.

Например, для ежедневной ротации логов сервиса `auditd` следует создать в каталоге `/etc/cron.daily/` скрипт с именем `logauditd`:

```
#!/bin/sh
/sbin/service auditd rotate
EXITVALUE=$?
if [ $EXITVALUE != 0 ]; then
    /usr/bin/logger -t auditd "Выход с кодом: [$EXITVALUE]"
fi
exit 0
```

Часть имени каталога `frequently` необходимо изменить на `hourly`, `daily`, `weekly`, `monthly`, как указано в файле `/etc/crontab`, указав тем самым частоту выполнения скрипта (см. шаг 4)). Также, в файле `/etc/crontab` можно добавить любую другую частоту для ротации логов.

6) установить права на выполнение скрипту `logauditd`:

```
# chmod +x /etc/cron.daily/logauditd
```

7) запустить сервис `crond`:

```
# systemctl start crond
```

8) удостовериться, что сервис `crond` успешно запущен, выполнив команду:

```
# systemctl status crond | grep running
```

В результате проделанных действий должна производиться запись файла лога сервиса `auditd` с заданной частотой. Размер файла лога зависит от количества событий, накопленных за указанный период времени.

### 3.8.3. Использование аудита

Далее приводятся примеры аудита событий, которые фиксирует КСЗ ОС.

В записях журнала аудита содержится следующая информация: дата и время события, тип и содержание события, идентификатор субъекта доступа (например, uid=500 gid=500) и результат события (успешный/неуспешный).

### 3.8.3.1. События запуска и завершения аудита

Команда для поиска записей аудита, связанных с запуском и завершением функции аудита:

```
# ausearch -m DAEMON_START -m DAEMON_END
----
time->Fri Aug  3 13:35:19 2024
type=DAEMON_START msg=audit(1533292519.132:6245): op=start
ver=2.7.2 format=raw kernel=4.4.86-std-def-alt0.M80C.1
aid=4294967295 pid=378 uid=0 ses=4294967295 res=success
----
time->Fri Aug  3 18:03:41 2024
type=DAEMON_END msg=audit(1533308621.816:6246): op=terminate
aid=0 pid=1 subj=.86-std-def-alt0.M80C.1 aid=4294967295 pid=378
uid=0 ses=4294967295 res=success res=success
----
time->Mon Aug  6 11:32:01 2024
type=DAEMON_START msg=audit(1533544321.724:9778): op=start
ver=2.7.2 format=raw kernel=4.4.86-std-def-alt0.M80C.1
aid=4294967295 pid=371 uid=0 ses=4294967295 res=success
----
time->Mon Aug  6 18:05:40 2024
type=DAEMON_END msg=audit(1533567940.159:9779): op=terminate
aid=0 pid=1 subj=.86-std-def-alt0.M80C.1 aid=4294967295 pid=371
uid=0 ses=4294967295 res=success res=success
----
```

### 3.8.3.2. События, связанные с модификацией конфигурации

Команда для поиска записей аудита, связанных с модификацией конфигурации:

```
# ausearch -m config_change
time->Tue Aug 21 10:08:08 2024
type=CONFIG_CHANGE msg=audit(1534835288.836:126): aid=0 ses=4
op="add_rule" key=(null) list=4 res=1
----
time->Tue Aug 21 10:08:50 2024
type=CONFIG_CHANGE msg=audit(1534835330.507:129): aid=0 ses=4
op="add_rule" key=(null) list=4 res=1
----
```



Команды создания и поиска событий, связанных с модификацией конфигурации аудита:

```
# auditctl -w /etc/audit -p w -k audit_config
# ausearch -k audit_config
```

### 3.8.3.3. События, связанные с операцией чтения записей аудита

Команды создания и поиска событий, связанных с операцией чтения записей аудита:

```
# auditctl -w /var/log/audit/ -p r -k audit_log
# ausearch -i -k audit_log
```

----

```
type=PROCTITLE msg=audit(21.08.2024 16:41:44.439:79) :
proctitle=ausearch -i -k audit_log
type=PATH msg=audit(21.08.2024 16:41:44.439:79) : item=0
name=/var/log/audit/audit.log inode=528278 dev=08:02
mode=file,600 ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=CWD msg=audit(21.08.2024 16:41:44.439:79) : cwd=/root
```

```
type=SYSCALL msg=audit(21.08.2024 16:41:44.439:79) : syscall=open
success=yes exit=4 a0=0x7ffecf5d12cb a1=O_RDONLY a2=0x0 a3=0x169
items=1 ppid=1847 pid=1995 auid=test uid=root gid=root euid=root
suid=root fsuid=root egid=root sgid=root fsgid=root tty=pts0
ses=2 comm=ausearch exe=/sbin/ausearch key=audit_log
```

----

```
type=PROCTITLE msg=audit(12.09.2024 08:06:49.118:159) :
proctitle=ausearch -m DAEMON_START
type=PATH msg=audit(12.09.2024 08:06:49.118:159) : item=0
name=/var/log/audit/audit.log.4 inode=136304 dev=08:02
mode=file,400 ouid=root ogid=root rdev=00:00
obj=generic_u:object_r:unlabeled:s0 nametype=NORMAL
type=CWD msg=audit(12.09.2024 08:06:49.118:159) : cwd=/root
type=SYSCALL msg=audit(12.09.2024 08:06:49.118:159) :
syscall=open success=yes exit=3 a0=0x621c50 a1=O_RDONLY a2=0x1b6
a3=0x0 items=1 ppid=3839 pid=3879 auid=user uid=root gid=root
euid=root suid=root fsuid=root egid=root sgid=root fsgid=root
tty=pts0 ses=2 comm=ausearch exe=/sbin/ausearch
subj=generic_u:generic_r:generic_t:s0 key=audit_log
```

Попытаемся прочитать данные аудита напрямую из файла /var/log/audit от имени обычного пользователя:

```
$ cat /var/log/audit/audit.log
```

```
cat: /var/log/audit/audit.log: Отказано в доступе
```

Команда для поиска записей аудита, связанных с успешными/неуспешными попытками чтения записей аудита:

```
# ausearch -i -k open
```

### 3.8.3.4. События, связанные со службой аудита

Аудит регистрирует события следующего типа:

- DAEMON\_ERR – служба аудита остановилась из-за внутренней ошибки;
- DAEMON\_RESUME – служба аудита возобновила ведение журнала;
- DAEMON\_ROTATE – произошла ротация файлов журнала аудита;
- DAEMON\_ABORT – служба аудита остановилась из-за ошибки.

Команда для поиска записей аудита, связанных, например, с ротацией файлов журнала аудита:

```
# ausearch -m DAEMON_ROTATE
```

```
----
```

```
time->Wed Sep 12 11:40:32 2024
```

```
type=DAEMON_ROTATE msg=audit(1536741632.074:5567): op=rotate-logs
aid=? pid=? subj=?
```

```
----
```

```
time->Wed Sep 12 11:47:03 2024
```

```
type=DAEMON_ROTATE msg=audit(1536742023.793:5569): op=rotate-logs
aid=? pid=? subj=?
```

3.8.3.5. События, связанные с запросами на выполнение операций на объекте доступа

Примеры команд создания правила и поиска записей аудита, связанных с запросами на выполнение операций на объекте доступа:

```
# auditctl -w=/path/to/dir -k FDP_ACC
# ausearch -k FDP_ACC
```

Например, настроим наблюдение за каталогом /testdir:

```
auditctl -w /testdir -p rwa -k FDP_ACC
```

От имени пользователя test выполним команды:

```
$ ls /testdir
$ echo "">/testdir/123
```

Пример команды поиска неуспешных событий:

```
# ausearch -k FDP_ACC -sv no
```

## ЛКНВ.11102-01 99 01

```

----
time->Wed Sep 12 12:37:53 2024
type=PROCTITLE msg=audit(1536745073.268:115): proctitle="bash"
type=PATH msg=audit(1536745073.268:115): item=1
name="/testdir/123" nametype=CREATE
type=PATH msg=audit(1536745073.268:115): item=0 name="/testdir/"
inode=531918 dev=08:02 mode=040755 ouid=0 ogid=0 rdev=00:00
nametype=PARENT
type=CWD msg=audit(1536745073.268:115): cwd="/home/test"
type=SYSCALL msg=audit(1536745073.268:115): arch=c000003e
syscall=2 success=no exit=-13 a0=6c9bf0 a1=241 a2=1b6 a3=0
items=2 ppid=1930 pid=2048 auid=501 uid=501 gid=501 euid=501
suid=501 fsuid=501 egid=501 sgid=501 fsgid=501 tty=pts1 ses=2
comm="bash" exe="/bin/bash" key="FDP_ACC"
----

```

**Пример команды поиска успешных событий:**

```

# ausearch -k FDP_ACC -sv yes
----
time->Wed Sep 12 12:44:14 2024
type=PROCTITLE msg=audit(1536745454.174:121):
proctitle=6C73002D2D636F6C6F723D6175746F002F746573746469722F
type=PATH msg=audit(1536745454.174:121): item=0 name="/testdir/"
inode=531918 dev=08:02 mode=040755 ouid=0 ogid=0 rdev=00:00
nametype=NORMAL
type=CWD msg=audit(1536745454.174:121): cwd="/home/test"
type=SYSCALL msg=audit(1536745454.174:121): arch=c000003e
syscall=2 success=yes exit=3 a0=622ad0 a1=90800 a2=0 a3=507
items=1 ppid=2048 pid=2134 auid=501 uid=501 gid=501 euid=501
suid=501 fsuid=501 egid=501 sgid=501 fsgid=501 tty=pts1 ses=2
comm="ls" exe="/bin/ls" key="FDP_ACC"

```

**3.8.3.6. События, связанные с экспортом информации**

**Команды создания правила и поиска записей аудита, связанных с попытками экспортировать информацию:**

```

# auditctl -a always,exit -S open,openat
# ausearch -x /usr/bin/rsync | head
----
time->Wed Sep 12 13:10:33 2024
type=PROCTITLE msg=audit(1536747033.889:919):
proctitle=7273796E63002D2D666F726365002D2D69676E6F72652D6572726F7
273002D2D64656C657465002D2D64656C6574652D6578636C75646564002D2D65
78636C7564652D66726F6D3D2F6574632F6261636B75702D6578636C7564652E6
C7374002D2D6261636B7570002D6171525358002F686F6D65002F7661722F6261
636B75
type=PATH msg=audit(1536747033.889:919): item=0
name=686F6D652F757365722FD0A0D0B0D0B1D0BED187D0B8D0B920D181D182D0

```

```
BED0BB inode=532446 dev=08:02 mode=040755 ouid=500 ogid=500
rdev=00:00 nametype=NORMAL
type=CWD msg=audit(1536747033.889:919): cwd="/var/backup/latest"
type=SYSCALL msg=audit(1536747033.889:919): arch=c000003e
syscall=2 success=yes exit=0 a0=7ffcfbf44a90 a1=90800
a2=7ffcfbf44a90 a3=0 items=1 ppid=2385 pid=2386 auid=501 uid=0
gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0 ses=2
comm="rsync" exe="/usr/bin/rsync" key=(null)
```

### 3.8.3.7. События, связанные с информационными потоками

Межсетевой экран может учитывать решения, принимаемые по информационным потокам при помощи специальной цели AUDIT:

```
# iptables -A INPUT -j AUDIT --type=ACCEPT
```

Параметр `type` определяет значение поля `action` в журнале аудита (ACCEPT=0, DROP=1, REJECT=2).

Команда для просмотра записей аудита:

```
# ausearch -m NETFILTER_PKT
---
time->Wed Sep 12 10:40:33 2024
type=NETFILTER_PKT msg=audit(1536741633.470:287): action=0 hook=1
len=84 inif=eth0 outif=? smac=00:1c:f0:cc:55:dd
dmac=08:00:27:0b:b3:75 macproto=0x0800 saddr=8.8.8.8
daddr=192.168.3.182 ipid=23860 proto=1 icmp_type=0 icmpcode=0
----
time->Wed Sep 12 10:40:32 2024
type=NETFILTER_PKT msg=audit(1536741632.470:286): action=0 hook=1
len=84 inif=eth0 outif=? smac=00:1c:f0:cc:55:dd
dmac=08:00:27:0b:b3:75 macproto=0x0800 saddr=8.8.8.8
daddr=192.168.3.182 ipid=23606 proto=1 icmp_type=0 icmpcode=0
```

3.8.3.8. События, связанные с достижением ограничения неуспешных попыток аутентификации

В файле `/etc/pam.d/login` должна быть строка:

```
auth required pam_faillock.so authfail deny=n even_deny_root unlock_time=7200
```

где `n` – число попыток.

Команда для поиска записей, связанных с достижением ограничения неуспешных попыток аутентификации:

```
# ausearch -i -m RESP_ACCT_LOCK -m ANOM_LOGIN_FAILURES
----
time->Tue Jun 5 14:09:05 2024
```

## ЛКНВ.11102-01 99 01

```

type=ANOM_LOGIN_FAILURES msg=audit(1528200545.665:1362):
pid=28597 uid=0 auid=4294967295 ses=4294967295
subj=generic_u:generic_r:login_t:s0 msg='pam_faillock uid=500
exe="/usr/sbin/sshd" hostname=192.168.3.191 addr=192.168.3.191
terminal=ssh res=success'
----
time->Tue Jun 5 14:09:05 2024
type=RESP_ACCT_LOCK msg=audit(1528200545.665:1363): pid=28597
uid=0 auid=4294967295 ses=4294967295
subj=generic_u:generic_r:login_t:s0 msg='pam_faillock uid=500
exe="/usr/sbin/sshd" hostname=192.168.3.191 addr=192.168.3.191
terminal=ssh res=success'
----
time->Wed Jun 6 13:02:04 2024
type=ANOM_LOGIN_FAILURES msg=audit(1528282924.132:102): pid=2126
uid=0 auid=4294967295 ses=4294967295
subj=generic_u:generic_r:login_t:s0 msg='pam_faillock uid=500
exe="/usr/sbin/lightdm" hostname=? addr=? terminal=:0
res=success'
type=RESP_ACCT_LOCK msg=audit(10.09.2024 17:05:34.181:156) :
pid=3130 uid=root auid=unset ses=unset msg='pam_faillock uid=test
exe=/usr/sbin/lightdm hostname=? addr=? terminal=:0 res=success'
----
type=RESP_ACCT_LOCK msg=audit(10.09.2024 17:11:17.368:178) :
pid=3211 uid=root auid=unset ses=unset msg='pam_faillock uid=test
exe=/bin/login hostname=localhost addr=127.0.0.1
terminal=/dev/tty3 res=success'

```

**Команда поиска событий разблокировки пользователя (pam\_faillock -r - user test) с типом USER\_ACCT и msg=pam\_faillock:**

```

# ausearch -m USER_ACCT -x pam_faillock
----
time->Wed Sep 12 14:09:29 2024
type=USER_ACCT msg=audit(1536750569.586:21693): pid=2886 uid=0
auid=0 ses=8 msg='pam_faillock uid=503 reset=0
exe="/sbin/pam_faillock" hostname=host-105.localdomain addr=?
terminal=/dev/tty2 res=success'

```

**3.8.3.9. События, связанные с отклонением или принятием ФБО любого проверенного секрета и с изменениями заданных метрик качества**

В файле /etc/passwdqc.conf указывается число попыток ввода пароля для пользователей.

Команда создания правила для записей аудита, связанных с попытками чтения или модификации файла /etc/passwdqc.conf:

```

# auditctl -w /etc/passwdqc.conf -p rwx

```

Команда поиска записей аудита, после попыток пользователя изменить пароль, включая записи о неуспешных и успешных попытках:

```
# ausearch -f passwd
----
time->Tue Sep 11 14:12:47 2024
type=PROCTITLE msg=audit(1536664367.599:82): proctitle="passwd"
type=PATH msg=audit(1536664367.599:82): item=0
name="/etc/passwdqc.conf" inode=130995 dev=08:02 mode=0100644
ouid=0 ogid=0 rdev=00:00 nametype=NORMAL
type=CWD msg=audit(1536664367.599:82): cwd="/home/test"
type=SYSCALL msg=audit(1536664367.599:82): arch=c000003e
syscall=2 success=yes exit=3 a0=604f17 a1=0 a2=1b6 a3=0 items=1
ppid=2033 pid=2038 auid=501 uid=501 gid=501 euid=501 suid=501
fsuid=501 egid=26 sgid=26 fsgid=26 tty=pts2 ses=2 comm="passwd"
exe="/usr/bin/passwd" key=(null)
----
time->Tue Sep 11 14:12:51 2024
type=PROCTITLE msg=audit(1536664371.201:83): proctitle="passwd"
type=PATH msg=audit(1536664371.201:83): item=0
name="/etc/passwdqc.conf" inode=130995 dev=08:02 mode=0100644
ouid=0 ogid=0 rdev=00:00 nametype=NORMAL
type=CWD msg=audit(1536664371.201:83): cwd="/home/test"
type=SYSCALL msg=audit(1536664371.201:83): arch=c000003e
syscall=2 success=yes exit=4 a0=604f17 a1=0 a2=1b6 a3=0 items=1
ppid=2033 pid=2038 auid=501 uid=501 gid=501 euid=501 suid=501
fsuid=501 egid=26 sgid=26 fsgid=26 tty=pts2 ses=2 comm="passwd"
exe="/usr/bin/passwd" key=(null)
```

В файлах журнала так же будут присутствовать записи:

```
# journalctl -r
сен 11 14:12:10 host-105.localdomain audit: PROCTITLE
proctitle="passwd"
сен 11 14:12:12 host-105.localdomain passwd[2034]:
pam_tcb(passwd:chauthtok): Username obtained: test
сен 11 14:12:12 host-105.localdomain passwd[2034]:
pam_tcb(passwd:chauthtok): New password not obtained
---
сен 11 14:12:51 host-105.localdomain audit[2038]: SYSCALL
arch=c000003e syscall=2 success=yes exit=4 a0=604f17 a1=0 a2=1b6
a3=0 items=1 ppid=2033 pid=2038 auid=501 uid=501 gid=501 euid=501
suid=501 fsuid=501 egid=26 sgid=26 fsgid=26 tty=pts2 ses=2
comm="passwd" exe="/usr/bin/passwd" key=(null)
сен 11 14:12:51 host-105.localdomain audit: CWD cwd="/home/test"
сен 11 14:12:51 host-105.localdomain audit: PATH item=0
name="/etc/passwdqc.conf" inode=130995 dev=08:02 mode=0100644
ouid=0 ogid=0 rdev=00:00 nametype=NORMAL
сен 11 14:12:51 host-105.localdomain audit: PROCTITLE
proctitle="passwd"
сен 11 14:13:02 host-105.localdomain passwd[2038]:
pam_tcb(passwd:chauthtok): Username obtained: test
```

```
сен 11 14:13:02 host-105.localdomain passwd[2038]:
pam_tcb(passwd:chauthtok): Password for test changed by
test(uid=501)
```

### 3.8.3.10. Аудит использования механизма аутентификации

Команда для поиска записей аудита, связанных с использованием механизма аутентификации:

```
# ausearch -m USER_AUTH
----
type=USER_AUTH msg=audit(05.06.2024 10:53:10.996:463) : pid=13905
uid=root auid=unset ses=unset subj=generic_u:generic_r:login_t:s0
msg='op=PAM:authentication grantors=? acct=user
exe=/usr/sbin/sshd hostname=192.168.3.191 addr=192.168.3.191
terminal=ssh res=failed'
----
type=USER_AUTH msg=audit(05.06.2024 10:53:16.718:469) : pid=13920
uid=root auid=unset ses=unset subj=generic_u:generic_r:login_t:s0
msg='op=PAM:authentication grantors=? acct=user
exe=/usr/sbin/sshd hostname=192.168.3.191 addr=192.168.3.191
terminal=ssh res=failed'
----
type=USER_AUTH msg=audit(05.06.2024 10:54:03.485:471) : pid=13873
uid=root auid=unset ses=unset subj=generic_u:generic_r:login_t:s0
msg='op=PAM:authentication grantors=? acct=? exe=/bin/login
hostname=localhost addr=127.0.0.1 terminal=/dev/tty5 res=failed'
----
type=USER_AUTH msg=audit(12.09.2024 08:05:39.955:132) : pid=3370
uid=root auid=unset ses=unset subj=generic_u:generic_r:login_t:s0
msg='op=PAM:authentication
grantors=pam_shells,pam_succeed_if,pam_permit,pam_faillock,pam_moun
t,pam_gnome_keyring acct=user exe=/usr/sbin/lightdm hostname=?
addr=? terminal=:0 res=success'
----
type=USER_AUTH msg=audit(12.09.2024 08:05:54.583:152) : pid=3832
uid=user auid=user ses=2 subj=generic_u:generic_r:generic_t:s0
msg='op=PAM:authentication
grantors=pam_permit,pam_faillock,pam_mount acct=root exe=/bin/su
hostname=localhost addr=127.0.0.1 terminal=/dev/pts/0
res=success'
```

### 3.8.3.11. Аудит использования механизма идентификации

Команда для поиска записей аудита, связанных с использованием механизма идентификации:

```
# ausearch -m USER_LOGIN -i
----
type=USER_LOGIN msg=audit(10.09.2024 16:21:05.455:66) : pid=1294
uid=root auid=root ses=2 msg='op=login id=root exe=/bin/login'
```

```

hostname=host-105.localdomain addr=? terminal=/dev/tty2
res=success'
----
type=USER_LOGIN msg=audit(10.09.2024 16:46:38.366:95) : pid=1270
uid=root auid=test ses=5 msg='op=login id=test
exe=/usr/sbin/lightdm hostname=host-105.localdomain addr=?
terminal=/dev/tty1 res=success'
----
type=USER_LOGIN msg=audit(10.09.2024 16:49:10.308:119) : pid=2652
uid=root auid=unset ses=unset msg='op=login acct=test
exe=/bin/login hostname=host-105.localdomain addr=?
terminal=/dev/tty3 res=failed'
----
type=USER_LOGIN msg=audit(10.09.2024 17:11:01.636:166) : pid=2666
uid=root auid=test ses=10 msg='op=login id=test exe=/bin/login
hostname=host-105.localdomain addr=? terminal=/dev/tty3
res=success'

```

### 3.8.3.12. События, связанные со связыванием атрибутов безопасности пользователя с субъектом

Пример команд создания правила и поиска записей аудита событий, связанных со связыванием атрибутов безопасности пользователя с субъектом:

```

# auditctl -a always,exit -S execve -k exec
# ausearch -i -k exec
----
type=PROCTITLE msg=audit(11.09.2024 16:06:28.583:179) :
proctitle=systemctl start cups
type=PATH msg=audit(11.09.2024 16:06:28.583:179) : item=1
name=/lib64/ld-linux.so.2 inode=261639 dev=08:02 mode=file,755
ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=PATH msg=audit(11.09.2024 16:06:28.583:179) : item=0
name=/sbin/systemctl inode=670657 dev=08:02 mode=file,755
ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=CWD msg=audit(11.09.2024 16:06:28.583:179) : cwd=/root
type=EXECVE msg=audit(11.09.2024 16:06:28.583:179) : argc=3
a0=systemctl a1=start a2=cups
type=SYSCALL msg=audit(11.09.2024 16:06:28.583:179) :
syscall=execve success=yes exit=0 a0=0x6b1870 a1=0x6ca100
a2=0x6bce40 a3=0x59e items=2 ppid=3302 pid=3365 auid=test
uid=root gid=root euid=root suid=root fsuid=root egid=root
sgid=root fsgid=root tty=pts2 ses=2 comm=systemctl
exe=/sbin/systemctl key=exec
----
type=PROCTITLE msg=audit(11.09.2024 16:06:34.034:184) :
proctitle=cons.saver /dev/pts/2
type=PATH msg=audit(11.09.2024 16:06:34.034:184) : item=1
name=/lib64/ld-linux-x86-64.so.2 inode=261639 dev=08:02
mode=file,755 ouid=root ogid=root rdev=00:00 nametype=NORMAL

```



## ЛКНВ.11102-01 99 01

```

type=PATH msg=audit(11.09.2024 16:06:34.034:184) : item=0
name=/usr/lib/mc/cons.saver inode=662639 dev=08:02 mode=file,755
ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=CWD msg=audit(11.09.2024 16:06:34.034:184) : cwd=/home/test
type=EXECVE msg=audit(11.09.2024 16:06:34.034:184) : argc=2
a0=cons.saver a1=/dev/pts/2
type=SYSCALL msg=audit(11.09.2024 16:06:34.034:184) :
syscall=execve success=yes exit=0 a0=0x746a50 a1=0x7fff00fe2f20
a2=0x732dc0 a3=0x84 items=2 ppid=3368 pid=3369 auid=test uid=test
gid=test euid=test suid=test fsuid=test egid=test sgid=test
fsgid=test tty=pts2 ses=2 comm=cons.saver
exe=/usr/lib/mc/cons.saver key=exec
----
type=PROCTITLE msg=audit(11.09.2024 16:06:34.020:183) :
proctitle=mcedit
type=PATH msg=audit(11.09.2024 16:06:34.020:183) : item=1
name=/lib64/ld-linux.so.2 inode=261639 dev=08:02 mode=file,755
ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=PATH msg=audit(11.09.2024 16:06:34.020:183) : item=0
name=/usr/bin/mcedit inode=662634 dev=08:02 mode=file,755
ouid=root ogid=root rdev=00:00 nametype=NORMAL
type=CWD msg=audit(11.09.2024 16:06:34.020:183) : cwd=/home/test
type=EXECVE msg=audit(11.09.2024 16:06:34.020:183) : argc=1
a0=mcedit
type=SYSCALL msg=audit(11.09.2024 16:06:34.020:183) :
syscall=execve success=yes exit=0 a0=0x6c6660 a1=0x6c6740
a2=0x6a3090 a3=0x59e items=2 ppid=2033 pid=3368 auid=test
uid=test gid=test euid=test suid=test fsuid=test egid=test
sgid=test fsgid=test tty=pts2 ses=2 comm=mcedit exe=/usr/bin/mc
key=exec

```

Можно настроить наблюдение за системными вызовами, например, за изменением прав доступа к файлам. Команда создания правила в зависимости от архитектуры:

```
# auditctl -a exit,always -S chmod -S fchmod -S fchmodat -k chmod\*
```

В этом случае, любое изменение прав файла, например, /usr/bin/passwd:

```
# chmod -x /usr/bin/passwd
```

будет приводить к появлению новой записи в журнале аудита:

```
# ausearch -k chmod\*
```

```
----
```

```

time->Tue Jun 27 13:38:29 2017
type=PROCTITLE msg=audit(1498559909.890:218) :
proctitle=63686D6F64002D78002F7573722F62696E2F706173737764
type=PATH msg=audit(1498559909.890:218) : item=0
name="/usr/bin/passwd" inode=528372 dev=08:02 mode=0102711 ouid=0
ogid=26 rdev=00:00 nametype=NORMAL
type=CWD msg=audit(1498559909.890:218) : cwd="/root"

```

```
type=SYSCALL msg=audit(1498559909.890:218): arch=c000003e
syscall=268 success=yes exit=0 a0=fffffffffffffff9c a1=60e720
a2=580 a3=3c4 items=1 ppid=11310 pid=15179 auid=0 uid=0 gid=0
euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=tty2 ses=4
comm="chmod" exe="/bin/chmod" key="chmod*"
```

### 3.8.3.13. Аудит полнотекстовой записи привилегированных команд

Для аудита полнотекстовой записи привилегированных команд (команд, управляющих системными функциями) создать файл `/etc/audit/rules.d/20-privileged.rules` с правилом в зависимости от архитектуры:

```
-a always,exit -F euid=0 -S execve -k privileged
```

Команда поиска записей аудита событий, связанных с полнотекстовой записью привилегированных команд:

```
ausearch -k privileged
```

### 3.8.3.14. События, связанные с загрузкой/выгрузкой модулей

Для аудита событий, связанных с загрузкой/выгрузкой модулей, можно добавить строки правил в файл `/etc/audit/rules.d/20-modules.rules`:

```
-w /sbin/insmod -p x -k modules
-w /sbin/rmmod -p x -k modules
-w /sbin/modprobe -p x -k modules
```

И в зависимости от архитектуры:

```
-a always,exit -S init_module,fininit_module -F key=module-load
-a always,exit -S delete_module -F key=module-unload
```

Команды для поиска записей, связанных с выгрузкой/загрузкой модулей:

```
# ausearch -i -k modules
# ausearch -i -k module-load
# ausearch -i -k module-unload
```

### 3.8.3.15. События, связанных с модификацией режима выполнения ФБО

Команды создания правил, связанных с модификацией выполнения ФБО:

- модификация настроек аудита:

```
# auditctl -w /etc/systemd/journald.conf -p aw -k journald.conf
```

- включение/отключение служб при запуске:

```
# ausearch -i -m SERVICE_STOP,SERVICE_START
```

- модификация настроек выделения случайной памяти:

```
# auditctl -w /proc/sys/kernel/randomize_va_space -p wa -k randomize
```

- модификация настроек аутентификации и идентификации:

```
# auditctl -w /etc/pam.d/ -p wa -k auth
```

Команда поиска записей аудита:

```
# ausearch -k ключевое_слово
```

### 3.8.3.16. События, связанных с модификацией значений атрибутов безопасности

Команды создания правила и поиска записей аудита, связанных с модификацией значений атрибутов безопасности (каталоги должны существовать):

```
# auditctl -w /path/to/dir -p a -k FMT_MSA
# ausearch -k FMT_MSA
```

Команды создания правила и поиска записей аудита, связанных с модификацией настроек по умолчанию ограничительных правил и модификацией начальных значений атрибутов:

```
# auditctl -w /etc/profile -p w -k etcprofile
# ausearch -k etcprofile
```

Команды создания правила и поиска записей аудита, связанных с модификациями значений данных ФБО:

```
# auditctl -w /etc/tcb -p w -k tcb
# ausearch -k tcb
```

Поиск записей аудита, связанных с модификацией настроек межсетевого экрана:

```
# ausearch -i -m NETFILTER_CFG
----
type=PROCTITLE msg=audit(11.09.2024 16:50:28.535:247) :
proctitle=iptables -I INPUT -p udp -j ACCEPT
type=SYSCALL msg=audit(11.09.2024 16:50:28.535:247) :
syscall=setsockopt success=yes exit=0 a0=0x4 a1=ip
a2=IPT_SO_SET_REPLACE a3=0x67e0e0 items=0 ppid=1994 pid=3687
auid=test uid=root gid=root euid=root suid=root fsuid=root
egid=root sgid=root fsgid=root tty=pts1 ses=2 comm=iptables
exe=/sbin/xtables-multi key=(null)
type=NETFILTER_CFG msg=audit(11.09.2024 16:50:28.535:247) :
table=filter family=ipv4 entries=4
```

Для записи всех команд, введенных в консоль (tty), в файл /etc/pam.d/system-auth необходимо добавить строку:

```
session required pam_tty_audit.so enable=*
```

Команда для поиска записей:

```
# ausearch -i -m TTY
----
type=TTY msg=audit(11.09.2024 17:05:27.430:504) : tty pid=4172
uid=root auid=root ses=13 major=4 minor=3 comm=bash
data="ls",<ret>,<up>,<up>,<ret>
----
type=TTY msg=audit(11.09.2024 17:06:28.381:588) : tty pid=5271
uid=root auid=root ses=17 major=4 minor=4 comm=bash data="mkdir
tt",<backspace>,"est",<ret>,<up>,<up>,<ret>
```

Или для вывода в таблицу:

```
# aureport --tty
TTY Report
=====
# date time event auid term sess comm data
=====
1. 11.09.2024 17:05:27 504 0 ? 13 bash "ls",<ret>,<up>,<up>,<ret>
2. 11.09.2024 17:06:28 588 0 ? 17 bash "mkdir
tt",<backspace>,"est",<ret>,<up>,<up>,<ret>
```

### 3.8.3.17. События, связанные с ограничением числа параллельных сеансов

Отклонение нового сеанса, основанное на ограничении числа параллельных сеансов, настраивается в /etc/security/limits.conf (см. документ «Руководство администратора. ЛКНВ.11102-01 90 01»).

Команда для поиска записей, связанных с отклонением нового сеанса:

```
# ausearch -m ANOM_LOGIN_SESSIONS
----
time->Wed Sep 12 15:21:19 2024
type=ANOM_LOGIN_SESSIONS msg=audit(1536754879.189:55295) :
pid=4019 uid=0 auid=4294967295 ses=4294967295
msg='op=PAM:pam_limits acct="test2" exe="/bin/login"
hostname=localhost addr=127.0.0.1 terminal=/dev/tty5 res=failed'
```

### 3.8.3.18. События, связанные с изменениями даты и времени

Для регистрации изменений даты и времени, необходимо включить контроль над изменением значения времени.

Команда для регистрации событий, изменения времени через `clock_settime`, `settimeofday` и `adjtimex` в зависимости от архитектуры:

```
# auditctl -a exit,always -S clock_settime -S settimeofday -S adjtimex
-k FPT_STM
```

Команда поиска записей аудита:

```
# ausearch -k FPT_STM

time->Thu May 24 18:08:33 2024
type=PROCTITLE msg=audit(1527174513.884:58335):
proctitle=64617465002D2D7365743D323031382D30392D31322031343A33373
A3236
type=SYSCALL msg=audit(1527174513.884:58335): arch=c000003e
syscall=227 success=yes exit=0 a0=0 a1=7fff03d63f00 a2=0 a3=581
items=0 ppid=4620 pid=6795 auid=4294967295 uid=0 gid=0 euid=0
suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts6 ses=4294967295
comm="date" exe="/bin/date" key="FPT_STM"
----
time->Wed Sep 12 18:09:23 2024
type=PROCTITLE msg=audit(1536764963.097:58228):
proctitle=6461746500303532343138303832303138
type=SYSCALL msg=audit(1536764963.097:58228): arch=c000003e
syscall=227 success=yes exit=0 a0=0 a1=7ffe0c058ba0
a2=7ffe0c058ac0 a3=581 items=0 ppid=1946 pid=6061 auid=501 uid=0
gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0 ses=2
comm="date" exe="/bin/date" key="FPT_STM"
```

Команда удаления правила контроля над изменением значения времени:

```
# auditctl -d exit,always -S clock_settime -S settimeofday -S adjtimex
```

**3.8.3.19. События, связанные с попытками разблокирования интерактивного сеанса**

В журнал попадают все удачные и неудачные (если была попытка ввода пароля) попытки разблокирования интерактивного сеанса.

Команда для поиска событий, связанных с попытками разблокирования сеанса в графическом интерфейсе:

```
# journalctl -b | grep mate-screensaver
сен 13 10:53:38 host-105.localdomain mate-screensave[3275]:
UNSPECIFIED (__progname="mate-screensaver-chkpwd-helper" uid=501
gid=501 egid=24): pam_tcb(mate-screensaver:auth): Authentication
failed for teacher from teacher(uid=501)
сен 13 10:53:41 host-105.localdomain mate-screensave[3275]:
UNSPECIFIED (__progname="mate-screensaver-chkpwd-helper" uid=501
gid=501 egid=24) [3275]: pam_authenticate(mate-screensaver,
teacher): Authentication failure
```

```
сен 13 10:59:02 host-105.localdomain mate-screensave[3314]:
UNSPECIFIED (__progname="mate-screensaver-chkpwd-helper" uid=501
gid=501 egid=24): pam_tcb(mate-screensaver:auth): Authentication
passed for teacher from teacher(uid=501)
```

### Команда для поиска событий, связанных с попытками разблокирования сеанса

#### В КОНСОЛИ:

```
# journalctl -b | grep vlock
сен 13 11:02:12 host-105.localdomain vlock[3351]: Locked VC on
tty2 for test2 by (uid=503)
сен 13 11:02:14 host-105.localdomain vlock[3351]:
pam_tcb(vlock:auth): Authentication failed for test2 from
test2(uid=503)
сен 13 11:02:19 host-105.localdomain vlock[3351]:
pam_tcb(vlock:auth): Authentication passed for test2 from
test2(uid=503)
сен 13 11:02:19 host-105.localdomain vlock[3351]: Unlocked VC on
tty2 for test2 by (uid=503)
# ausearch -x vlock -m USER_AUTH
----
time->Wed Sep 12 15:04:38 2024
type=USER_AUTH msg=audit(1536753878.151:63250): pid=7265 uid=0
aid=0 ses=13 msg='op=PAM:authentication grantors=? acct="root"
exe="/usr/bin/vlock" hostname=host-105.localdomain addr=?
terminal=tty2 res=failed'
----
time->Wed Sep 12 15:04:43 2024
type=USER_AUTH msg=audit(1536753883.223:63266): pid=7265 uid=0
aid=0 ses=13 msg='op=PAM:authentication grantors=pam_tcb
acct="root" exe="/usr/bin/vlock" hostname=host-105.localdomain
addr=? terminal=tty2 res=success'
```

### Команды создания и поиска событий, связанных с блокированием

#### интерактивного сеанса механизмом блокирования сеанса (в консоли):

```
# auditctl -w /usr/bin/vlock -p x -k block
# ausearch -k block
----
time->Wed Sep 12 15:56:10 2024
type=CONFIG_CHANGE msg=audit(1536756970.362:102): auid=501 ses=4
op="add_rule" key="block" list=4 res=1
----
time->Wed Sep 12 15:56:16 2024
type=PROCTITLE msg=audit(1536756976.527:106): proctitle="vlock"
type=PATH msg=audit(1536756976.527:106): item=1 name="/lib64/ld-linux-
x86-64.so.2" inode=261639 dev=08:02 mode=0100755 ouid=0 ogid=0
rdev=00:00 nametype=NORMAL
```

## ЛКНВ.11102-01 99 01

```

type=PATH msg=audit(1536756976.527:106): item=0 name="/usr/bin/vlock"
inode=662838 dev=08:02 mode=0102711 ouid=0 ogid=24 rdev=00:00
nametype=NORMAL
type=CWD msg=audit(1536756976.527:106): cwd="/root"
type=EXECVE msg=audit(1536756976.527:106): argc=1 a0="vlock"
type=SYSCALL msg=audit(1536756976.527:106): arch=c000003e syscall=59
success=yes exit=0 a0=6d4e90 a1=6d4ed0 a2=6bf340 a3=59e items=2
ppid=1504 pid=2290 auid=0 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=24
sgid=24 fsgid=24 tty=tty2 ses=2 comm="vlock" exe="/usr/bin/vlock"
key="block"

```

**Команды создания и поиска событий, связанных с блокированием интерактивного сеанса механизмом блокирования сеанса в графическом интерфейсе:**

```

# auditctl -w /usr/bin/mate-screensaver-command -p x -k block_mate
# ausearch -k block_mate
----
time->Wed Sep 12 16:50:20 2024
type=PROCTITLE msg=audit(1536760220.072:172):
proctitle=6D6174652D73637265656E73617665722D636F6D6D616E64002D2D6C6F636
B
type=PATH msg=audit(1536760220.072:172): item=1 name="/lib64/ld-linux-
x86-64.so.2" inode=261639 dev=08:02 mode=0100755 ouid=0 ogid=0
rdev=00:00 nametype=NORMAL
type=PATH msg=audit(1536760220.072:172): item=0 name="/usr/bin/mate-
screensaver-command" inode=699133 dev=08:02 mode=0100755 ouid=0 ogid=0
rdev=00:00 nametype=NORMAL
type=CWD msg=audit(1536760220.072:172): cwd="/home/user"
type=EXECVE msg=audit(1536760220.072:172): argc=2 a0="mate-screensaver-
command" a1="--lock"
type=SYSCALL msg=audit(1536760220.072:172): arch=c000003e syscall=59
success=yes exit=0 a0=8cd931 a1=8cd980 a2=904470 a3=59e items=2 ppid=1
pid=5259 auid=501 uid=501 gid=501 euid=501 suid=501 fsuid=501 egid=501
sgid=501 fsgid=501 tty=(none) ses=4 comm="mate-screensave"
exe="/usr/bin/mate-screensaver-command" key="block_mate"

```

### 3.8.3.20. События, связанные с выполнением сбоя и прерывания обслуживания

Например, инициируем сбой:

```
# killall -SEGV cupsd
```

**Команда для поиска записей аудита, связанных с выполнением сбоя и прерывания обслуживания:**

```

# ausearch -i -m ANOM_ABEND,SERVICE_STOP
----
type=SERVICE_STOP msg=audit(11.09.2024 17:28:19.673:749) : pid=1
uid=root auid=unset ses=unset msg='unit=org.cups.cupsd

```

```
comm=systemd exe=/lib/systemd/systemd hostname=? addr=?
terminal=? res=failed'
----
type=ANOM_ABEND msg=audit(11.09.2024 17:28:19.661:748) :
auid=unset uid=root gid=root ses=unset pid=3349 comm=cupsd
exe=/usr/sbin/cupsd sig=SIGSEGV
```

### 3.8.3.21. События, связанные с использованием функций распределения ресурсов с учетом приоритетности обслуживания

Создание правила и поиск записей аудита, связанных с использованием функций распределения ресурсов с учетом приоритетности обслуживания:

```
# auditctl -a always,exit -S setpriority -k nice
# nice -n 19 factor 1223412456656757
# ausearch -i -k nice
```

### 3.8.3.22. События, связанные с обращением к функциям распределения ресурсов, управляемых ФБО

Команды создания правил и поиск записей аудита, связанных с обращением к функциям распределения ресурсов, управляемых ФБО, в том числе в зависимости от архитектуры:

```
# auditctl -a always,exit -S clone -k fork
# ausearch -i -k fork

# auditctl -a always,exit -S mmap2,brk -k memory
# ausearch -i -k memory
```

### 3.8.3.23. События, связанные с изменением информации о пользователях/группах

Для фиксации событий, которые вносят изменения в пользовательские аккаунты добавьте следующие строки в файл `/etc/audit/audit.rules`:

```
# audit_account_changes
-w /etc/group -p wa -k audit_account_changes
-w /etc/passwd -p wa -k audit_account_changes
-w /etc/gshadow -p wa -k audit_account_changes
-w /etc/shadow -p wa -k audit_account_changes
-w /etc/security/opasswd -p wa -k audit_account_changes
```



### 3.8.3.24. Отчет обо всех попытках входа в систему, изменения учетных записей

Команда `aureport` позволяет вывести отчет о всех попытках входа в систему:

```
# aureport -l
```

```
Login Report
```

```
# date time auid host term exe success event
```

```
=====
1. 16.05.2024 13:10:22 -1 host-15.localdomain /dev/tty1
   /usr/sbin/lightdm yes 80
2. 16.05.2024 20:28:15 root host-15.localdomain /dev/tty2
   /bin/login no 214
3. 16.05.2024 20:28:19 -1 host-15.localdomain /dev/tty2
   /bin/login yes 224
4. 16.05.2024 20:28:29 newuser host-15.localdomain /dev/tty4
   /bin/login no 227
5. 01.06.2024 13:16:30 -1 host-15.localdomain /dev/tty1
   /usr/sbin/lightdm yes 79
6. 01.06.2024 13:17:08 user 192.168.3.191 sshd /usr/sbin/sshd no
   85
7. 01.06.2024 13:17:10 -1 192.168.3.191 /dev/pts/1 /usr/sbin/sshd
   yes 94
8. 01.06.2024 13:17:51 root 192.168.3.191 sshd /usr/sbin/sshd no
   95
11. 01.06.2024 13:19:00 user host-15.localdomain /dev/tty3
   /bin/login no 102
12. 01.06.2024 13:19:07 -1 host-15.localdomain /dev/tty3
   /bin/login yes 112
```

Команда для вывода отчета о неудачных попытках входа в систему:

```
# aureport -l --failed
```

```
Login Report
```

```
# date time auid host term exe success event
```

```
=====
1. 16.05.2024 20:28:15 root host-15.localdomain /dev/tty2
   /bin/login no 214
2. 16.05.2024 20:28:29 newuser host-15.localdomain /dev/tty4
   /bin/login no 227
3. 01.06.2024 13:17:08 user 192.168.3.191 sshd /usr/sbin/sshd no
   85
4. 01.06.2024 13:17:51 root 192.168.3.191 sshd /usr/sbin/sshd no
   95
5. 01.06.2024 13:19:00 user host-15.localdomain /dev/tty3
   /bin/login no 102
```

Команда для вывода отчета об изменениях пользовательских учетных записей:

```
# aureport -m
Account Modifications Report
=====
# date time auid addr term exe acct success event
=====
1. 24.05.2024 18:42:41 500 host-15.localdomain pts/0
   /usr/bin/passwd testuser yes 101
```

### 3.9. Средства контроля запуска компонентов программного обеспечения

В ОС Альт СП механизм контроля запуска компонентов программного обеспечения реализуется при помощи программы control++.

#### 3.9.1. Принцип работы control++

При установке режима ограничений и прав формируются данные, обеспечивающие возможность возврата к исходному состоянию при сбросе текущего режима. Также при каждой установке режима сохраняется описание установленного режима.

При каждой установке режима происходит предварительный сброс текущего режима.

При запуске программы для переключения режима, например, control++ <название\_режима>, которому соответствует вариант ограничений ulims\_x, вариант прав perm\_y и сценарий sh\_z, программа попытается скопировать файл /etc/control++/ulimits/ulims\_x в каталог /etc/security/limits.d/ (таким образом, чтобы он имел наивысший приоритет среди уже имеющихся файлов ограничений), попытается применить права на файлы в соответствии с описанием в /etc/control++/permissions/perm\_y и запустить сценарий /etc/control++/permissions/sh\_z/do. В случае невозможности осуществления какой-либо из операций будет выведено сообщение об ошибке. Если какой-то из параметров не указан (например, название сценария оболочки), то соответствующая операция не будет выполняться.

При проверке соответствия текущего состояния системы параметрам установленного ранее режима осуществляется:

- сравнение содержимого наиболее приоритетного ulimits-файла из каталога `/etc/security/limits.d/` с содержимым ulimits-файла, сохраненного при установке данного режима;
- сравнение текущих режимов всех файлов, затронутых установкой данного режима, с текущим описанием прав, которое носит название установленного подрежима;
- запуск test-сценария данного режима, при наличии такого сценария (его отсутствие не считается ошибкой).

Если какой-то из файлов, перечисленных в описании прав для данного режима, отсутствует в системе, то установка данного режима не будет считаться по этой причине успешной. Также, при проверке соответствия режима, права отсутствующего файла не считаются несоответствующими правам, указанным в описании режима.

**Примечание.** После установки режима черного или белого списка до перезагрузки системы восстановление целостности системы (если система контроля целостности `ima-evm` не инициализирована) осуществляется командой:

```
# integalert fix
```

Если система контроля целостности `ima-evm` инициализирована, восстановление целостности системы выполняется с помощью команды:

```
# integrity-applier
```

### 3.9.2. Настройка

#### 3.9.2.1. Основной режим

Параметры `control++` определяются файлом `ini`-формата `/etc/control++/control++.conf`. Данный файл состоит из секций описания каждого из режимов. Название секции соответствует названию режима. Каждая из секций может состоять из определения варианта ограничений (`ulimits`), варианта набора прав (`permissions`) и запускаемого сценария оболочки для данного режима (`scripts`).

Далее приведен пример описания режима под названием `workstation`, который имеет тип `ulimits` под названием `u_x`, тип `permissions` под названием `p_y` и запускаемый сценарий оболочки `s_z/do`:

```
[workstation]
ulimits = u_x
permissions = p_y
scripts = s_z
```

Все файлы настроек могут быть отредактированы вручную системным администратором при настройке нужных конфигураций.

### 3.9.2.2. Режим ULimits

Файлы `ulimits` для каждого варианта ограничений находятся в файлах `/etc/control++/ulimits/<название_варианта>`.

### 3.9.2.3. Режим прав на файлы

Файлы с описанием набора прав являются файлами `ini`-формата и находятся в файлах `/etc/control++/permissions/<название_варианта>`. Описание прав может состоять из следующих секций:

- `file` – секция, задающая права на файл, абсолютный путь которого определяется значением `path` данной секции;
- `dir` – секция, задающая права на файлы каталога и всех содержащихся в нем файлов (без учета содержимого подкаталогов), абсолютный путь которого определяется значением `path` данной секции;
- `dir_r` – вариация секции `dir`, задающая права не только на файлы каталога, но и на все содержащиеся в нем файлы с учетом содержимого подкаталогов (рекурсивный обход дерева подкаталогов);
- `list` – секция, задающая права на файлы, список абсолютных путей которых задан в текстовом файле, абсолютный путь которого определяется значением `path` данной секции;
- `list_r` – вариация секции `list`, задающая права не только на файлы из списка, но и на все файлы перечисленных в нем каталогов с учетом содержимого подкаталогов;

- `whitelist` – вариация секции `list_r`, для которой устанавливаемый режим файлов определен как `*****` (\* означает не менять данный бит режима), при этом для всех остальных файлов базового каталога устанавливается режим `**_**_**_`;
- `blacklist` – вариация секции `list_r`, для которой устанавливаемый режим файлов определен как `**_**_**_`.

Помимо значения `path` для всех секций могут быть определены значения следующих параметров:

- `owner` – название учетной записи владельца файла;
- `group` – название группы, к которой относится файл;
- `mode` – режим файла в формате `gwxgwxgwx` (например, `rw-rw-rw-` означает разрешить всем чтение и запись, но запретить всем запуск данного файла; `rwx-----` означает разрешить все действия владельцу файла и запретить все действия всем остальным. Для того чтобы не изменять какой-то бит режима, следует использовать символ \*, например, `r**r**r**` означает разрешить всем чтение файла и не менять остальные права).

Единственным обязательным параметром является `path`. При отсутствии определения остальных параметров данные свойства файла не будут изменены.

Для секций `dir` и `list` (а также всех их вариаций) может быть определен дополнительный параметр `excluded_paths`, определяющий набор каталогов, содержимое которых (в том числе содержимое вложенных каталогов) не будет затронуто применением прав, описанных данной секцией.

Для секций `dir`, `dir_r` и `list_r`, `whitelist`, `blacklist` может быть определен дополнительный параметр `mode_for_dirs`, определяющий режим для каталогов, затрагиваемых применением прав, описанных данной секцией (по умолчанию режим для каталогов определяется так же, как для обычных файлов).

Для секций `list`, `list_r`, `whitelist` и `blacklist` может быть определен дополнительный параметр `base_dir`, определяющий каталог, рассматриваемый при применении данного режима прав как каталог верхнего уровня.

Например, в секции `whitelist` параметру `base_dir` присвоено значение `/home/your_home_dir/`, в списке файлов, к которым должен быть применен данный режим, указаны два пути – `/d1/f1`, `f2` и `excluded_paths` присвоено значение `/d0`, тогда при установке данного режима файлы `/home/your_home_dir/d1/f1` и `/home/your_home_dir/f2` станут исполняемыми, файлы каталога `/home/your_home_dir/d0` (включая содержимое вложенных каталогов) сохранят свой режим, а все остальные файлы каталога `/home/your_home_dir/` станут неисполняемыми. Значение параметра `path` секции `list` и производных от нее секций должно представлять собой абсолютный путь к текстовому файлу, каждая строка которого представляет собой абсолютный путь какого-либо файла системы, при этом допустимы комментарии, обозначаемые комбинацией символов `//`.

Пример описания набора прав:

```
[file]
path = ~/some_dir_1/some_file_1
owner = some_user
group = some_group
mode = rwxrwx---
```

```
[file]
path = ~/some_dir_1/some_file_2
mode = rw-rw----
```

```
[dir_r]
path = ~/some_dir_2/
owner = some_user
group = some_group
mode = **x**x**x
mode_for_dirs = r*xr*xr*x
```

```
[list]
path = ~/list_of_executables.txt
excluded_paths = "/some_path/", "/some_other_path/"
owner = some_user
group = some_group
mode = **x**x**-
```

```
[blacklist]
path = ~/some_blacklist.txt
```

```
[whitelist]
path = ~/some_whitelist.txt
base_dir = /mnt/some_vol/
```

```
excluded_paths = /some_path_inside_base_dir/
```

**Примечание.** При установке прав для секции `whitelist` следует помнить, что при наличии `/etc/control++/scripts/<название_режима>/do` – запускаемого сценария у режима, данный файл сценария следует учесть в списке разрешенных для запуска файлов или изменить порядок действий в главном файле настроек (переместить определение переменной `scripts` выше определения переменной `permissions`). В противном случае возможна ситуация, при которой в результате установки режима прав данный файл станет неисполняемым, что приведет к невозможности завершения установки данного режима.

#### 3.9.2.4. Управление режимами

Управление режимами, осуществление записи в каталог `/etc/security/limits.d/`, работа с `control++` выполняется от администратора.

Установка режима:

```
control++ <название_режима>
```

Сброс текущего режима:

```
control++ --reset
```

Отображение списка доступных режимов:

```
control++ --list
```

Проверка соответствия состояния системы текущему режиму:

```
control++ --status
```

Отображение содержимого главного файла настройки:

```
control++ --conf
```

Отображение справочной информации:

```
control++ --help
```

#### 3.9.2.5. Настройка режима контроля запуска компонентов программного обеспечения

Текстовый файл `/etc/control++/black_list` – черный список, каждая строка которого представляет собой абсолютный путь к файлу данной системы. По единой команде производится запрещение запуска (исполнения) всех файлов черного списка (по умолчанию – для всех пользователей).

Текстовый файл `/etc/control++/white_list` – белый список, каждая строка которого представляет собой абсолютный путь к файлу данной системы.

По единой команде производится запрещение запуска (исполнения) всех файлов системы, кроме файлов белого списка, а права на запуск (исполнение) файлов белого списка остаются неизменными (по умолчанию, может быть изменено настройками).

Настройка режима контроля запуска компонентов программного обеспечения выполняется от администратора.

Для настройки черного и (или) белого списка отредактировать содержимое файла `/etc/control++/control++.conf`, включив описание соответствующих режимов (`wl` и `blacklist`), каждый из которых имеет тип `permissions` (под названием `wl` и `bl`):

```
[wl]
permissions = wl
[blacklist]
permissions = bl
```

**Пример создания конфигурационного файла белого списка**

`/etc/control++/permissions/wl:`

```
[whitelist]
path = "/etc/control++/white_list"
base_dir = "/"
excluded_paths = "boot", "dev", "etc", "lost+found", "media",
"mnt", "proc", "run", "srv", "sys", "tmp", "var"
mode_for_dirs = *****
```

**Пример создания конфигурационного файла черного списка**

`/etc/control++/permissions/bl:`

```
[blacklist]
path = "/etc/control++/black_list"
```

Запустить генерацию белого списка:

```
bash /etc/control++/wl.sh
```

**Примечание.** Перед использованием скрипта генерации белого списка `/etc/control++/wl.sh` необходимо установить пакет `control++-wl`, находящийся на компакт-диске с ОС.

В файл `/etc/control++/white_list` будут записаны все исполняемые файлы. Можно удалить из файла `/etc/control++/white_list` какую-либо программу, например, `/usr/bin/dig` (удалив соответствующую строку из файла `/etc/control++/white_list`).



Сформировать черный список. Например, занести в черный список программу Mozilla Firefox:

```
# vim /etc/control++/black_list
/usr/bin/firefox
```

Просмотреть список доступных режимов:

```
# control++ --list
Available modes:
wl
blacklist
```

Установка режима применения черного списка (blacklist):

```
# control++ blacklist
Setting 'blacklist' mode for the 'permissions'
unit.....[DONE]
Mode set OK
Current mode is 'blacklist'
```

После установки данного режима, пользователю будет запрещен запуск программ, перечисленных в файле /etc/control++/black\_list (в примере Mozilla Firefox).

Для установки режима применения белого списка (wl), необходимо выполнить команду:

```
control++ wl
Setting 'wl' mode for the 'permissions'
unit.....[DONE]
Mode set OK
Current mode is 'wl'
```

**Примечание.** Выполнение команды может занять довольно продолжительное время (время зависит от количества установленных в системе файлов).

После установки данного режима, пользователю будет разрешен запуск только тех программ, которые перечислены в файле /etc/control++/white\_list, в примере все исполняемые файлы системы, кроме /usr/bin/dig:

```
$ dig
bash: /usr/bin/dig: Отказано в доступе
$ pwd
/home/user
```

Для сброса существующего режима следует выполнить команду:

```
control++ --reset
Restoring initial state:
Restoring initial state for the 'permissions'
unit.....[DONE]
Mode reset OK
```

### 3.10. Надежное хранение данных

Администратор должен определить, на какой период возможна недоступность данных, и, в соответствии с этим, предоставить рекомендации по использованию средств повышения доступности, таких как RAID-массивы. В ОС Альт СП есть поддержка RAID-массивов.

## 4. УКАЗАНИЯ ПО ЭКСПЛУАТАЦИИ КСЗ

### 4.1. Порядок старта

КСЗ является неотъемлемой частью ядра ОС Альт СП и системных библиотек. Старт КСЗ выполняется автоматически после запуска ПЭВМ и отработки набора программ BIOS.

КСЗ запускается вместе с ОС, проходя непосредственно через этапы досистемной загрузки, тестирования ядром окружения процессов и запуска ФС, а также инициализации первичного процесса `init`.

В связи с этим порядок старта служб КСЗ напрямую связан со стартом самой ОС и ее ядра и происходит следующим образом:

- 1) ядро запускается программой начальной загрузки, распаковывает себя и инициализирует устройство отображения, запускает проверку другого оборудования, присоединенного к компьютеру;
- 2) ядро загружает модули обнаруженных устройств;
- 3) ядро запускает процессы ядра;
- 4) ядро монтирует корневую файловую систему только для чтения и выполняет проверку файловой системы;
- 5) ядро монтирует корневую файловую систему;
- 6) ядро запускает из файла `/sbin/init` процесс `init`, и это считается началом загрузки непосредственно ОС, во время которого (`sysinit`) выполняются следующие действия:
  - устанавливается имя машины (`hostname`);
  - конфигурируются параметры ядра;
  - устанавливается раскладка клавиш и системный шрифт;
  - активируются разделы подкачки;
  - корневая система проверяется программой `fsck`, и если программа `fsck` ошибок не обнаружила, файловая система монтируется в режиме чтение/запись;

- проверяются зависимости модулей ядра;
- выполняется проверка других файловых систем;
- монтируются локальные файловые системы;
- включаются квоты;
- монтируется раздел подкачки;
- КСЗ развертывается вместе с началом загрузки ОС.

Правильность старта КСЗ рекомендуется проверять непосредственно после каждого развертывания.

#### 4.2. Проверка правильности старта КСЗ

Рекомендуется убедиться при помощи утилиты `systemctl` в том, что системные службы запущены и работают, их список и состояние не отличаются от обычного для данной системы (набор служб может отличаться в зависимости от параметров установки и настройки).

#### 4.3. Периодическая проверка целостности КСЗ

##### 4.3.1. Порядок проверки

Регулярно (не реже чем раз в две недели) администратору рекомендуется выполнять контроль состава установленного программного обеспечения на предмет его соответствия политике безопасности предприятия.

Ежедневно администратором должно проверяться наличие вредоносного программного обеспечения.

Периодически (не реже чем раз в месяц) должна происходить смена паролей (кодов) пользователей/администраторов.

Периодически (не реже чем раз в неделю) администратором должна проверяться целостность программной и информационной частей ОС Альт СП.

Периодически (не реже чем раз в месяц) администратор должен проводить тестирование функций защиты информации системы, в которой эксплуатируется ОС Альт СП.

В случае обнаружения уязвимостей в программных модулях ОС Альт СП устранение уязвимости осуществляется путем установки сертифицированного обновления, либо путем принятия иных организационно-технических мер, направленных на затруднение возможности эксплуатации уязвимости. При этом сами меры носят временный характер, а их использование допустимо до момента выпуска соответствующего обновления.

Контроль целостности неизменяемых файлов и обновлений ОС можно осуществить с помощью скрипта `verify-checksum` см. п. 3.6.4.1.

#### 4.3.2. Контроль целостности файлов паролей и списка групп

Файлы учетных записей и групп (`/etc/passwd` и `/etc/group`) интенсивно используются в процессе администрирования, и ОС Альт СП предоставляет программные средства для проверки правильности их синтаксиса.

##### 4.3.2.1. Проверка целостности файлов паролей

Файл `/etc/passwd` проверяется командой `pwck`. Данная команда последовательно анализирует записи и проверяет, что каждая запись содержит:

- правильное количество полей;
- уникальное имя пользователя;
- действительные идентификаторы пользователей и групп;
- действительную первичную группу;
- действительный домашний каталог;
- действительный командный процессор.

Для проведения теста необходимо авторизоваться в системе от имени администратора (`root`), открыть файл `/etc/passwd` в текстовом редакторе `mcedit` и выполнить следующие изменения:

- изменить строку `adm:x:3:4:adm:/var/adm:/dev/null`  
на `adm:x: adm:/var/adm:/dev/null`
- изменить строку `bin:x:1:1:bin:/dev/null`  
на `adm:x:1:1:bin:/dev/null`

## ЛКНВ.11102-01 99 01

- изменить строку `daemon:x:2:2:daemon:/:/dev/null`  
на `daemon:x:22222:22222:daemon:/:/dev/null`
- изменить строку `mail:x:8:12:mail:/var/spool/mail:/dev/null`  
на `mail:x:8:12:mail_test:/var/spool/mail:/dev/null`
- изменить строку `ftp:x:14:50:FTP User:/var/ftp:/dev/null`  
на `ftp:x:14:50:FTP User:/var/ftp_test:/dev/null`
- изменить строку `news:x:9:13:news:/var/spool/news:/dev/null`  
на `news:x:9:13:news:/var/spool/news:/bin/bash_test`

Сохранить внесенные изменения и выйти из редактора.

Запустить программу `pwck` в режиме «только чтение» для просмотра всех ошибок:

```
pwck -r
```

Ожидаемые результаты: результат работы программы `pwck` должен быть выведен на консоль. ОС Альт СП должна обнаружить все синтаксические ошибки и вывести их на консоль:

```
pwck: shadow files will not be checked (tcb)
пользователь daemon: группа 22222 не существует
неверная запись в файле паролей
удалить строку 'adm:x: adm:/var/adm:/dev/null'? Нет
пользователь 'lp': каталог '/var/spool/lpd' не существует
пользователь 'news': каталог '/var/spool/news' не существует
пользователь 'news': каталог '/bin/bash_test' не существует
пользователь 'uucpr': каталог '/var/spool/uucpr' не существует
пользователь 'ftp': каталог '/var/ftp_test' не существует
пользователь 'named': каталог '/var/lib/named' не существует
пользователь 'mailman': каталог '/usr/share/mailman' не
существует
пользователь 'xfs': каталог '/etc/X11/fs' не существует
пользователь 'postgres': каталог '/var/lib/pgsql' не существует
пользователь 'gdm': каталог '/var/lib/gdm' не существует
пользователь 'exim': каталог '/var/spool/exim' не существует
пользователь 'colord': каталог '/var/colord' не существует
pwck: изменений не внесено
```

**Примечание.** В ОС Альт СП для хранения хэшированных паролей используется `tcb` и поэтому файл `/etc/shadow` не используется, а файлы `/etc/tcb/*/shadow` программой `pwck` не проверяются. Кроме того, создается ряд

предопределенных учетных записей, домашние каталоги для которых появляются только после установки собственно пакетов со службами.

#### 4.3.2.2. Проверка целостности списка групп

Командой `grpck` выполняется проверка файлов `/etc/group` и `/etc/gshadow`.

Данная команда последовательно анализирует записи и проверяет, что каждая запись содержит:

- правильное количество полей;
- уникальное имя группы;
- действительный список членов и администраторов.

Для проведения теста зайти в систему с учетной записью администратора (`root`), открыть файл `/etc/group` в текстовом редакторе `mcedit` и выполнить следующие изменения:

- изменить строку `daemon:x:2:root`

на `daemon:root`

- изменить строку `adm:x:4:root`

на `root:x:4:root`

Сохранить внесенные изменения и выйти из редактора.

Запустить программу `grpck` в режиме «только чтение» для просмотра всех ошибок:

```
grpck -r
```

Ожидаемые результаты: результат работы программы `grpck` должен быть выведен на консоль. ОС Альт СП должна обнаружить все синтаксические ошибки и вывести их на консоль:

```
повторяющаяся запись в файле групп
удалить строку 'root:x:0: '? Нет
повторяющаяся запись в файле групп
удалить строку ' daemon:root'? Нет
'root' член группы 'root' в /etc/group но не в /etc/gshadow
неверная запись в файле групп
удалить строку 'daemon:root'? Нет
отсутствует соответствующая группа в файле /etc/group
удалить строку 'daemon:x::root'? Нет
```

отсутствует соответствующая группа в файле /etc/group  
удалить строку 'adm:x::root'? Нет  
grpck: изменений не внесено



## 5. ОБНОВЛЕНИЕ ОС АЛЬТ СП

Системы, имеющие выход в сеть Интернет, получают обновления из публичного репозитория в соответствии с выбранной веткой для нужного дистрибутива.

**Примечание.** Актуальная ссылка на репозиторий указана в формуляре. Текущая версия дистрибутива указана в директории `/etc/apt/sources.list.d/` в конфигурационном файле описания источников с расширением `.list`.

Для систем, не имеющих выхода в сеть Интернет, рекомендуется установить отдельный сервер обновлений на базе ОС Альт СП, находящийся вне защищенного контура, и организовать ограниченный доступ к этому серверу. Настройка сервера обновлений осуществляется при помощи модуля «Сервер обновлений» (пакет `alterator-mirror`) ЦУС (см. документ «Руководство администратора. ЛКНВ.11102-01 90 01»).

Информирование потребителей о мерах, направленных на нейтрализацию выявленных уязвимостей ОС Альт СП, и выпускаемых обновлениях, влияющих на безопасность, ОС Альт СП, выполняется путем публикации информации на официальном сайте изделия (<https://altsp.su>) или по электронной почте.

Данные обновления доводятся до потребителей через публичный репозиторий до проведения соответствующих испытаний изделия по требованиям безопасности информации.

После проведения испытаний предприятие-изготовитель может предоставить потребителям обновления или обновленный дистрибутив ОС Альт СП в виде электронной ссылки для самостоятельного скачивания.

При получении обновлений ОС Альт СП, прошедших процедуру испытаний по требованиям безопасности информации, перед их установкой необходимо проверить подлинность и целостность полученных обновлений с использованием алгоритмов, указанных предприятием-изготовителем.

В случае невозможности установки критического обновления должны быть разработаны ограничения по применению ОС Альт СП, которые должны отражаться в нормативных документах и (или) политике безопасности организации-потребителя.

Выполнение обновления осуществляется в соответствии документом «Руководство администратора. ЛКНВ.11102-01 90 01».

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

АРМ	– автоматизированное рабочее место;
БД	– база данных;
КСЗ	– комплекс средств защиты;
НСД	– несанкционированный доступ;
ОС	– операционная система;
ПИ	– программное изделие;
ПК	– программный комплекс;
ПРД	– правило разграничения доступа;
ПЭВМ	– электронная вычислительная машина;
ФБО	– функции безопасности объекта;
ФС	– файловая система;
ЦУС	– центр управления системой.



