

Закрытое акционерное общество

"АСТРА СТ"

ТЕРМИНАЛЬНАЯ ВЫЧИСЛИТЕЛЬНАЯ

СИСТЕМА «MIRAGE»

Описание КСЗ

АСТР.51240-01 92 01

Листов 56

АННОТАЦИЯ

Данный документ содержит описание принципов работы системы Mirage, работы её подсистем, интерфейсов системы с пользователем. Кроме того, описываются используемые в системе механизмы защиты, приводится модель защиты информации.

СОДЕРЖАНИЕ

1. ОБЩЕЕ ОПИСАНИЕ ПРИНЦИПОВ РАБОТЫ СИСТЕМЫ.....	4
2. ОБЩАЯ СХЕМА КСЗ.....	8
3. МОДЕЛЬ ЗАЩИТЫ ИНФОРМАЦИИ.....	11
3.1 Мандатный контроль доступа.....	11
3.2 Дискреционный контроль доступа.....	12
3.3 Идентификация и Аутентификация.....	13
4. ОПИСАНИЕ ИНТЕРФЕЙСОВ КСЗ С ПОЛЬЗОВАТЕЛЯМИ.....	14
5. ОПИСАНИЕ ПОДСИСТЕМ.....	15
5.1 Подсистема контроля целостности.....	15
5.2 Подсистема регистрации событий.....	16
5.3 Подсистема контроля доступа.....	19
5.4 Подсистема загрузки терминалов.....	33
5.5 Подсистема терминального обмена.....	36
5.6 Подсистема печати.....	38
6. ОПИСАНИЕ МЕХАНИЗМОВ ЗАЩИТЫ.....	43
6.1 Описание механизмов идентификации и аутентификации.....	43
6.2 Описание механизма защиты памяти.....	45
6.3 Описание механизма изоляции программ.....	47
6.4 Описание механизма изоляции файловых систем.....	47
6.5 Описание средств защиты ввода/вывода на внешние носители.....	48
7. ОПИСАНИЕ ИНТЕРФЕЙСОВ МОДУЛЕЙ КСЗ.....	49
7.1 Сигналы.....	49
7.2 Каналы.....	49
7.3 Сокеты.....	49
ПРИЛОЖЕНИЕ 1.....	51
ПРИЛОЖЕНИЕ 2.....	52

1. ОБЩЕЕ ОПИСАНИЕ ПРИНЦИПОВ РАБОТЫ СИСТЕМЫ

TBC Mirage (далее просто система) – программно-аппаратный комплекс, автоматизированная система обработки информации, предназначенная для эксплуатации в локальной сети в информационных системах с высокими требованиями к защите информации от несанкционированного доступа. В системе используется технология терминального доступа к данным. В качестве АРМ используются терминалы – аппаратно ограниченные рабочие станции без каких бы то ни было средств вывода и собственных накопителей информации. Пользователи, работая на терминале, фактически работают на одном (или нескольких) серверах терминального доступа, при этом терминал получает с сервера и отображает изображение рабочего сеанса, и отправляет на сервер данные с устройств ввода (клавиатура и мышь). Таким образом хранение, обработка и ввод/вывод информации осуществляются централизованно на сервере.

Централизация хранения и обработки данных на сервере позволяет сосредоточить в одном месте функции управления безопасностью системы, делает систему легко масштабируемой и повышает общую устойчивость системы защиты. Отсутствие на АРМ устройств ввода/вывод на внешние носители снимает проблему контроля ввода/вывода данных с рабочих станций.

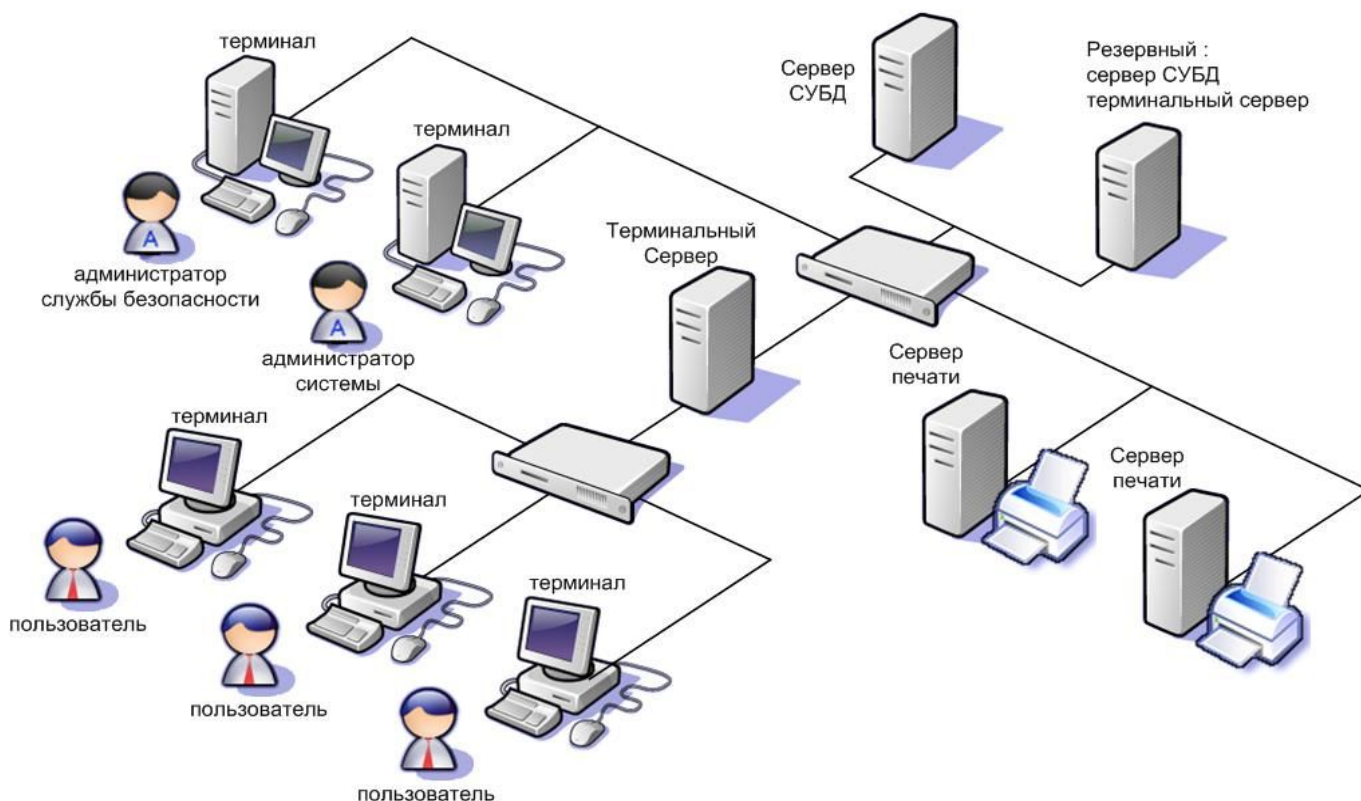


Рисунок 1.1 – Общая схема TBC Mirage

На рисунке 1.1. изображена общая схема TBC Mirage. Исходя из состава выполняемых функций, можно выделить следующие компоненты системы:

- терминальный сервер;
- сервер служебной базы данных (сервер СУБД);
- сервер печати;
- терминалы.

Все компоненты системы соединены каналами связи и могут быть разнесены географически.

Сервер терминального доступа является базовым компонентом системы Mirage. Сервер обеспечивают одновременную работу в системе множества пользователей в режиме терминального доступа и выполняет следующие базовые функции:

- авторизацию пользователей;
- идентификацию и загрузку терминалов;
- запуск и завершение пользовательского сеанса;
- выполнение пользовательских приложений;
- контроль доступа и защита данных.

Сервер служебной базы данных хранит информацию об объектах доступа, субъектах доступа, ПРД, настройках системы, журнал регистрации. Эта база данных недоступна конечному пользователю и обрабатывается служебными программами системы Mirage;

Сервер печати обеспечивает возможность контролируемого вывода документов на твёрдую копию, включающую в себя учёт, маркировку и непосредственно печать документов.

Терминалы аппаратно ограниченные рабочие станции пользователей, работа на которых ведётся в терминальном режиме. Терминалы не содержат накопителей информации, а также средств ввода/вывода на внешние носители. Загрузка терминала осуществляется по сети с терминального сервера.

Все компоненты системы функционируют под управлением ОС Linux, при этом операционная система терминала ограничена в наборе базовых функций:

- не поддерживается командная строка;
- не поддерживается работа с жёстким диском;
- не поддерживаются ввод/вывод информации на внешние носители;
- не поддерживается подключение внешних устройств.

Система защиты TBC Mirage построена на основе собственного КСЗ с применением ряда стандартных средств защиты данных ОС Linux и включает в себя следующие механизмы:

- идентификация и аутентификация объектов доступа;
- мандатный контроль доступа на основе меток конфиденциальности;
- дискреционный контроль доступа;
- аудит всех событий;
- изоляция программ;
- изоляция файловых систем пользовательских сеансов;
- обнуление высвобождаемой памяти и высвобождаемого пространства файловой системы;
- контроль ввода/вывода на внешние носители.

Правам доступа к ресурсам системы имеют только пользователи, на которых в системе заведена учётная запись и которые успешно прошли идентификацию и аутентификацию. Каждый пользователь обладает определёнными полномочиями на доступ к ресурсам и данным системы. Для разграничения доступа используются параллельно два принципа: дискреционный контроль доступа и мандатный контроль доступа.

Дискреционный контроль применяется для разграничения доступа к следующим объектам системы:

- терминал;
- защищаемый файл.

Мандатный контроль применяется для разграничения доступа к объектам:

- принтер;
- шаблон документа;
- защищаемый файл;
- процесс.

Применение двух принципов разграничения доступа позволяет быстро и эффективно решать задачу разделения и управления доступом к имеющимся ресурсам системы. Механизмы мандатного и дискреционного контроля доступа дополняются механизмами изоляции программ и файловой системы пользователя, которые обеспечивают замкнутую программную среду для каждого пользователя. Механизм изоляции программ реализован ядром операционной системы, и заключается в защите областей памяти программы от

чтения и записи со стороны других программ.

Механизм изоляции файловых систем заключается в разделении файловой системы (ФС) сервера на изолированные друг от друга области – файловые системы пользователей. Пользователь, работая в своей изолированной ФС, не имеет прямого доступа ни к корневой ФС сервера, ни к изолированным ФС других пользователей, при этом для обмена информации между изолированными ФС определено два санкционированных способа:

- папка "обмен" для обмена файлами между файловыми системами пользователей;
- папка "ввод/вывод" для ввода-вывода информации из АС.

Ввод/вывод информации из АС подлежит обязательному контролю и выполняется уполномоченным на это пользователем на сервере терминального доступа. Ввод либо вывод информации на рабочих станциях невозможен в силу использования терминального решения и программно-аппаратных ограничений, накладываемых на терминал.

Факт ввода/вывода информации из АС автоматически отмечается в журнале событий.

Система Mirage обеспечивает возможность регистрации большого числа событий, в том числе события НСД, события об ошибках, системные события, изменения полномочий и правил разграничения доступа, события ввода/вывода и т.д., при этом источниками событий могут быть:

- сервисы, интерактивные приложения и другие программные компоненты системы;
- модуль защиты (в ядре операционной системы);
- прикладные программы, модифицированные для работы в системе.

Все события сохраняются в журнале событий. Для каждого события фиксируется время свершения события, категория, значимость, идентификаторы пользователя и процесса.

Реализованные механизмы СЗИ, в совокупности с технологией терминального доступа, а также техническими средствами контроля доступа, образуют эффективный комплекс программно-технических средств по защите информации.

2. ОБЩАЯ СХЕМА КСЗ

Комплекс программно-технических средств по защите информации от НСД реализован в рамках системы защиты информации от НСД, состоящей из следующих подсистем:

- контроля доступа;
- регистрации событий;
- загрузки терминалов;
- терминального обмена;
- печати;
- обеспечения целостности.

На рисунке 2.1 схематично изображена архитектура КСЗ – показаны компоненты системы, связи между ними и их взаимодействие с операционной системой и прикладными задачами.

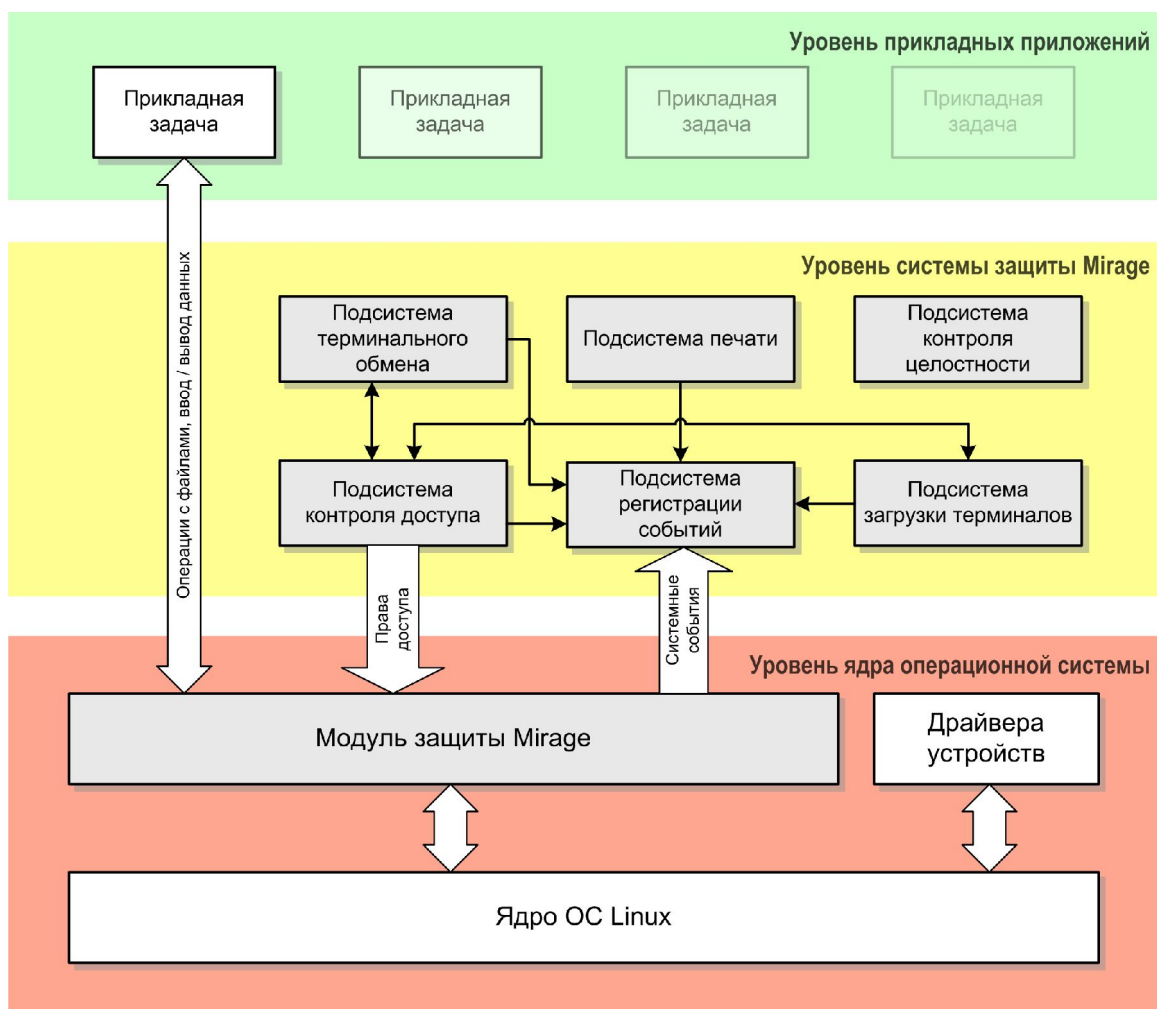


Рисунок 2.1 – Архитектура КСЗ

Прикладные задачи взаимодействуют с системой посредством системных вызовов к

ядру операционной системы. Модуль защиты загружается при старте системы, модифицирует системные вызовы и выполняет правила разграничения доступа, предоставляя или запрещая прикладным задачам доступ к ресурсам системы.

Модуль защиты взаимодействует с подсистемами контроля доступа и регистрации событий. От подсистемы контроля доступа модуль защиты получает правила разграничения доступа. События, генерируемые модулем защиты (в том числе события НСД и системные события ОС Linux), поступают в подсистему регистрации событий и сохраняются в журнале событий. Таким образом, взаимодействие прикладных задач друг с другом, с ядром, файловой системой и оборудованием контролируется и регистрируется системой защиты Mirage.

Подсистемы терминального обмена и загрузки терминалов обеспечивают работу сеанса пользователя в терминальной системе.

Подсистема загрузки терминалов обеспечивает следующие базовые функции:

- приём от пользователя данных для авторизации;
- авторизация пользователя посредством подсистемы контроля доступа;
- загрузка на терминал образа операционной системы.

Подсистема терминального обмена обеспечивает следующие базовые функции:

- инициализация графического пользовательского сеанса;
- приём и передача данных между терминалом и сервером;
- завершение сеанса по команде пользователя или администратора.

Контроль целостности программных компонентов системы реализуется подсистемой контроля целостности на этапе загрузки системы, при этом подсистема не взаимодействует с каким-либо компонентами системы Mirage, так как стартует первой. Дальнейшая загрузка системы выполняется только после успешной проверки целостности.

Подсистема печати обеспечивает контролируемый вывод документов на устройства печати, маркировку документов, регистрацию событий в журнале печати посредством подсистемы регистрации событий.

Все подсистемы СЗИ, за исключением подсистемы контроля целостности, взаимодействуют с подсистемами регистрации событий и контроля доступа.

Подсистема контроля доступа осуществляет идентификацию, аутентификацию и авторизацию субъектов доступа, а также контроль доступа субъектов доступа к защищаемым ресурсам в соответствии с правилами разграничения доступа.

Подсистема регистрации событий осуществляет приём событий от компонентов

системы и запись их в журнал событий.

3. МОДЕЛЬ ЗАЩИТЫ ИНФОРМАЦИИ

В TBC Mirage реализована многоуровневая система защиты информации от НСД, сочетающая в себе комплекс организационных мер и программно-аппаратных средств. Основными способами защиты информации от НСД являются разграничение доступа, идентификация и аутентификация.

Для разграничения доступа в системе используются две политики управления доступом: мандатная и дискреционная.

3.1 Мандатный контроль доступа

Мандатное управление доступом является отличительной особенностью системы, так как базовая операционная система Linux не поддерживает мандатный контроль доступа. Модель мандатного контроля доступа основана на базе классической модели Белла-ЛаПадуды и оперирует понятиями «метка конфиденциальности» и «уровень допуска».

Метка конфиденциальности – элемент информации, содержащейся в объекте, который характеризует уровень конфиденциальности информации в иерархической классификации.

Уровень допуска – разрешение субъекту доступа обращаться к объекту доступа определённого уровня конфиденциальности.

Мандатный контроль является обязательным и осуществляется автоматически, иными словами он производится при каждом обращении субъекта к объекту.

Модель мандатного контроля доступа в системе Mirage реализует два основных правила:

- 1) Субъект может читать объект, если уровень допуска субъекта не ниже метки конфиденциальности объекта (другими словами метка субъекта доминирует над меткой объекта);
- 2) Субъект может записать в объект, если метка конфиденциальности объекта не ниже уровня допуска субъекта (другим словами ни при каких условиях степень секретности объекта не понижается).

3.2 Дискреционный контроль доступа

Модель дискреционного контроля доступа, реализованная в системе, основана на базе классической модели Харрисона-Руззо-Ульмана. Данная модель реализует произвольное управление доступом субъектов к объектам и контроль за распространением прав доступа. Права доступа описываются явно в матрице доступа, в которой строки – субъекты, столбы – объекты, в ячейках содержится набор прав доступа субъекта к объекту. При этом модель предполагает, что субъекты одновременно являются и объектами. Это сделано для того, чтобы включить в область действия модели отношения между субъектами.

Система Mirage функционирует на базе операционной системы Linux, в которой реализована дискреционная модель разграничения доступа. Данная модель имеет ограничение, заключающееся в том, что права доступа можно задать только для трёх категорий субъектов: владельца файла, ассоциированной с файлом группы, и всех прочих.

Модель дискреционного доступа в системе Mirage, расширяет классическую модель Харрисона-Руззо-Ульмана, и, дополняя модель дискреционного доступа ОС Linux, имеет следующие особенности:

- 1) Вводится понятие "Группа пользователей". Пользователи могут состоять более чем в одной группе, а группа может состоять в другой группе. Группа не является субъектом доступа, но участвует в правилах разграничения доступа;
- 2) Вводится понятие "Категория пользователя". Пользователь системы относится к одной из трёх категорий: "Обычный пользователь", "Администратор системы", "Администратор безопасности";
- 3) Пользователи с категорией "Обычный пользователь" не могут изменять права доступа к объектам. Правом изменять права доступа субъектов к объектам обладают только пользователи с категорией "Администратор системы" и "Администратор безопасности", таким образом, распространение прав доступа строго контролируется;
- 4) Вводится понятие "Защищаемый файл". Из объектов файловой системы объектами доступа, для которых выполняется дискреционный контроль, являются только защищаемые файлы, иными словами, файл, не помеченный как защищаемый, не подлежит дискреционному контролю доступа;

- 5) Для защищаемого файла определены следующие права доступа: право чтения и право на исполнение. Доступ на запись к защищаемым файлам запрещён.

3.3 Идентификация и Аутентификация

Дискреционный и мандатный контроль доступа невозможен без обязательной идентификации субъектов доступа. Механизмы аутентификации и идентификации являются обязательными компонентами модели защиты. Ни один пользователь не может начать работу с системой, не идентифицировав себя и не предоставив системе информацию аутентификации, подтверждающую, что пользователь действительно является тем, за кого себя выдаёт.

Система Mirage, помимо классической схемы идентификации и аутентификации пользователя по логину и паролю, поддерживает идентификацию и аутентификацию по биометрическим параметрам человека, в частности – по отпечатку пальца.

Использование для идентификации и аутентификации отпечатка пальца имеет преимущества по сравнению со схемой логин/пароль:

- 1) устраняется риск утери пароля;
- 2) упрощается процедура входа в систему (пользователю не нужно запоминать и вводить пароль).

Аутентификация по отпечатку пальца может дополнять авторизацию по паролю.

Система поддерживает следующие схемы авторизации:

- 1) идентификация по логину, аутентификация по паролю;
- 2) идентификация по логину, аутентификация по отпечатку пальца;
- 3) идентификация по логину, аутентификация по паролю и отпечатку пальца;
- 4) идентификация и аутентификация по отпечатку пальца.

Конкретную схему авторизации устанавливает администратор в соответствии с выбранной политикой безопасности.

4. ОПИСАНИЕ ИНТЕРФЕЙСОВ КСЗ С ПОЛЬЗОВАТЕЛЯМИ

Взаимодействие КСЗ и пользователя системы может производиться в трёх режимах:

1. Режим работы на консоли сервера. Используется системным администратором при установке системы, её первоначальном конфигурировании и диагностике неисправностей. В этом режиме администратору предоставляется стандартный интерфейс командной строки UNIX систем и псевдографический интерфейс конфигурирования системы, предоставляемой программой "Системный конфигурактор". Ввод данных производится с клавиатуры сервера, вывод – через дисплей сервера.
2. Режим загрузки терминала. Используется всеми пользователями системы в схеме авторизации LP. Пользователю предоставляется ограниченный интерфейс по вводу логина и пароля. Ввод данных производится с клавиатуры терминала, вывод – через дисплей терминала.
3. Режим терминальной работы. Основной режим, используемый всем пользователями, зарегистрированными в системе. Пользователю предоставляется графический интерфейс, виртуальный рабочий стол. Ввод данных производится через клавиатуру терминала, вывод – через дисплей терминала.

5. ОПИСАНИЕ ПОДСИСТЕМ

5.1 Подсистема контроля целостности

5.1.1 Описание назначения подсистемы

Подсистема контроля целостности предназначена для контроля целостности объектов файловой системы (файлов и директорий), входящих в состав системы Mirage.

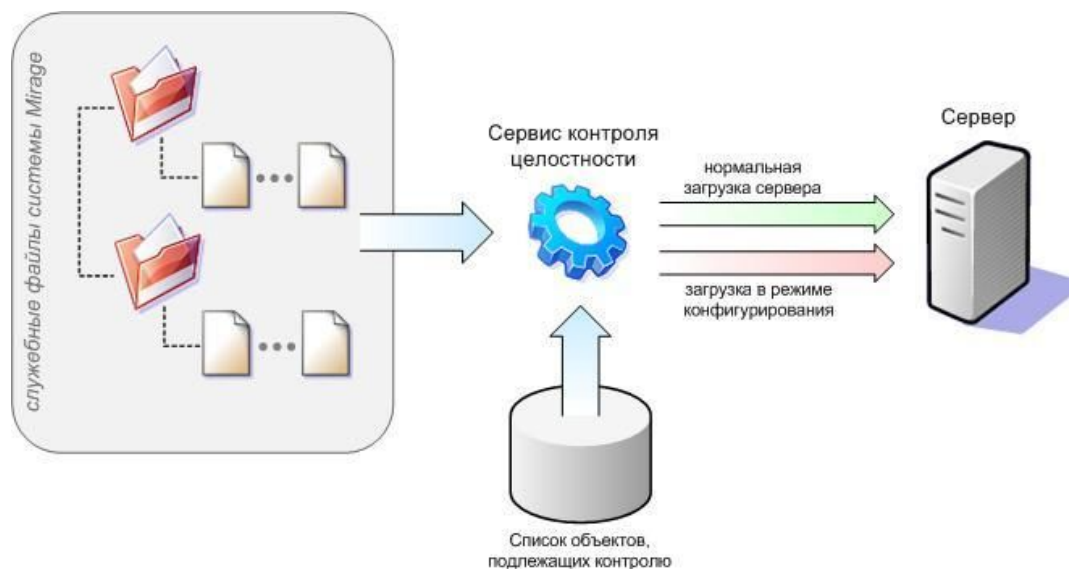


Рисунок 5.1 – Условная схема работы подсистемы контроля целостности

Подсистема контроля целостности выполняет следующие функции:

- подсчитывает контрольные суммы объектов файловой системы;
- проверяет на целостность объекты файловой системы.

5.1.2 Состав подсистемы

Подсистема контроля целостности реализована сервисом контроля целостности.

5.1.3 Описание работы

Сервис контроля целостности производит контроль целостности объектов файловой системы при загрузке операционной системы сервера. Обязательному контролю подлежат файлы и каталоги, входящие в состав системы Mirage.

В случае, если будет обнаружено изменение любого из объектов, на экран выводится информационное сообщение, и загрузка сервера приостанавливается до нажатия любой клавиши. Дальнейшая загрузка сервера возможна только в режиме конфигурирования.

Список объектов системы, подлежащих контролю, автоматически корректируется в случае установки/удаления каких-либо компонентов системы программой установки. Сам список объектов также подлежит обязательному контролю. Контроль целостности объектов производится по их контрольной сумме.

В режиме конфигурирования с помощью сервиса контроля целостности можно пересчитать контрольные суммы файлов и каталогов, в том случае если файлы или каталоги системы были сознательно изменены и требуется пересчитать контрольные суммы.

5.2 Подсистема регистрации событий

5.2.1 Описание назначения подсистемы

Подсистема регистрации событий предназначена для:

- ведения журнала событий по всем происходящим в системе событиям;
- организации доступа администратора системы к журналу событий;
- своевременной сигнализации о значимых событиях.

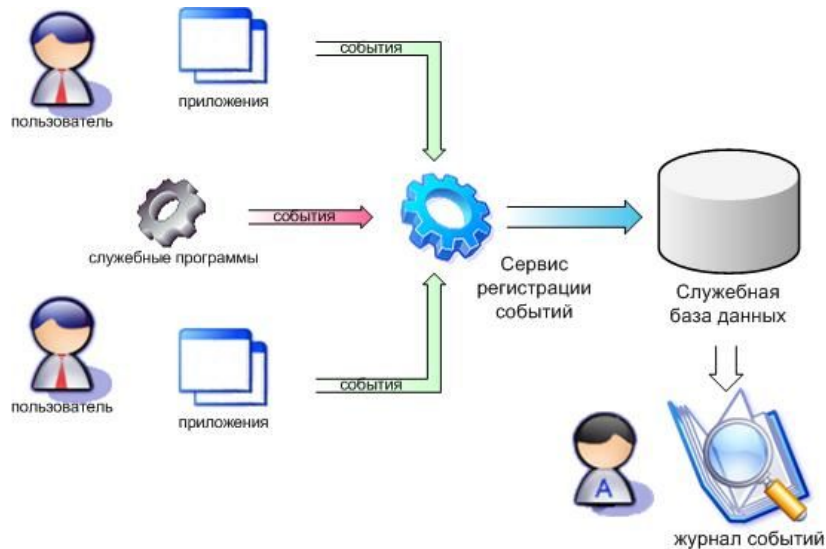


Рисунок 5.2 – Условная схема работы подсистемы регистрации событий

Для решения этих задач в подсистеме регистрации событий реализованы следующие функции:

- Приём информации о событиях от программных компонентов системы;
- Сохранение событий в журнале событий;
- Автоматическое удаление устаревших событий;
- Предоставление администратору графического интерфейса для просмотра журнала событий и навигации по нему;
- Сигнализирование о значительных событиях через графического уведомления.

5.2.2 Состав подсистемы

Подсистема регистрации событий состоит из следующих функциональных частей:

1. Сервис регистрации событий;
2. Программа просмотра событий;
3. Сигнализатор.

Подробное описание этих компонентов и протоколов их взаимодействия можно найти в документе «Руководство по КСЗ» и соответствующих разделах документа «Описание программы».

5.2.3 Описание работы

Все происходящие в системе события принимаются сервисом регистрации и регистрируются в журнале событий. Генерировать события могут:

- Сервисы, интерактивные приложения и другие программные компоненты системы;
- Модуль защиты (в ядре операционной системы);
- Прикладные программы, модифицированные для работы в системе.

Непосредственную запись в журнал событий, располагающийся в служебной базе данных, осуществляет сервис регистрации событий. Каждое событие описывается структурой, содержащей следующие поля:

- Штамп времени;
- Идентификатор приложения (название или путь его запуска), зарегистрировавшего событие;
- Идентификатор процесса работающего приложения;
- Идентификатор пользователя, запустившего это приложение;
- Категория события;
- Значимость события;
- Текстовое сообщение – комментарий к событию.

Все события классифицируются по категориям и значимости. Категория определяет тип события, его принадлежность к какой-либо подсистеме и/или группе событий.

Возможные категории событий:

- Вход, выход пользователей;
- Запуск, завершение программ;
- Печать документов;
- Доступ к объектам доступа;
- События, связанные с изменением состояния объектов и субъектов, изменением полномочий субъектов доступа;
- Прочие события (в том числе и просто сообщения об ошибках).

Значимость определяет важность, критичность события. Возможные классы значимости:

- Отладочное сообщение. Предназначено для трассировки и контроля работы новых разрабатываемых приложений;
- Информация. Событие носит лишь информативный, уведомляющий характер;
- Предупреждение. Событие носит предупреждающий характер;
- Ошибка. Информирование об отклонениях от нормальной работы;
- Несанкционированный доступ;
- Критическая ошибка. Событие информирует о невозможности дальнейшей работы какой-либо функциональной части (например, не может продолжить свою работу один из сервисов системы).

Текстовое сообщение в составе события содержит пояснение к данному событию, условия его возникновения, коды ошибок и прочую информацию, способную помочь администратору и разработчику исправить возможные ошибки или неполадки, настроить и оптимизировать систему.

Журнал событий хранится в служебной базе данных системы. Для навигации по журналу событий используется программа просмотра событий.

Администратор может оперативно получать информацию о событиях, удовлетворяющих заданному критерию фильтрации, с помощью сигнализатора, который следит за потоком событий, поступающих в журнал событий и своевременно визуально информирует об этом администратора.

5.3 Подсистема контроля доступа

5.3.1 Описание назначения

Подсистема контроля доступа предназначена для идентификации, аутентификации и авторизации субъектов доступа, а также контроля доступа субъектов доступа к защищаемым ресурсам в соответствии с правилами разграничения доступа.

Подсистема контроля доступа выполняет следующие функции:

1. Идентификацию субъектов доступа и объектов доступа;
2. Идентификация терминалов и аппаратных узлов сервера;
3. Аутентификация и авторизация пользователей;
4. Проверка статусов пользователей, сеансов, принтеров по запросу сервисов, интерактивных приложений и других программных компонентов системы;
5. Управление ПРД и контроль выполнения ПРД;
6. Обнуление освобождаемых областей памяти на внешних носителях информации.

5.3.2 Состав подсистемы

Подсистема контроля доступа состоит из следующих функциональных частей:

1. Сервис авторизации;
2. Сервис сетевой авторизации;
3. Модуль защиты;
4. Агент загрузки терминала;
5. Центр управления и центр мониторинга;
6. Утилита настройки метки безопасности;

Подробное описание этих компонентов и протоколов их взаимодействия можно найти в соответствующих разделах документа "Описание программы".

5.3.3 Описание работы

Все действия по идентификации, авторизации субъектов системы выполняет сервис авторизации, к которому обращаются программные компоненты системы.

Источниками запросов на авторизацию могут быть:

- сервис загрузки терминалов;
- сервис сетевой авторизации;

– сеансовый менеджер.

Задача агента загрузки терминалов – приём данных для авторизации и запуск операционной системы терминала. После включения терминала агент загрузки терминалов допускает его загрузку только после успешного прохождения процедуры авторизации. В зависимости от модели терминала, агент загрузки терминала может располагаться в одном из ниже перечисленных мест:

- в области ПЗУ BIOS материнской платы терминала;
- в области ПЗУ платы сетевого контроллера, установленного в терминал (BootROM);
- в загрузочной области FLASH-диска, установленного в терминал.

В случае расположения агента в области ПЗУ платы сетевого контроллера, агент, помимо всего прочего, выполняет функцию контроля загрузки, блокируя возможность загрузки операционной системы с других носителей.

Модуль защиты в составе ядра операционной системы реализует политику защиты объектов файловой системы в соответствии с правилами разграничения доступа.

Для контроля доступа в системе используются параллельно два принципа: дискреционный контроль доступа и мандатный контроль доступа. В таблице 5.1 приведён перечень объектов доступа и реализованных в системе способов контроля доступа к ним.

Таблица 5.1 – Контроль доступа субъектов доступа к объектам доступа

<i>Объект доступа</i>	<i>Контроль доступа</i>		<i>Примечание</i>
	<i>Дискреционный</i>	<i>Мандатный</i>	
Терминал	+	–	Контролируется возможность входа пользователя в систему с терминала.
Принтер	–	+	Контролируется печать документов пользователями.
Шаблон документа	–	+	Контролируется применение шаблонов к документам при их печати.

<i>Объект доступа</i>	<i>Контроль доступа</i>		<i>Примечание</i>
	<i>Дискреционный</i>	<i>Мандатный</i>	
Защищаемый файл	+	+	Контроль дискреционного доступа предусматривает контроль чтения и запуска и автозапуска. Контроль мандатного доступа позволяет ограничить пользователю все виды доступа к файлу.
Процесс	-	+	Контроль потоков. Метки процессам назначаются автоматически при открытии процессами конфиденциальных файлов.

Знаком "+" в таблице обозначено наличие контроля в системе, знаком "-" - его отсутствие.

Использование сразу двух принципов контроля доступа позволяет быстро и эффективно решать задачу разделения и управления доступом к имеющимся ресурсам системы.

5.3.3.1 Дискреционный контроль доступа

Матрица доступа системы хранится в служебной базе данных в виде списков связей субъектов доступа с объектами системы. Просмотр и конфигурирование этих связей осуществляется в программе "Центр управления". Модуль безопасности получает матрицу доступа из служебной базы данных через сервис авторизации.

На рисунке 5.3 изображена условная схема работы дискреционного контроля доступа.

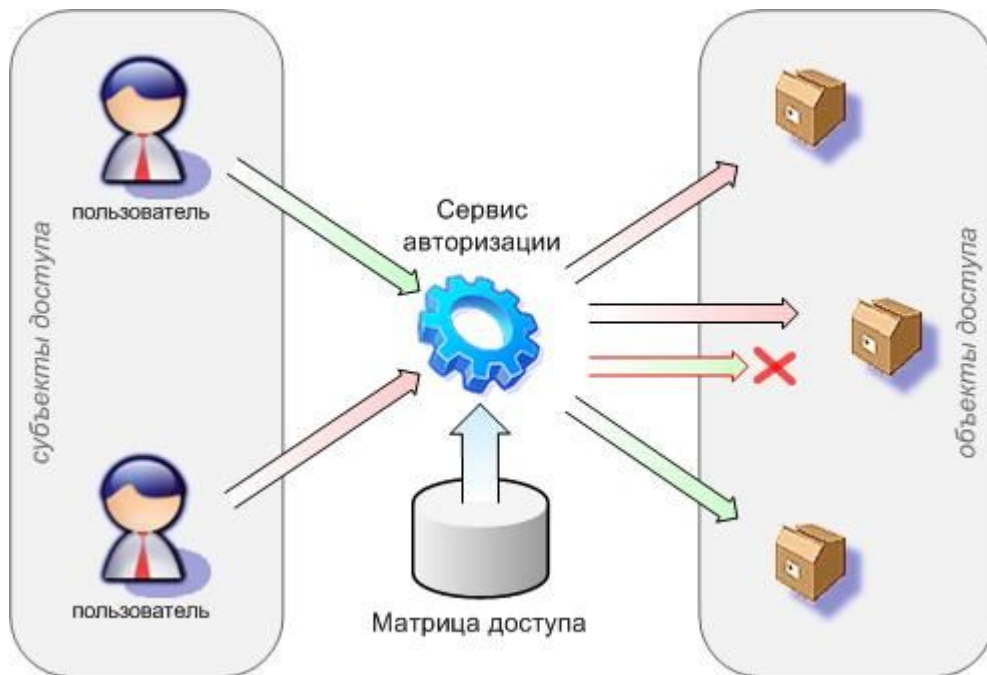


Рисунок 5.3 – Условная схема работы дискреционного контроля доступа

Следует учесть следующие особенности реализации дискреционного контроля доступа:

1. В отношении входа пользователей с терминалов, исполнения программ, чтения защищаемых файлов действует принцип "что не разрешено – запрещено". Так например, если у пользователя *U* не настроен доступ к терминалу *T* (не создана соответствующая связь в Центре управления), то доступ этого пользователя к этому терминалу запрещён.
2. Доступ на запись к защищаемым файлам запрещён всем пользователям системы.

Для удобства администрирования и разделения прав доступа в системе существует понятие "*Группа пользователей*". Такой подход позволяет менять дискреционные права доступа сразу множества пользователей, изменяя права доступа соответствующих групп пользователей. Права, которыми обладает группа, наследуются всеми её членами. При этом следует учитывать следующие аспекты:

- Группа может, в свою очередь, состоять в другой группе (или группах).
- Пользователь может состоять в нескольких группах одновременно. При этом наследование прав происходит по принципу приоритета запретов.

Например, если пользователь *U* находится в группах *G1* и *G2*, причём группе *G1* разрешено запускать приложение *A*, а группе *G2* – запрещено, то пользователю *U*

запрещено запускать приложение А.

5.3.3.2 Мандатный контроль доступа

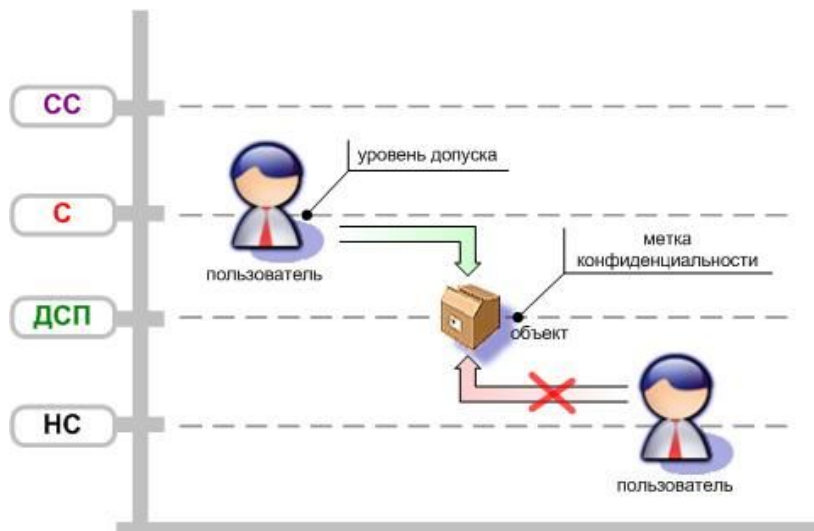


Рисунок 5.4 – Условная схема работы мандатного контроля доступа

Принцип мандатного управления доступом реализован в системе следующим образом: объектам системы назначаются метки конфиденциальности (безопасности), субъектам системы назначаются в соответствие уровни допуска. Метка конфиденциальности и уровни допуска пользователей могут быть следующих типов (в скобках указано принятое сокращение):

- НЕСЕКРЕТНО (“НС”);
- ДЛЯ СЛУЖЕБНОГО ПОЛЬЗОВАНИЯ (“ДСП”);
- СЕКРЕТНО (“С”);
- СОВЕРШЕННО СЕКРЕТНО (“СС”).

Постоянными носителями меток безопасности в системе являются:

- принтеры;
- документы;
- файлы и приложения;
- шаблоны документов.

Метки безопасности принтеров, документов и их шаблонов хранятся в служебной базе данных. Метки безопасности файлов и директорий хранятся в файловой системе (более подробно в документе “Описание программы”, раздел “Модуль защиты”). Порядок назначения метки безопасности файлу описан в документе “Руководство пользователя”, п. 3.2.

Мандатный контроль доступа осуществляется при выполнении следующих операций:

1. Создание печатного документа (первый этап печати документа).

Контролируется, что метка безопасности печатного документа не превышает уровень допуска пользователя, создающего этот документ. Описание контроля этой операции приведено в таблице 5.2.

Таблица 5.2 – Мандатный контроль создания печатного документа

<i>Документ</i>	<i>"НС"</i>	<i>"ДСП"</i>	<i>"С"</i>	<i>"СС"</i>
<i>Пользователь</i>				
<i>"НС"</i>	+	-	-	-
<i>"ДСП"</i>	+	+	-	-
<i>"С"</i>	+	+	+	-
<i>"СС"</i>	+	+	+	+

2. Назначение шаблона печатному документу.

Контролируется, что метка безопасности шаблона документа не ниже метки безопасности документа. Описание контроля этой операции приведено в таблице 5.3.

Таблица 5.3 – Мандатный контроль назначения шаблона печатному документу

<i>Документ</i>	<i>НС</i>	<i>ДСП</i>	<i>С</i>	<i>СС</i>
<i>Шаблон</i>				
<i>НС</i>	+	-	-	-
<i>ДСП</i>	+	+	-	-
<i>С</i>	+	+	+	-
<i>СС</i>	+	+	+	+

3. Печать документов.

Контролируется, что метка безопасности принтера, на котором будет произведена печать документа, не ниже метки безопасности самого печатного документа. Описание контроля этой операции приведено в таблице 5.4.

Таблица 5.4 – Мандатный контроль печати документа

<i>Документ</i>	<i>НС</i>	<i>ДСП</i>	<i>С</i>	<i>СС</i>
<i>Принтер</i>				
<i>НС</i>	+	-	-	-
<i>ДСП</i>	+	+	-	-
<i>С</i>	+	+	+	-
<i>СС</i>	+	+	+	+

4. Исполнение программ.

Контролируется, что пользователь не может запустить программу, исполняемый файл которой имеет метку безопасности, превышающую уровень допуска пользователя. Описание контроля этой операции приведено в таблице 5.5.

Таблица 5.5 – Мандатный контроль исполнения программ

<i>Программа</i>	<i>НС</i>	<i>ДСП</i>	<i>С</i>	<i>СС</i>
<i>Пользователь</i>				
<i>НС</i>	+	-	-	-
<i>ДСП</i>	+	+	-	-
<i>С</i>	+	+	+	-
<i>СС</i>	+	+	+	+

5. Открытие файла программой.

Открытие файла программой – сложный процесс, контроль которого включает в себя сопоставление меток безопасности открываемого файла, процесса программы, уровня допуска пользователя, от имени которого была запущена программа, анализ меток и режимов открытия уже открытых программой файлов и пр. Подробнее контроль открытия файла описан в п. 5.3.3.5

Порядок назначения уровней допуска пользователям, меток безопасности принтерам, документам и шаблонам документов, файлам и директориям описан в документе "Руководство по КСЗ".

5.3.3.3 Режим "Фиктивная запись" директории

Для директорий в системе существует возможность назначения специального режима "Фиктивная запись". Включение этого режима изменяет логику работы контроля доступа к файлам таким образом, что в случае, если доступ к файлу на запись запрещён, программе, выполнявшей запись, сообщается, что доступ разрешён и операция записи завершилась успешно, однако фактически запись не выполняется.

Обработка режима фиктивной записи производится в два этапа:

- при открытии файла на запись, в случае если по ПРД доступ запрещён, файл успешно открывается и приложению возвращается дескриптор открытого файла;
- при выполнении операции записи в этот файл, запись не производится, но приложению сообщается, что все данные успешно записаны.

Включение режима фиктивной записи для директорий позволяет, без снижения уровня защищённости, добиться работоспособности множества прикладных приложений, не приспособленных для работы в системе с мандатным контролем доступа к файлам.

Включение режима фиктивной записи производится следующим образом:

- вручную пользователем в обозревателе файлов;
- автоматически модулем безопасности при создании директории, название которой начинается с точки (например ".gconf2").

В директориях, название которых начинается с символа ".", прикладные программы хранят конфигурационные и временные файлы. Эти файлы не содержат конфиденциальных данных, так как хранят только настройки, необходимые для функционирования приложения. В случае, если приложение работает с конфиденциальными данными, необходимо предотвратить запись данных в конфигурационные файлы и, следовательно, повышение их метки, что и обеспечивается назначением директории режима фиктивной записи.

Выключение режима фиктивной записи для директории производится пользователем в обозревателе файлов.

5.3.3.4 Контроль потоков информации

Кроме принтеров, файлов, документов и их шаблонов, метки безопасности могут иметь и процессы программ. Система присваивает метки безопасности процессам приложений

пользователя в тех случаях, когда эти процессы открывают файлы, содержащие конфиденциальную информацию. Узнать метку безопасности процесса можно в программе "Центр мониторинга".

При этом необходимо знать следующие аспекты, реализующие контроль потоков информации:

- Метка конфиденциальности процесса не может превышать уровень допуска пользователя, так как пользователь не может открыть файлы, содержащие информацию уровня конфиденциальности выше, чем уровень допуска пользователя;
- Процесс не может открыть на запись файл, метка безопасности которого ниже метки безопасности процесса;
- Процесс не может открыть на чтение файл, метка безопасности которого выше метки безопасности процесса, в тех случаях, когда этот процесс имеет хотя бы один открытый на запись файл, метка безопасности которого ниже метки открываемого файла;
- Процесс не может открыть на запись файл, если последний уже открыт другим процессом.

5.3.3.5 Контроль создания файлов

Контроль создания файла происходит в соответствии с алгоритмом, блок-схема которого приведена на рисунке 5.5.

На рисунке 5.5:

U – уровень допуска пользователя;

F – метка безопасности файла;

D – наивысшая метка безопасности из всех меток родительских директорий.

Например, для директории "/home/test/1/" параметр D может быть вычислен как:

$D = \max(\text{метка } "/home/", \text{метка } "/home/test/", \text{метка } "/home/test/1/")$

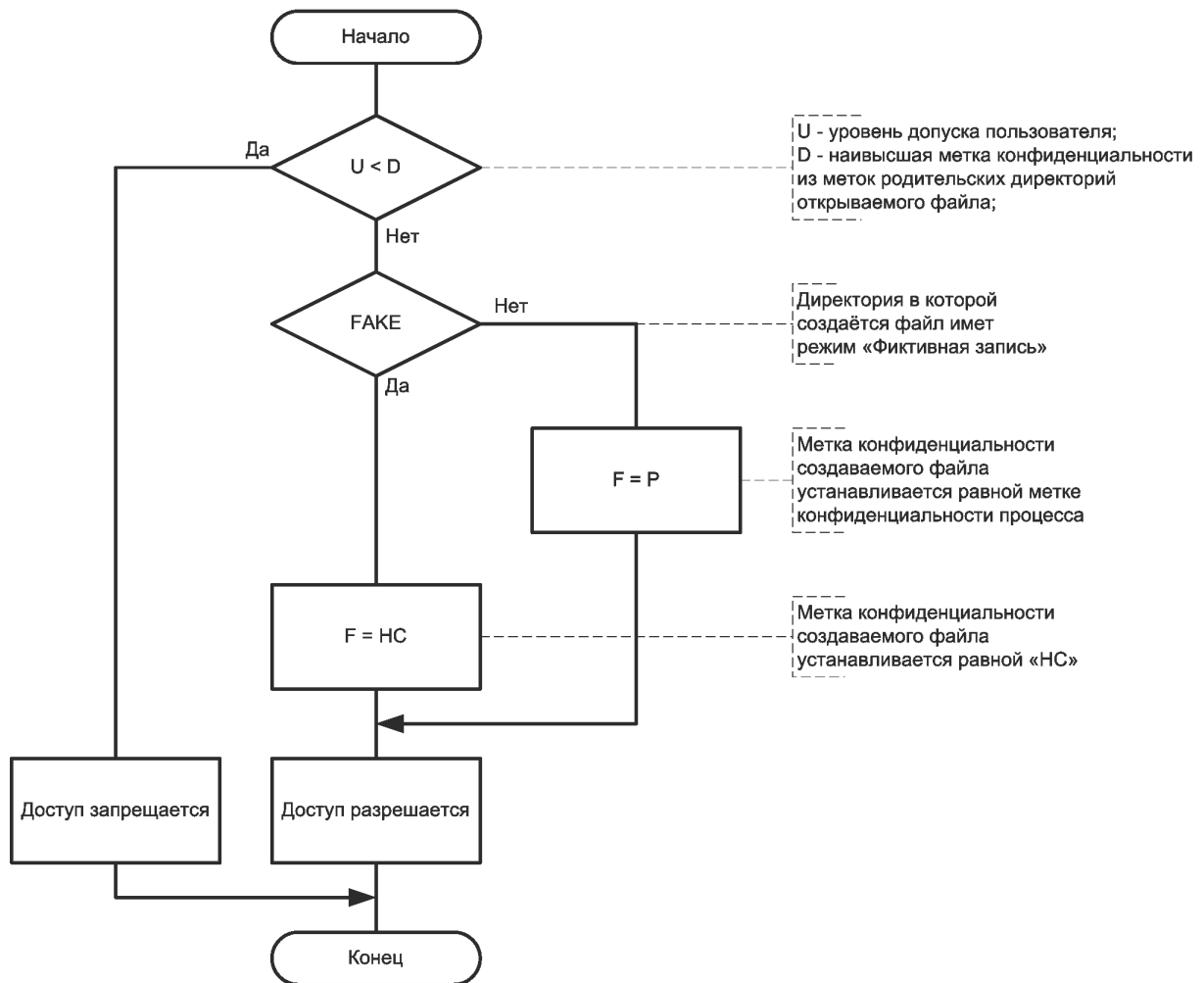


Рисунок 5.5 - Блок-схема алгоритма контроля создания файла

5.3.3.6 Контроль открытия файлов

Контроль открытия файла происходит в соответствии с алгоритмом, блок-схема которого приведена на рисунке 5.6.

На рисунке 5.5:

U – уровень допуска пользователя;

F – метка безопасности файла;

D – наивысшая метка безопасности из всех меток родительских директорий;

P – метка безопасности процесса, равная наивысшей метке безопасности из всех меток файлов, когда-либо открытым процессом в режиме чтения (в том числе “чтения-записи”);

FPF – при открытии процессом файла на чтение проверяется, открыт ли этим процессом какой-либо другой файл, метка безопасности которого ниже метки конфиденциальности открываемого файла.

открытии.

5.3.3.7 Категории пользователей

Пользователи, зарегистрированные в системе, могут иметь одну из возможных категорий (в скобках указано условное обозначение):

- обычный пользователь (П);
- администратор системы (А);
- администратор безопасности (Б).

В таблице 5.6 представлен перечень привилегий пользователей в зависимости от их категорий.

Таблица 5.6 – Привилегии различных категорий пользователей

<i>Операция, действие</i>	<i>П</i>	<i>А</i>	<i>Б</i>
<i>Создание, удаление, правка учётных записей</i>			
– групп пользователей	–	+	+
– пользователей	–	–	+
– терминалов	–	+	+
– принтеров	–	+	+
– защищаемых файлов	–	+	+
<i>Печать документов</i>			
Создание, удаление, правка шаблонов	–	+	+
Печать документов	+	+	+
Отмена печати своего документа ¹	+	+	+
Отмена печати любого документа ¹	–	+	+
<i>Блокирование, разблокирование</i>			
– учётных записей пользователей	–	+	+
– учётных записей принтеров	–	+	+
– своего сеанса	+	+	+
– любого сеанса (блокирование администратора)	–	+	+
<i>Завершение программ и сеансов</i>			
Принудительное завершение программы пользователя	–	+	+
Принудительное завершение сеанса пользователя	–	+	+
<i>Работа с ПРД</i>			
Настройка дискреционных прав на вход пользователей в систему с терминала	–	+	+

<i>Операция, действие</i>	<i>П</i>	<i>А</i>	<i>Б</i>
Настройка дискреционных прав на защищаемые файлы	-	-	+
Установка метки безопасности файла ²	+	+	+
Повышение метки безопасности файла ²	+	+	+
Понижение метки безопасности файла	-	-	+
<i>Запуск служебных программ³</i>			
Запуск центра управления	-	+	+
Запуск центра мониторинга	-	+	+
Запуск программы просмотра событий	-	+	+
Запуск менеджера печати	+	+	+
Автозапуск сигнализатора ⁴	-	+	+

1 – Отменить печать документа можно только для документов в состоянии "Ожидание".

2 – Установить или повысить метку безопасности файла можно только до уровня допуска пользователя.

3 – Запуск программ возможен только если это разрешено ПРД.

4 – По умолчанию сигнализатор запускается автоматически у всех пользователей группы "Сигнализация".

5.4 Подсистема загрузки терминалов

5.4.1 Описание назначение подсистемы

Подсистема загрузки терминалов предназначена для обеспечения внесансового взаимодействия терминала с сервером терминального доступа, обеспечения загрузки терминала и авторизованного входа пользователей в систему. Условная схема работы подсистемы представлена на рисунок 5.7.

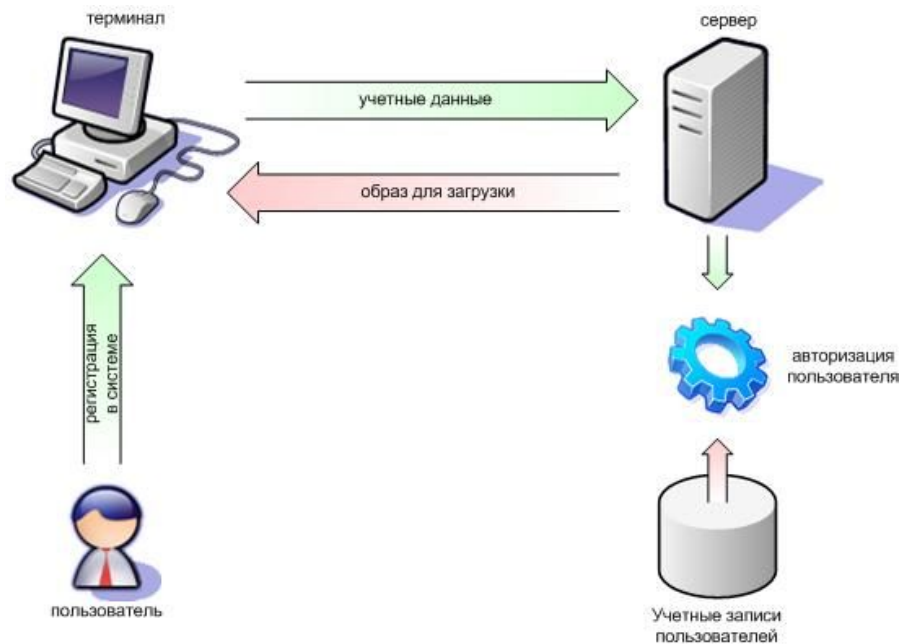


Рисунок 5.7 – Условная схема работы подсистемы загрузки терминалов

Подсистема загрузки терминала выполняет следующие функции:

- принимает от пользователя данные для авторизации;
- производит авторизацию пользователя (при непосредственном участии подсистемы контроля доступа);
- загружает на терминал образ специальной, функционально ограниченной ОС, соответствующей модели терминала;
- распаковывает образ ОС в оперативной памяти и передает управление ядру операционной системы;
- запускает программу – графический терминальный клиент для инициализации терминального обмена.

5.4.2 Состав подсистемы

Подсистема загрузки терминалов состоит из следующих компонентов:

- Агент загрузки терминала;
- Сервис загрузки терминалов;
- Функционально ограниченная ОС терминала.

Подробное описание этих компонентов и протоколов их взаимодействия можно найти в соответствующих разделах документа “Описание программы”.

5.4.3 Описание работы

После включения терминала и успешного прохождения процедуры самотестирования запускается агент загрузки терминала, который взаимодействует с пользователем в интерактивном режиме, запрашивая данные для авторизации, и запрашивает у сервиса загрузки терминалов разрешение на дальнейшую загрузку и вход в систему.

Получив запрос на загрузку, сервис загрузки терминалов запрашивает и получает от Агента данные для авторизации, передаёт их в подсистему контроля доступа и получает ответ о разрешении или запрете дальнейшей загрузки терминала и входа пользователя в систему.

Если загрузка разрешена, сервис загрузки терминала передаёт агенту данные для загрузки терминала и входа пользователя в систему:

- IP-адрес, назначаемый терминалу;
- маску подсети;
- IP-адрес терминального сервера.

Далее сервис загрузки терминалов передаёт по запросу агента образ ОС, необходимой для дальнейших стадий загрузки терминала. Эта передача занимает несколько секунд.

ОС терминала – функционально ограниченная операционная система Linux, работающая на терминальном компьютере, в которой недоступны следующие функции:

- командная строка (консоль);
- запуск произвольных программ;
- работа с жестким диском;
- ввод-вывод данных на внешние носители;
- подключение любых внешних устройств.

Назначение этой ОС – обеспечение работы графического терминального клиента.

Полностью получив образ ОС, агент распаковывает его в своей оперативной памяти и передаёт управление ядру операционной системы.

Получив управление, ядро инициализирует необходимые для терминальной работы устройства, такие как видеоконтроллер, сетевой контроллер и т.п., и создаёт в памяти виртуальную файловую систему, которая содержит лишь минимальный набор программ, необходимых для запуска и работы графического терминального клиента. Загрузка операционной системы (и терминала) завершается запуском программы, графического терминального клиента, и началом сеанса пользователя.

5.5 Подсистема терминального обмена

5.5.1 Назначение подсистемы

Подсистема терминального обмена предназначена для обеспечения взаимодействия терминала с терминальным сервером в рамках терминального сеанса. Условная схема работы подсистемы представлена на рисунок 5.8.



Рисунок 5.8 – Условная схема работы подсистемы терминального обмена

Подсистема терминального обмена выполняет следующие функции:

- взаимодействует с подсистемой контроля доступа для авторизации пользователя при его входе в систему;
- запускает графический сервер, оконный и файловый менеджеры и панель рабочего стола;
- передает с терминала на графический сервер события от клавиатуры и мыши;
- передает с графического сервера на терминал изображение рабочего стола;
- обеспечивает блокирование и разблокирование сеанса по требованию пользователя (при участии подсистемы контроля доступа);
- обеспечивает принудительное блокирование и разблокирование сеанса по требованию администратора (при участии подсистемы контроля доступа);
- обеспечивает завершение сеанса.

5.5.2 Состав подсистемы

Подсистема терминального обмена состоит из следующих компонентов:

- Сервис терминального обмена;

- Графический терминальный клиент;
- Графический сервер;
- Канал терминального подключения;
- Сеансовый менеджер;
- Рабочий стол Gnome;
- Оконный менеджер;
- Файловый менеджер;
- Блокиратор сеанса;
- Панель рабочего стола и программа формирования главного меню.

Подробное описание этих компонентов и протоколов их взаимодействия можно найти в соответствующих разделах документа "Описание программы".

5.5.3 Описание работы

Работа клиентской части:

Запуск графического терминального клиента – финальная стадия загрузки терминала (см п. 5.4.3, "Описание работы подсистемы загрузки терминалов"). Графический терминальный клиент выполняет следующие функции:

- устанавливает соединение с сервисом терминального обмена;
- отправляет события клавиатуры и мыши сервису терминального обмена;
- периодически запрашивает и получает от сервиса терминального обмена изображение рабочей среды пользователя;
- отрисовывает на терминале изображение рабочей среды пользователя;
- перезагружает терминал, в случае возникновения неисправимых сбоев.

Работа серверной части:

Задачей серверной части является приём соединений от клиентов, организация работы множества сеансов. После приёма соединения от графического терминального клиента сервис терминального обмена запускает для этого клиента канал терминального подключения, который организует для авторизованного пользователя выделенный канал терминального обмена.

Канал терминального подключения запускает сеансовый менеджер, если сеанс был завершён, или подключается к уже работающему сеансу, если пользователь вышел из

системы без завершения своего сеанса.

Задачей сеансового менеджера является организация и управление сеансом пользователя. При своём запуске сеансовый менеджер обеспечивает запуск необходимых для работы сеанса программ и приложений, таких как графический сервер, рабочий стол Gnome, оконный и файловый менеджер, панель рабочего стола и программа формирования главного меню пользователя. В зависимости от производительности сервера терминального доступа, процесс инициализации нового сеанса может занимать 5-10 секунд. Результатом работы сеансового менеджера и всех запущенных им программ является рабочая среда пользователя. Кроме того, сеансовый менеджер производит управление сеансом (блокирование, завершение), получая соответствующие команды от блокиратора сеанса, панели рабочего стола и подсистемы контроля доступа.

Блокирование сеанса не прерывает работы всех запущенных сеансовым менеджером и самим пользователем программ, в то время как завершение сеанса приводит к их завершению.

Сбой связи между терминалом и сервером (то есть между графическим терминальным клиентом и каналом терминального подключения) приведёт к перезагрузке терминала и повторному прохождению пользователем авторизации. Сеанс пользователя в это время будет работать без изменений.

5.6 Подсистема печати

5.6.1 Описание назначения

Подсистема печати предназначена для обеспечения процесса печати документов. Условная схема работы подсистемы приведена на рисунке 5.9.

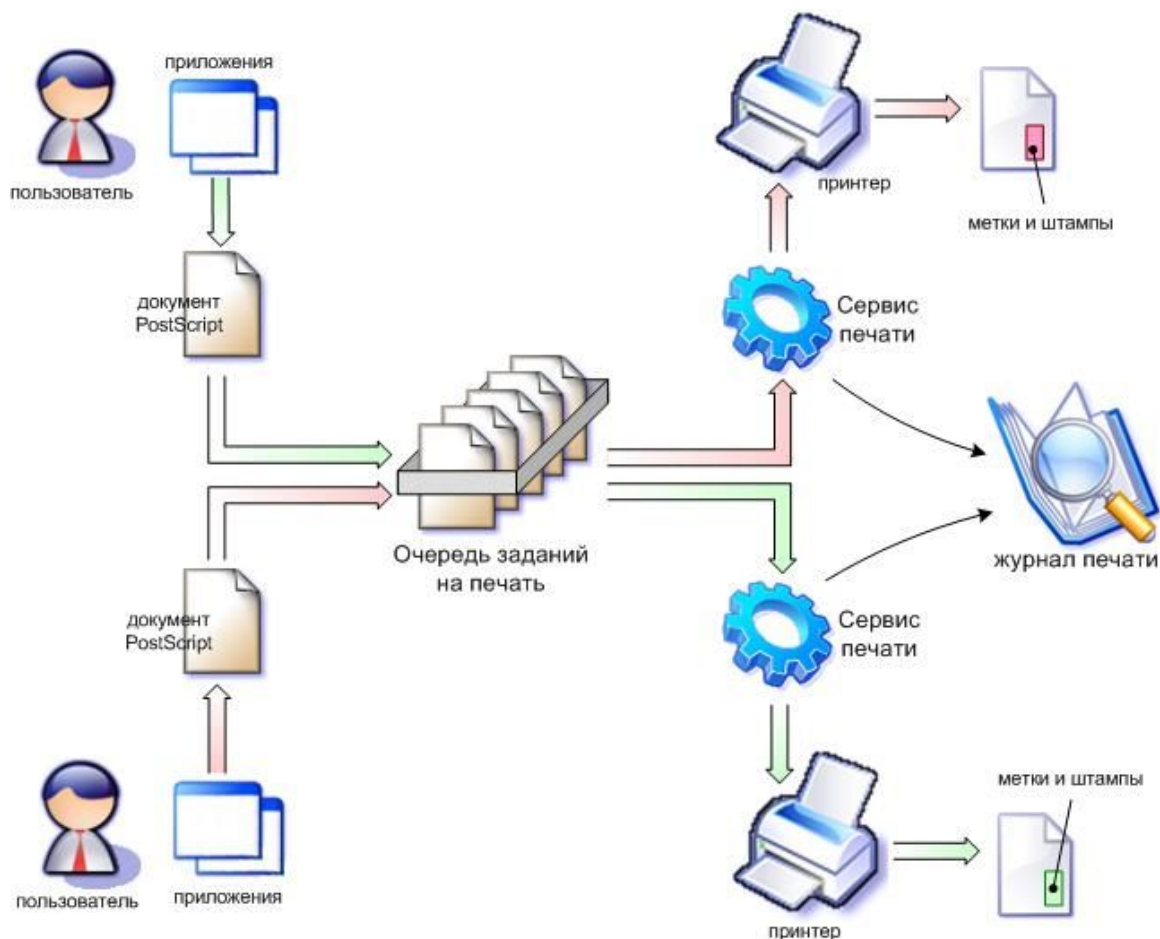


Рисунок 5.9 – Условная схема работы подсистемы печати

Подсистема печати выполняет следующие функции:

- принимает документ от прикладного ПО и помещает его в служебную БД в виде печатного документа;
- предоставляет пользователю графический интерфейс для управления списком готовых к печати документов и формирования заданий на печать документов;
- при необходимости, преобразует документ к формату Postscript;
- маркирует документ необходимыми штампами в соответствии с выбранным шаблоном документа;
- ведёт журнал печати документов;
- отправляет документ на принтер;
- регистрирует печать документов (посредством подсистемы регистрации событий).

5.6.2 Состав подсистемы

Подсистема печати состоит из следующих компонентов:

- сервис печати;
- утилита печати;
- утилита настройки параметров документов;
- менеджер печати;
- редактор шаблонов.

5.6.3 Описание работы

Сервис печати устанавливается на выделенном сервере печати. Остальные компоненты подсистемы печати входят в состав программного обеспечения сервера терминального доступа. Допускается полное объединение серверов терминального доступа и печати на одном компьютере. В рабочей терминальной системе возможно произвольное количество серверов печати, а один сервер печати может обслуживать произвольное количество подключенных к нему принтеров.

Программное обеспечение сервера терминального доступа взаимодействует с сервером печати, отдавая ему команды на выполнение заданий печати. В свою очередь, сервер печати взаимодействует с сервером СУБД: получает документы из служебной базы данных и помещает результат обработки документа в журнал печати.

Печать документов в системе производится в два этапа. На первом этапе приложения посредством утилиты печати помещают документы в служебную БД, формируя список готовых документов. На втором этапе пользователь формирует задания на печать документов. Информация о заданиях на печать помещается в журнал печати документов.

Каждый документ, помещённый в список готовых документов, описывается следующими реквизитами:

- Наименование документа;
- Идентификатор пользователя – автора документа;
- Метка безопасности документа;
- Форма документа;
- Учётный номер документа;
- Перечень сведений, подлежащих засекречиванию;
- Шаблон документа.

В каждом задании на печать документов системой указывается:

- ссылка (внутренний учётный номер) на документ из списка готовых документов;
- время печати;
- метка безопасности печатного документа;
- количество печатаемых копий;
- идентификатор принтера, на котором производится печать;
- состояние задания на печать.

Задание на печать документа может находится в следующих состояниях:

- ожидание печати;
- идёт печать;
- ошибка печати;
- успешно напечатан.

Для одного документа из списка готовых документов может быть несколько заданий на печать в журнале печати. Пользователь может редактировать список готовых документов, добавляя или удаляя документы, а также изменяя их свойства. В журнал печати пользователь может только добавлять задания. Отменить задание можно только в одном случае – если сервер печати ещё не начал исполнение этого задания (то есть задание на печать находится в состоянии “Ожидание печати”).

Каждый печатаемый документ маркируется штампами в соответствии с шаблоном документа.

Для управления набором шаблонов используется утилита “Редактор шаблонов”, позволяющая добавлять, удалять и модифицировать шаблоны.

Шапмы могут содержать текст, графические изображения, геометрические примитивы и любые другие элементы оформления. В них может выводиться такая информация, как автор документа, его наименование, метка безопасности документа, дата и время печати, прочие атрибуты. Шапмы могут различаться для первой, последней и любых других страниц. Редактируя шаблон, можно изменять положение, содержание и форму печатаемых штампов.

После того, как сформировано задание на печать, отправляется команда сервису печати. Так как в системе может быть несколько физических серверов печати, то команда на печать отдаётся тому из них, к которому подключен принтер, указанный в задании на печать документа.

Сервис печати, обрабатывая очередь заданий по команде менеджера печати,

последовательно выбирает документы, которые находятся в состоянии "Ожидание печати". Каждый такой документ пропускается через последовательность фильтров, подготавливающих документ к отправке на устройство печати. После того, как документ обработан, он отправляется на устройство печати, которым может являться принтер или файл. После завершения печати документа сервис печати вносит изменения в журнал печати. А именно, сохраняет время печати, количество копий, а также меняет состояние задания ("успешно напечатан" или "ошибка печати").

В процессе печати документа возможно возникновение ошибок. Это могут быть ошибки, вызванные отказом оборудования, например, заедание бумаги в принтере, либо это могут быть ошибки обработки документа, например, неприемлемый для принтера шаблон документа. В случае происхождения ошибки на любом из этапов сервис печати переводит задание на печать в состояние "Ошибка" и переходит к следующему заданию. Для повторной печати документа необходимо сформировать новое задание на печать документа.

Любой принтер системы может быть заблокирован по желанию администратора. В этом случае сервис печати перестаёт обрабатывать задания на печать для этого принтера. Задания остаются в состоянии "Ожидание печати" до момента разблокирования принтера.

6. ОПИСАНИЕ МЕХАНИЗМОВ ЗАЩИТЫ

6.1 Описание механизмов идентификации и аутентификации

В таблице 6.1 приведён перечень объектов и способов их идентификации в системе.

Таблица 6.1 – Идентификация объектов в системе

<i>Объект</i>	<i>Способ идентификации</i>	<i>Примечание</i>
Терминал	По MAC-адресу платы сетевого адаптера.	Идентификатор терминала (MAC-адрес) отображается при включении терминала в верхнем правом углу.
Аппаратные узлы сервера	По параметрам процессора, кодам моделей и производителей установленных PCI-устройств, параметрам мостов материнской платы сервера и MAC-адресам сетевых адаптеров.	Аппаратная идентификация производится при загрузке ОС сервера.
Пользователи	По логину или отпечатку пальца (в зависимости от используемой в системе схемы авторизации)	После прохождения авторизации пользователи внутри системы идентифицируются по уникальным числовым идентификаторам (UID), назначаемым пользователям при регистрации их учётных записей. Идентификация по UID пересекается с идентификацией пользователей Linux, что позволяет дополнительно использовать средства защиты информации, предлагаемые операционной системой.

<i>Объект</i>	<i>Способ идентификации</i>	<i>Примечание</i>
Процессы	По уникальному числовому идентификатору процесса в linux (PID).	
Защищаемые файлы и приложения	По абсолютному пути их местоположения в файловой системе.	Внутри системы защищаемые файлы идентифицируются по уникальным числовым идентификаторам (FID), назначаемым файлам при регистрации их учётных записей.
Принтеры	По названию.	Внутри системы принтеры идентифицируются по уникальным числовым идентификаторам, назначаемым принтерам при регистрации их учётных записей.

Способ идентификации и аутентификации пользователей при их входе в систему и разблокировании сеанса зависит от используемой в системе схемы авторизации. Возможные варианты схем перечислены в таблице 6.2.

Таблица 6.2 – Перечень возможных в системе схем авторизации пользователей

<i>Условное обозначение схемы</i>	<i>Способ идентификации</i>	<i>Способ аутентификации</i>
FF	по отпечатку пальца	по отпечатку пальца
LF	по логину	по отпечатку пальца
LP	по логину	по паролю
LPF	по логину	по паролю и отпечатку пальца

Для чтения биометрических данных пользователя (отпечатка пальца) используется манипулятор мышь со встроенным считывателем, подключенный к терминалу пользователя. Идентификация, аутентификация и авторизация пользователя производится централизованно подсистемой контроля доступа на сервере терминального доступа.

В таблице 6.3 приведены параметры биометрической идентификации пользователей.

Таблица 6.3 – Параметры биометрической идентификации

<i>Параметр</i>	<i>Описание</i>	<i>Вероятность</i>
Отказ в регистрации образца	Отказ в регистрации образца возможен в силу природных особенностей папиллярных узоров на пальцах пользователя.	0.03
Ошибочное разрешение доступа	Разрешение доступа от имени другого пользователя из-за схожести их отпечатков.	0.001
Ошибочный запрет доступа	Отказ в доступе в силу того, что система не признала зарегистрированного пользователя.	0.1

6.2 Описание механизма защиты памяти

Условно выделим во всей памяти сервера области, важные в плане защиты информации:

- Часть оперативной памяти, в которой размещаются запущенные на сервере приложения (процессы);
- Часть оперативной памяти, в которой располагается и функционирует ядро операционной системы;
- Память носителей информации, занимаемая файловой системой.

Механизм защиты памяти обеспечивает очистку памяти программы при её завершении, изоляцию памяти работающих программ и очистку памяти занимаемой файлом при его удалении.

6.2.1 Очистка памяти программ

Очистка памяти программы производится как при выделении памяти, так и при её освобождении при завершении программы.

Выделение памяти процессу производится с помощью системного вызова `brk()`, при этом страницы памяти процессу фактически не выделяются, память только резервируется для последующего выделения. Выделение физических страниц памяти производится по первому обращению процесса к зарезервированной памяти, при этом процессор вызывает обработчик прерывания в ядре ОС. При этом завершающим этапом выделения страницы является её обнуление функцией ядра `clear_page()`. Таким образом все страницы памяти, выделяемые процессу будут обнулёнными.

Завершение процесса сопровождается освобождением страниц памяти. Освобождаемые страницы памяти очищаются модулем безопасности. Следует учесть, что не все страницы

памяти завершающегося процесса могут быть освобождены и, следовательно, очищены, так как страницы могут быть разделяемыми:

- с процессом родителя;
- с разделяемой библиотекой;
- с дочерними потомками;
- с любыми другими процессами, использующими механизм разделяемой памяти.

Страницы памяти, которые были доступны процессу только на чтение, очищению также не подлежат.

Модуль безопасности анализирует счётчик использования страницы памяти и флаги страницы. В случае, если страница никем более не используется и доступна на запись, модуль безопасности производит обнуление содержимого страницы перед её освобождением. Таким образом, освобождённая страница памяти не будет содержать данных завершённого приложения.

Разделяемые страницы памяти не обнуляются, но счётчик использования таких страниц уменьшается и страницы будут обнулены при завершении последнего использующего их процесса.

6.2.2 Изоляция памяти программ

Система Mirage функционирует в многозадачной ОС Linux, при этом ядро ОС гарантирует изоляцию адресных пространств программ. Для каждой программы в системе задано её виртуальное адресное пространство. Изоляция адресных пространств реализована за счёт страничной организации памяти и таблиц трансляции виртуальных адресов в физические, которые поддерживаются процессором. Виртуальные адресные пространства разных программ при трансляции их на физические страницы памяти не пересекаются, что делает невозможным прямой доступ одних программ к памяти других. Память, занимаемая ядром Linux, недоступна ни одному из непривилегированных пользователей Linux (то есть никому из пользователей системы Mirage).

6.2.3 Очистка памяти занимаемой файлами

Механизм очистки памяти занимаемой файлами реализован в модуле безопасности и обеспечивает двукратное заполнение маскирующей информацией областей памяти, занимаемых файлом при его удалении. В качестве маскирующего байта используется байт с

попеременно включенными и выключенными битами – 01010101 (ASCII код этого байта соответствует символу "Z"). Маскирование информации гарантирует, что после удаления файла данные, хранимые в нём, не могут быть прочитаны или восстановлены.

6.3 Описание механизма изоляции программ

Изоляция программ в системе достигается посредством защиты областей памяти, занимаемой каждой программой, ядром операционной системы (см. п. 6.2), что делает невозможным прямой доступ одних программ к памяти других программ.

6.4 Описание механизма изоляции файловых систем

Механизм изоляции файловых систем заключается в разделении файловой системы сервера терминального доступа (далее по тексту "корневая ФС") на части – выделенные файловые системы пользователей. Эти части изолированы между собой таким образом, что пользователь, работая в своей изолированной ФС, не имеет доступа ни к корневой ФС, ни к изолированным ФС других пользователей. Этот механизм позволяет пользователям иметь свои конфиденциальные файлы, не доступные другим пользователям системы. Условная схема работы этого механизма представлена на рисунке 6.1.

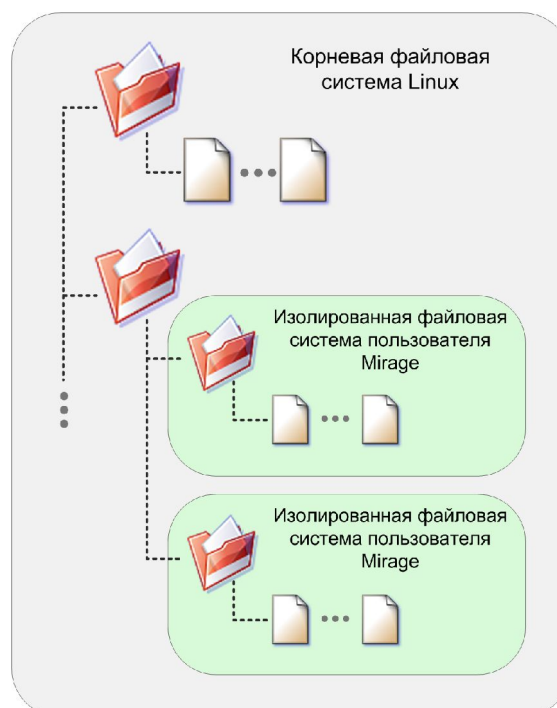


Рисунок 6.1 – Условная схема организации изолированных файловых систем

6.5 Описание средств защиты ввода/вывода на внешние носители

Ввод/вывод информации из АС выполняется на сервере терминального доступа. Ввод либо вывод информации на рабочих станциях невозможен в силу использования терминального решения и программно-аппаратных ограничений, накладываемых на терминал:

- терминал не содержит собственных накопителей информации;
- терминал не содержит устройств вывода информации;
- ОС терминала не поддерживает работу с жёстким диском;
- ОС терминала не поддерживает подключение внешних устройств.

Ввод/вывод подлежит обязательному контролю и выполняется уполномоченным на это пользователем. Факт ввода/вывода информации из АС автоматически отмечается в журнале событий.

7. ОПИСАНИЕ ИНТЕРФЕЙСОВ МОДУЛЕЙ КСЗ

Компоненты КСЗ взаимодействуют друг с другом посредством механизмов межпроцессового и межсистемного взаимодействия:

- 1) сигналы;
- 2) каналы (pipes);
- 3) сокеты.

7.1 Сигналы

Сигналы являются программными прерываниями, которые посылаются процессу, когда случается некоторое событие. Набор возможных сигналов регламентирован ОС Linux, при этом каждый сигнал имеет целочисленное значение и приводит к строго определенным действиям. Источниками сигналов могут выступать модули КСЗ, прикладные задачи, ядро операционной системы. Сигналы подразделяются на системные и прикладные.

Системные сигналы информируют процесс о завершении дочерних процессов, принудительном завершении процесса, а также об ошибках ввода/вывода и ошибках аппаратуры.

Прикладные сигналы используются модулями КСЗ для информирования связанных с ними компонентов о свершении некоторых событий.

7.2 Каналы

Канал обеспечивает однонаправленную связь между двумя процессами. Модули КСЗ используют каналы для обмена данными между потоками многопоточных процессов. Пример использования каналов – передача документа сервисом печати в цепочку дочерних процессов, маркирующих и отправляющих документ на принтер.

7.3 Сокеты

Сокеты обеспечивают двухстороннюю связь типа "точка-точка" между двумя процессами. Они являются основными компонентами межсистемной и межпроцессовой связи. Каждый сокет представляет собой конечную точку связи, с которой может быть связано некоторое имя. Он также имеет определенный тип и один или нескольких связанных с ним

процессов.

Модули КСЗ используют локальные unix-сокеты для взаимодействия друг с другом в пределах одной вычислительной системы. Для примера: обмен через unix-сокеты используют все компоненты КСЗ (сервисы и прикладные задачи) для взаимодействия с подсистемой регистрации событий.

Для взаимодействия с терминалами, а также компонентами функционирующими на других серверах используются датаграммные и потоковые Интернет-сокеты. Пример: для загрузки образа операционной системы на терминал используется обмен через датаграммные сокеты (протокол UDP). Обмен данными между терминалом и сервером в рамках терминального сеанса осуществляется с помощью потоковых Интернет-сокетов (протокол TCP).

ПРИЛОЖЕНИЕ 1

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

- АС – Автоматизированная система.
- БД – База данных.
- ДСП – Для служебного доступа.
- КСЗ – Комплекс средств защиты.
- НДВ – Недекларированные возможности.
- НС – Несекретно.
- НСД – Несанкционированный доступ.
- ОС – Операционная система.
- ППС – Пункт перечень сведений, подлежащих засекречиванию.
- ПО – Программное обеспечение.
- ПРД – Правила разграничения доступа.
- С – Секретно.
- СВТ – Средства вычислительной техники.
- СС – Совершенно секретно.
- СУБД – Система управления базами данных.
- ТВС – Терминальная вычислительная система.
- УН – Учётный номер.
- ФС – Файловая система.

ПРИЛОЖЕНИЕ 2

ПЕРЕЧЕНЬ ТЕРМИНОВ

Авторизация. Проверка права входа пользователя в систему (предоставление пользователю полномочий на вход и работу в системе).

Администратор безопасности (защиты). Субъект доступа, ответственный за защиту системы от несанкционированного доступа к информации.

Администратор системы. Субъект доступа, ответственный за администрирование системы.

Аутентификация. Проверка принадлежности субъекту доступа предъявленного им идентификатора; подтверждение подлинности.

Безопасность информации. Состояние защищенности информации, обрабатываемой средствами системы, от внутренних или внешних угроз.

Дискреционный контроль доступа. Разграничение доступа между поименованными субъектами доступа и поименованными объектами доступа.

Журнал печати. Таблица служебной БД, содержащая множество записей о заданиях на печать.

Журнал событий. Таблица служебной БД, содержащая множество записей о всех произошедших в системе событиях.

Задание на печать. Запись в журнале печати, соответствующая одному запросу на печать документа.

Идентификация. Присвоение субъектам и объектам доступа идентификатора и (или) сравнение предъявляемого идентификатора с перечнем зарегистрированных в системе идентификаторов.

Конфиденциальная информация. Информация, требующая защиты.

Мандатный контроль доступа. Разграничение доступа субъектов доступа к объектам доступа, основанное на характеризуемой меткой конфиденциальности (безопасности) информации, содержащейся в объектах, и официальном разрешении (уровне допуска) субъектов обращаться к информации такого уровня конфиденциальности.

Матрица доступа. Таблица, содержащая дискреционные права доступа субъектов доступа к объектам доступа.

Метка безопасности (конфиденциальности). Элемент информации, который характеризует конфиденциальность информации, содержащейся в объекте.

Несанкционированный доступ к информации (НСД). Доступ к информации, нарушающий правила разграничения доступа с использованием штатных средств, предоставляемых средствами вычислительной техники или автоматизированной системой.

Объект доступа. Единица информационного ресурса автоматизированной системы, доступ к которой регламентируется правилами разграничения доступа.

Очередь печати. Список заданий на печать, находящихся в состоянии 'ожидание печати', организуемый подсистемой печати при выводе графических документов на твёрдую копию.

Рабочая среда. То же, что и Сеанс

Пароль. Идентификатор субъекта доступа, который является его (субъекта) секретом.

Печатный (графический) документ. Специально подготовленный для печати образ определённого формата, получаемый на выходе процедуры печати прикладного приложения. Документ может содержать текстовую и графическую информацию.

Пользователь. Субъект доступа, обладающий учётными реквизитами, определённым уровнем полномочий, определяющим его доступ к объектам системы.

Приложение. Прикладная программа, расположенная на носителях сервера терминального доступа и эксплуатируемая пользователями терминально.

Сеанс. Совокупность запущенных авторизованным пользователем программ на сервере терминального доступа.

Сервер печати. Функциональная часть системы, обеспечивающая процесс печати документов (вывода графических документов на твёрдую копию).

Сервер терминального доступа (терминальный сервер). Функциональная часть системы, обеспечивающая терминальную работу авторизованных пользователей.

Сервисы системы. Специализированные программы, запускаемые на функциональных частях системы, выполняющие служебные функции. Запуск этих программ осуществляется при загрузке операционной системы, а остановка – при её (операционной системы) завершении.

Сессия. Период времени, в течение которого пользователь работает в своем сеансе в терминальном режиме.

Система Mirage. Программно-аппаратный комплекс, автоматизированная система обработки информации, совокупность взаимосвязанных функциональных частей, методов и средств защиты информации, автоматизации её обработки, контроля её целостности и передачи по каналам связи.

Служебная БД. Функциональная часть системы, обеспечивающая процесс хранения служебной информации (учётных записей, матрицы доступа и т.д.).

Событие. Факт выполнения какой-либо операции в системе, смены состояния статусов её объектов и субъектов, возникшей ошибки или отклонения работы системы от нормального режима эксплуатации

Субъект доступа. Лицо, действия которого регламентируются правилами разграничения доступа.

Терминал. Устройство ввода/вывода информации с ограниченным набором периферийного оборудования, не имеющее возможностей по обработке и хранению данных, подключению внешних устройств. Используется только для отображения видеoinформации, поступающей с сервера терминального доступа, и передачи на сервер терминального доступа команд, принимаемых с устройств ввода (клавиатура и мышь).

Уровень допуска. Выданное системой субъекту доступа разрешение обращаться к объектом доступа определённого уровня конфиденциальности.

Уровень полномочий. Совокупность всех прав доступа субъекта доступа.

Целостность информации. Способность средства вычислительной техники или автоматизированной системы обеспечивать неизменность информации в условиях случайного и (или) преднамеренного искажения (разрушения).

Шаблон документа. Программа на языке postscript, на основе которой сервер печати маркирует документы при их печати соответствующими штампами. Шаблоны создаются и модифицируются в редакторе шаблонов.

Штамп. Графический объект (или набор объектов) наносимый на печатные документы в момент их вывода на твёрдую копию. (рамки, защитная графика, информация о

пользователе, дате, количестве копий, прочее).