Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЁТ**

**По учебной практике ‘Управление учетными записями пользователей’**

Студент: Райлян Арсений Рооманович

Дисциплина/Профессиональный модуль: Учебная практика

Выполнил студент

Группы: 2ОИБАС-1322

Преподаватель

Сибирев И.В.

Оценка за работу :\_\_\_\_\_\_\_

**Москва – 2024г.**

**Задание 1**

**Часть первая**

1. Установил винду
2. Для перехода в режим настройки и редактирования учетных записей необходимо в панели управления выбрать категорию Учетные записи пользователей, и в этой категории щелкнуть на ссылке Добавление и изменение учетных записей пользователей.

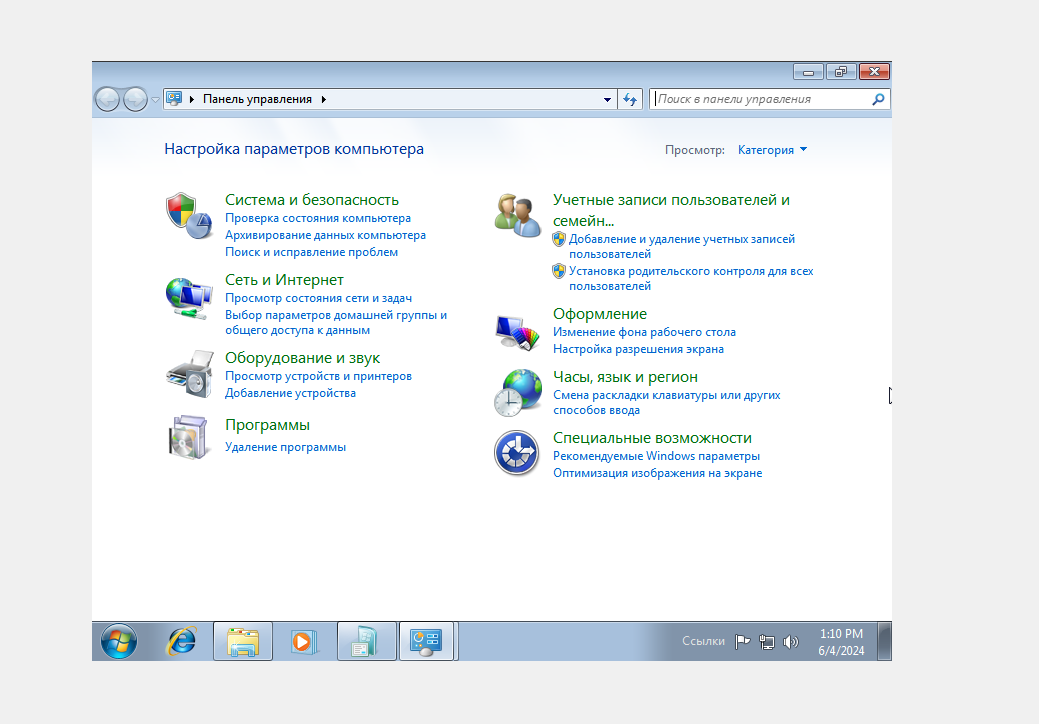
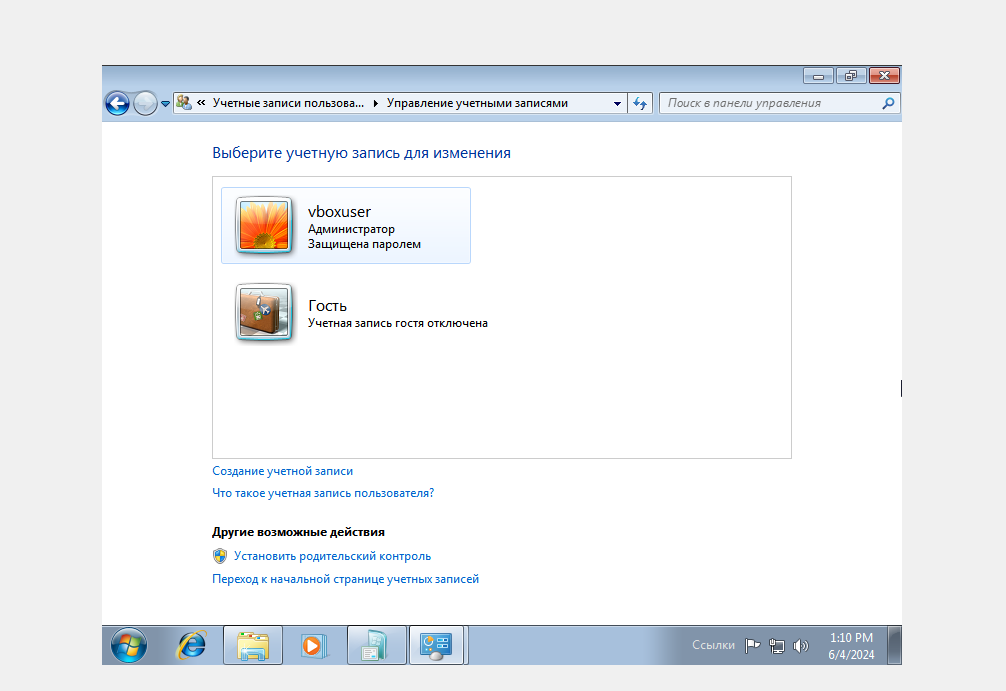
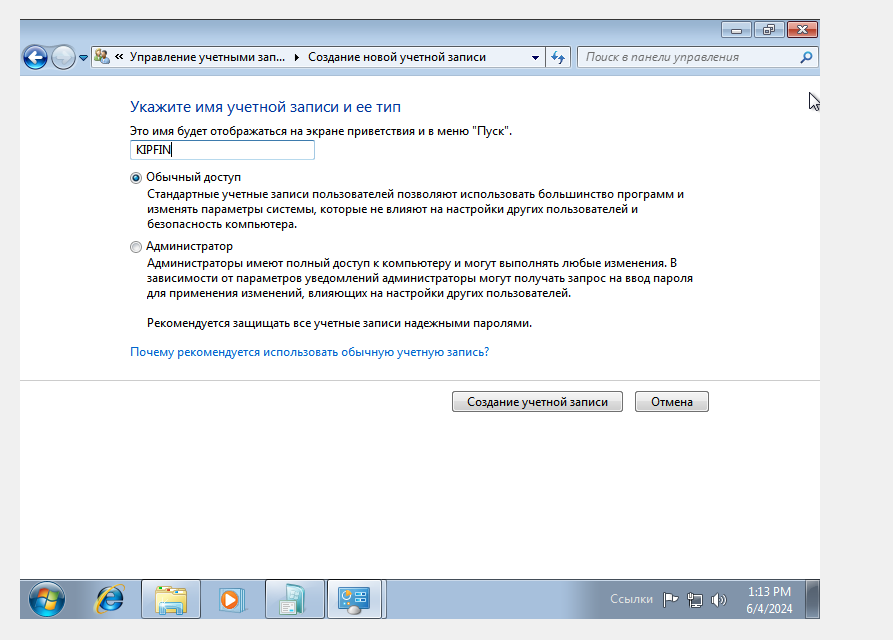


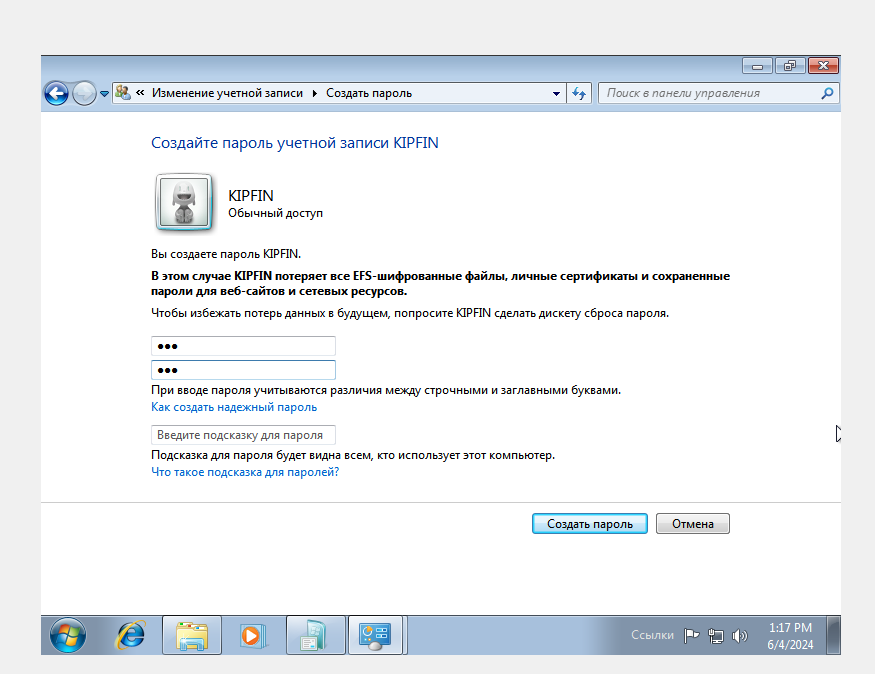
Рисунок 1



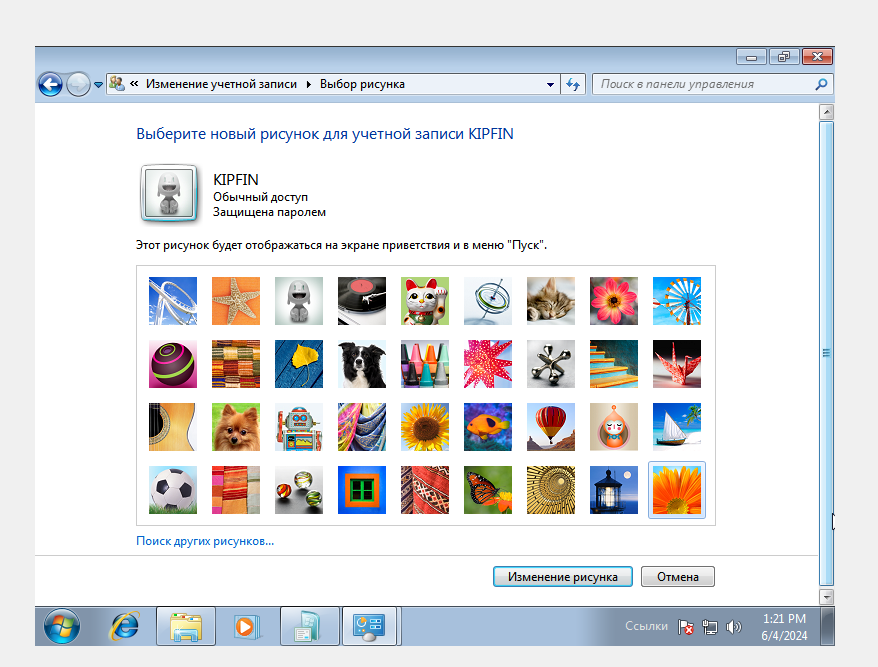
1. Далее создаем новую учебную запись и ставим имя KIPFIN



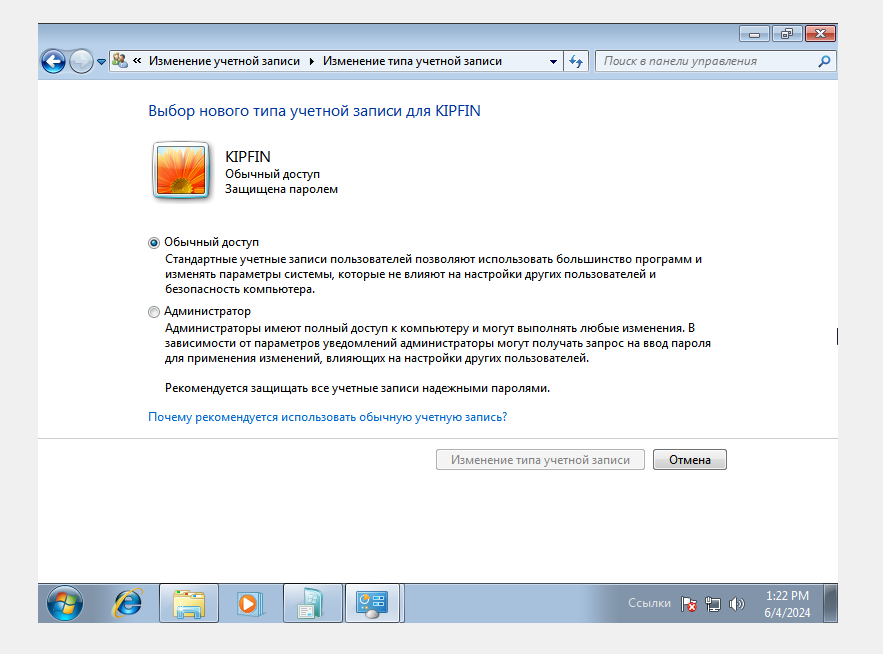
1. Изменяем нашу учетную запись, меняем пароль, ставим пароль 123



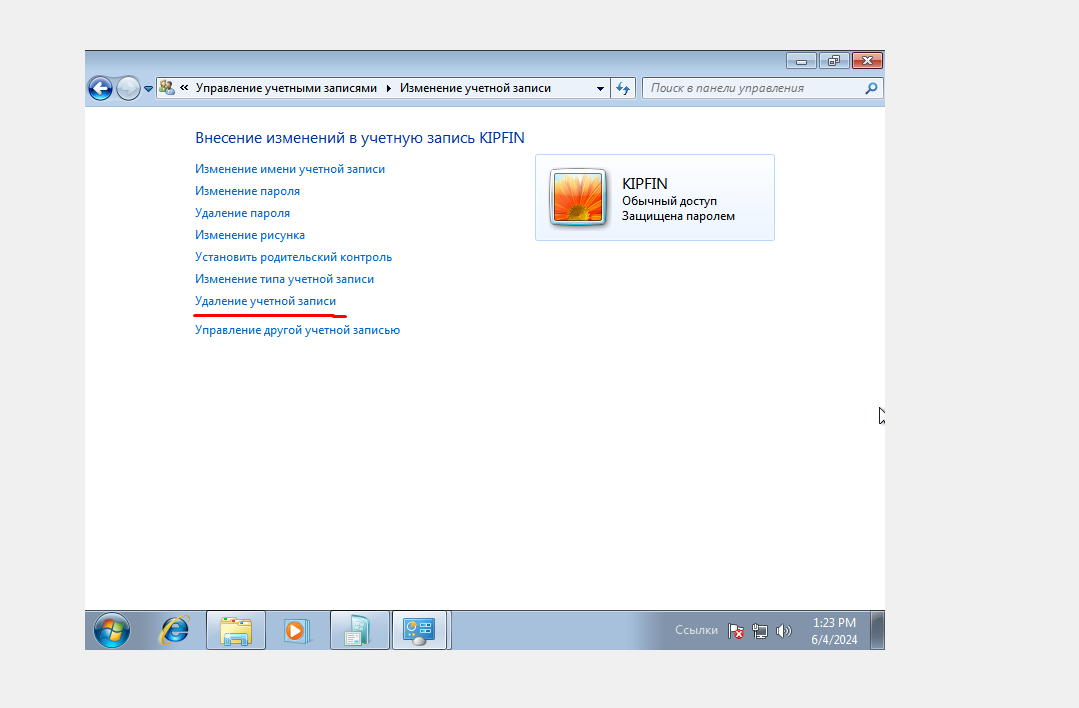
1. Изменяем аватарку нашего пользователя



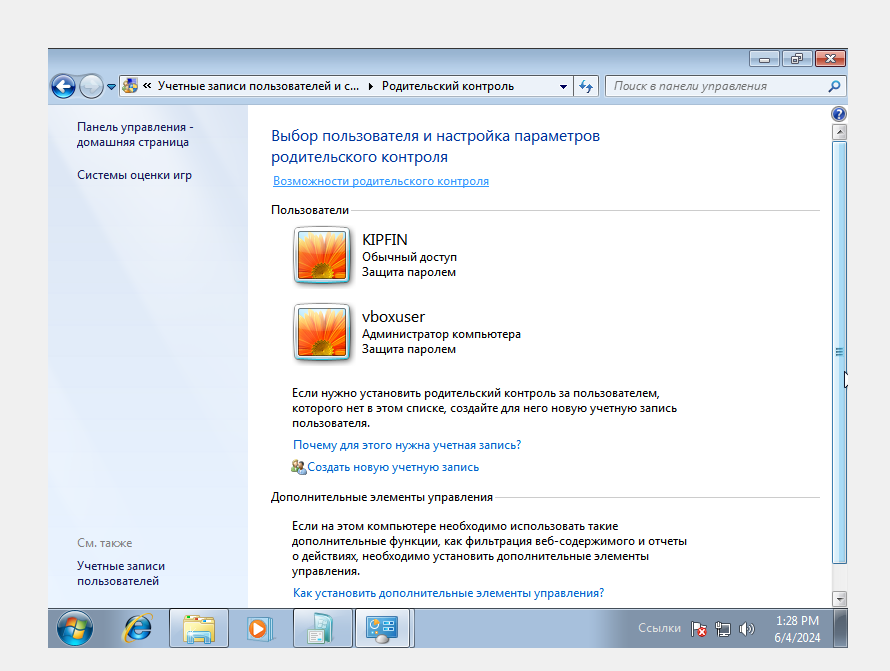
1. Далее можем изменить тип учетной записи, но я не буду этого делать



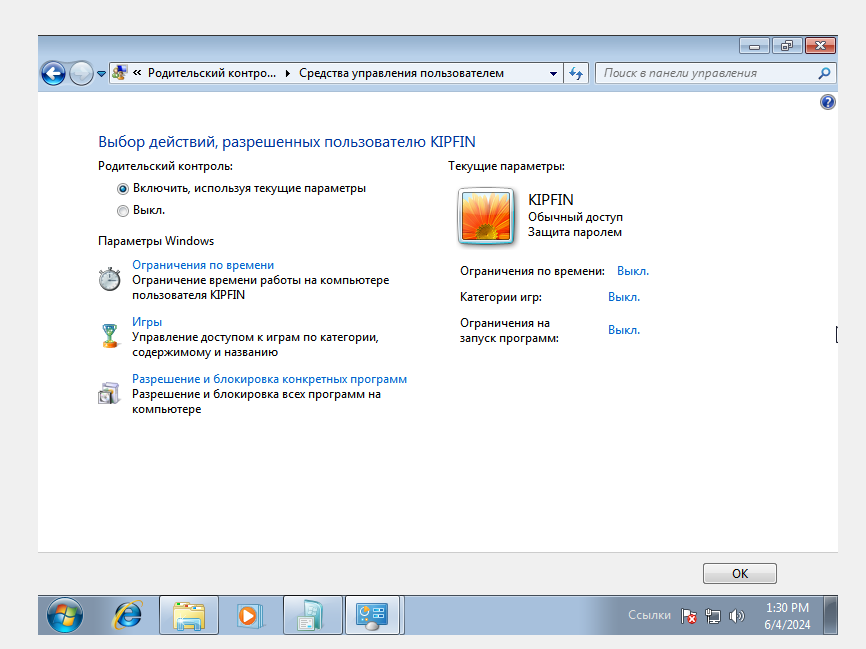
1. Далее можем удалить учетную запись, но мы этого не будем делать



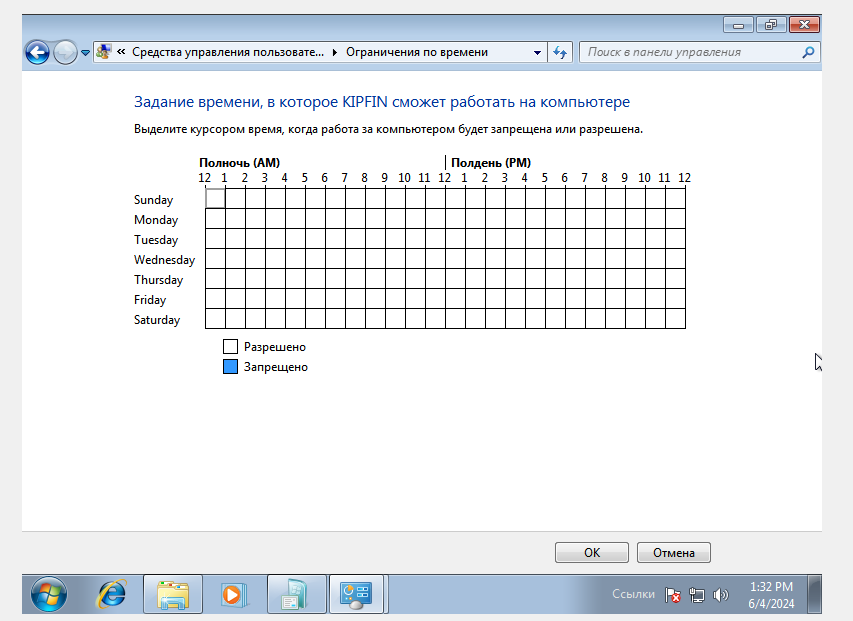
1. Далее переходим в родительский контроль и выбираем учетную запись KIPFIN



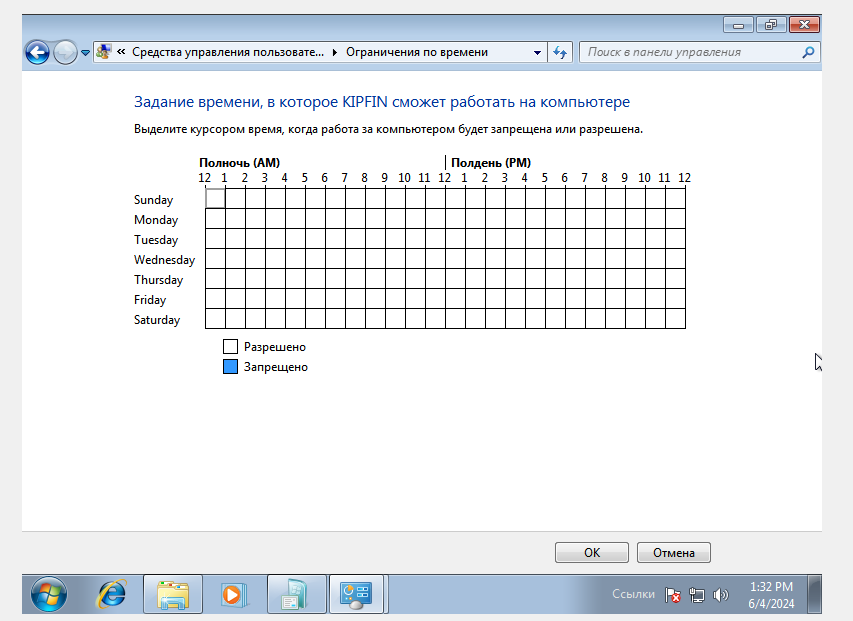
1. И Включаем родительский контроль



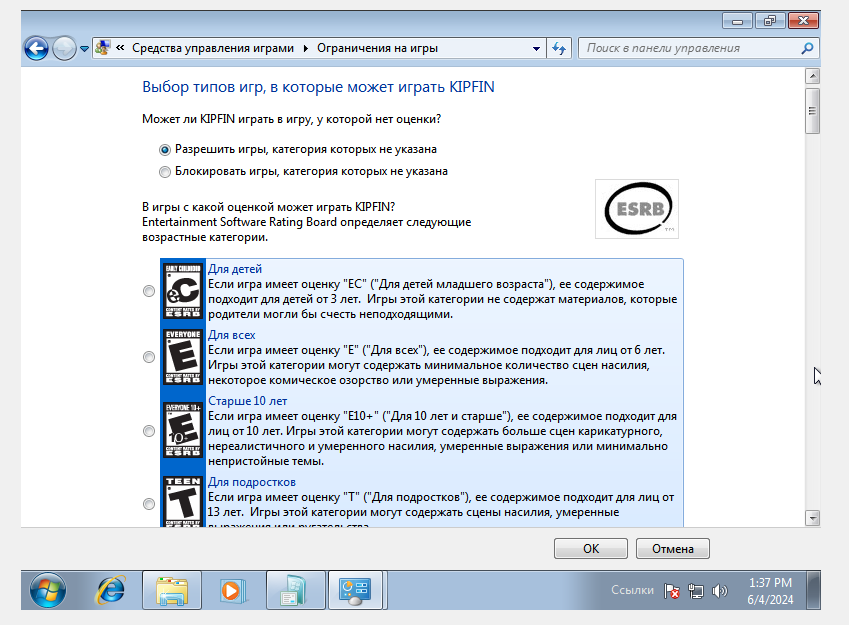
1. Далее мы можем ограничить по времени использования компьютера, но я этого не буду делать



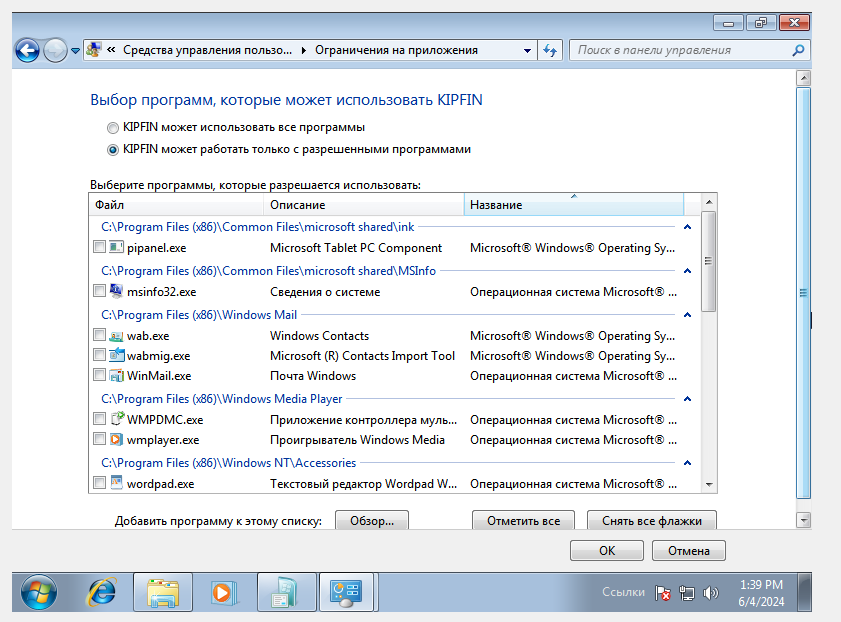
1. Далее можем ограничить пользованием игр



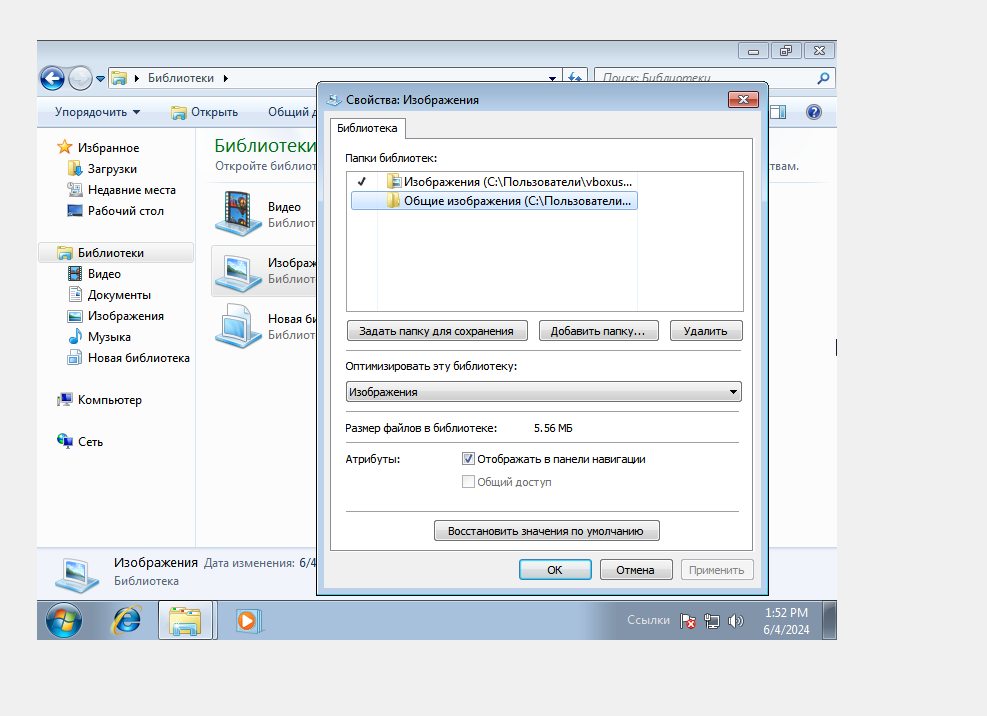
1. Можем задать категорию игр



1. Так же мы можем выбрать программы, которые можно использовать, а какие нет



1. Я не смог настроить безопасность файла, ибо когда пытаешься открыть свойства файла, там нет иконки безопасность.



**Часть вторая**

**Квоты диска и настройки квот для сетевых папок в Windows Server 2016**

В Windows Server 2016 можно использовать квоты диска для управления объемом используемого дискового пространства для пользователей и групп. Квоты позволяют ограничивать объем данных, который пользователь или группа может хранить на диске.

Чтобы настроить квоты диска для сетевых папок в Windows Server 2016, выполните следующие шаги:

1. Откройте "Диспетчер сервера" (Server Manager) на сервере.

2. В меню "Управление" выберите "Управление файловыми службами" (File and Storage Services).

3. Выберите "Роли файловых служб" (File and iSCSI Services).

4. Далее выберите "Серверные квоты" (File Server Resource Manager).

5. В разделе "Действия" (Actions) выберите "Настроить квоты" (Configure Quotas).

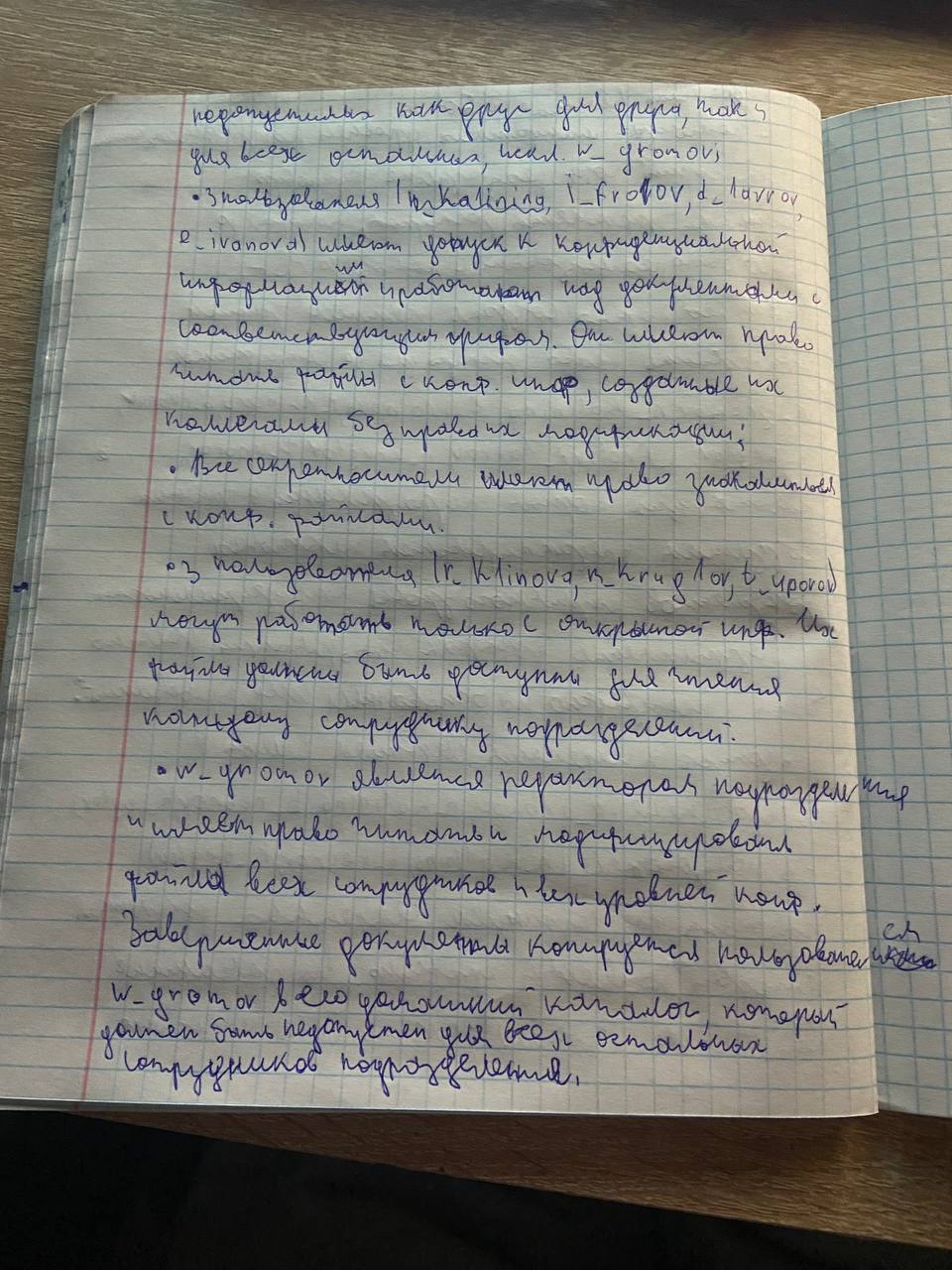
6. Выберите том, на котором находится ваша сетевая папка, и нажмите "Создать квоту" (Create Quota).

7. Укажите параметры квоты, такие как лимиты для пользователей или групп, уведомления и предупреждения о превышении лимита.

8. После завершения настройки квоты нажмите "ОК" для сохранения изменений.

Теперь квоты диска будут применены к вашей сетевой папке, и пользователи или группы будут ограничены в использовании дискового пространства в соответствии с настройками квот.

**Задание 2**



**Задание 3**

**Цель лабораторной работы**

Изучить и освоить администрирование основных параметров мандатного управления доступом в ОССН Astra Linux Special Edition с применением графических утилит и консольных команд.

**Используемое программное обеспечение**

Для выполнения лабораторной работы используется установленный дистрибутив ОС Astra Linux  Smolensk.

**Задание на лабораторную работу**

В ходе лабораторной работы необходимо произвести установку и настройку пакетов OpenLDAP, Kerberos, NSS, PAM, NFS, SMB, NMBD.

***Практическая часть***

Отключили блокировку экрана.

Создали учётную запись user с паролем Student123. Добавили учётную запись user во вторичную группу astra-admin (рис 1)

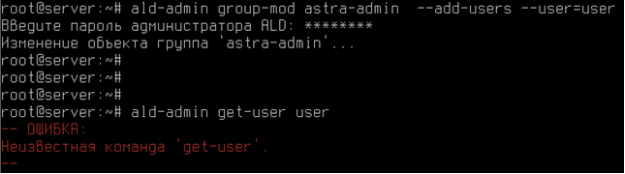


Рисунок 1

Начали работу со входа в ОССН в графическом режиме с учётной записью пользователей “Управление политикой безопасности” через команду sudo fly-admin-smc (рис 2)

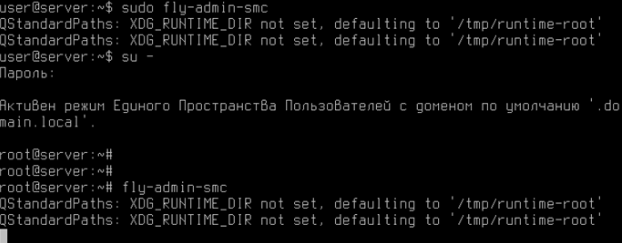


Рисунок 2

Модифицировали параметры мандатного управления доступом (рис 3)

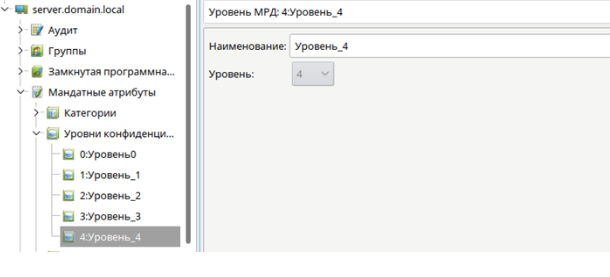


Рисунок 3

Создали учётную запись пользователя user1, установив максимальный уровень доступа (рис 4)

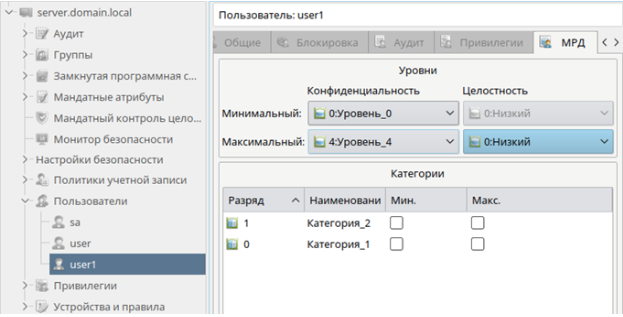


Рисунок 4

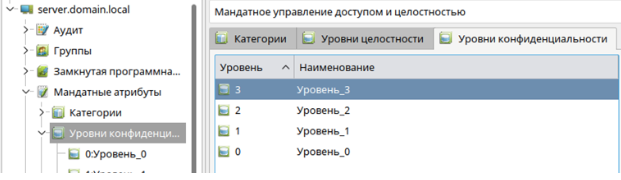


Рисунок 5

Открыть учётную запись пользователя user1 и в закладке «Дополнительные» в элементе «Максимальный уровень» проверить наличие записи «(4)», при этом, в списке выбора уровня «Уровень\_4» будет отсутствовать. Создать учетные записи user20, user30, user40. Задать в произвольном порядке значения для параметров «Конфиденциальность», «Целостность», «Категории».

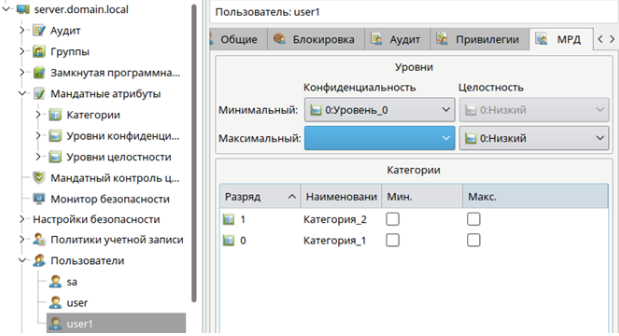


Рисунок 6

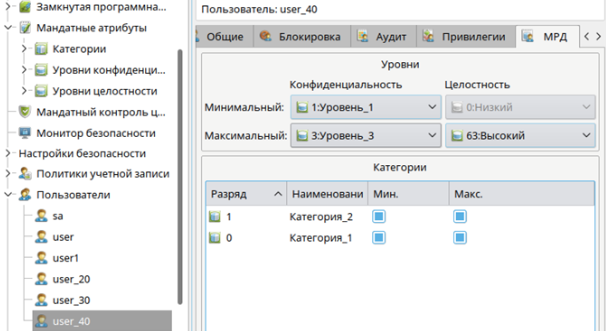


Рисунок 7

Ознакомится с содержимым каталога etc/parsec/macdb. Вывести в терминал Fly параметры мандатного управления доступом для учётной записи пользователей user1, user20, user30, user40. Мандатные атрибуты, назначенные конкретным учётным записям пользователей, перечисляются в каталоге /etc/parsec/macdb. Для выполнения задания 7 осуществить следующие действия:

* 1. **запустить терминал Fly и перейти в каталог /etc/parsec/macdb (команда cd /etc/parsec/macdb )**
  2. **вывести на экран содержимое каталога /macdb ( ls )**
  3. **просмотреть содержимое файлов в каталоге /macdb ( sudo cat “имя файла” )**
  4. **прочитать параметры учётной записи user1 командой sudo grep “user1” \***
  5. **определить максимальный уровень доступа учётной записи user1 командой sudo grep «user1» \* | cut -d : -f 5**
  6. **определить минимальный уровень доступа учётной записи user1 командой sudo grep «user1» \* | cut -d : -f 3 и проверить его соответствие данным, отображаемым в графической утилите «Управление политикой безопасности».**

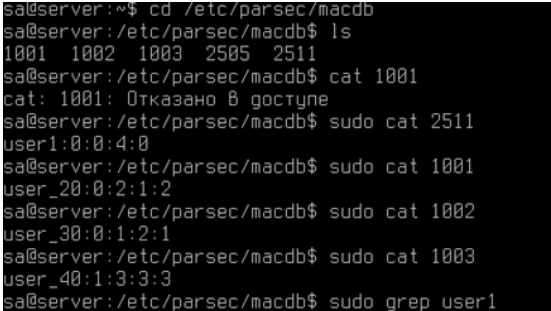


Рисунок 8

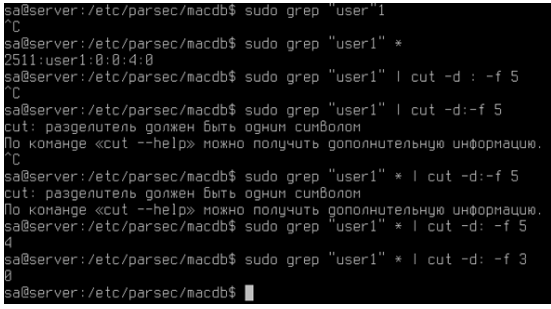


Рисунок 9

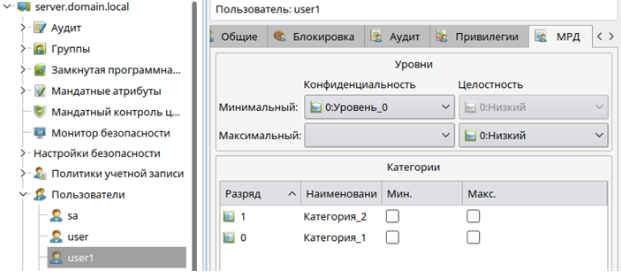


Рисунок 10

Создать неиерархические категории с использованием графической утилиты «Управление политикой безопасности». Для этого выполнить следующие действия:

* 1. в разделе «Категории» удалить все Категории.
  2. в разделе «Категории» создать новую неиерархическую категорию с именем «Otdel1», «Разряд» — 0;
  3. в разделе «Категории» создать новые неиерархические категории: «Otdel2» («Разряд» — 1), «Upravlenie» («Разряд» — 2).

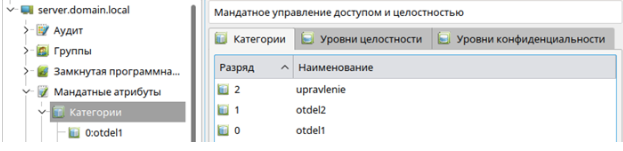


Рисунок 11

Изменить набор неиерархических категорий с использованием графической утилиты «Управление политикой безопасности» (sudo fly-admin-smc), для этого выполнить следующие действия в разделе «Категории»:

* 1. **выбрать неиерархическую категорию «Otdel1» и ввести наименование «Отдел\_1»;**
  2. **аналогично переименовать неиерархические категории «Otdel2» и «Upravlenie» в «Отдел\_2» и «Управление» соответственно;**
  3. **проанализировать возможность одновременного изменения элемента «Разряд».**

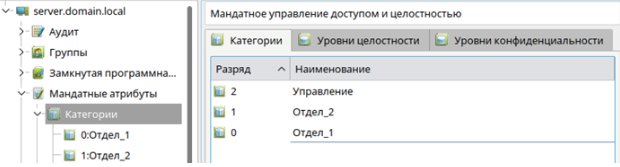


Рисунок 12

Изменить мандатный уровень доступа с использованием графической утилиты «Управление политикой безопасности», для этого выполнить следующие действия:

* 1. создать новую группу с именем «office1» и задать первичную группу учётной записи пользователя user1 — «office1»;
  2. создать новую учетную запись пользователя user2 и установить её первичную группу — «officel»;
  3. в закладке «МРД» осуществить попытку выбора минимального набора неиерархических категорий — «Отдел\_2» и проанализировать результат;
  4. в закладке «МРД» выбрать максимальный уровень доступа — «Уровень\_3», максимальный набор неиерархических категорий — «Отдел\_2», после чего задать минимальный набор неиерархических категорий — «Отдел\_2»;
  5. открыть параметры учётной записи пользователя user1 и выбрать максимальный уровень доступа — «Уровень\_3», максимальный набор неиерархических категорий — «Отдел\_1», минимальный набор неиерархических категорий — «Отдел\_1»;.
  6. создать учётную запись пользователя rukoffice1 и задать первичную группу: «office1» ;
  7. в закладке «МРД» выбрать максимальный уровень: «Уровень\_3», максимальный набор категорий: «Отдел\_1», «Отдел\_2», «Управление».

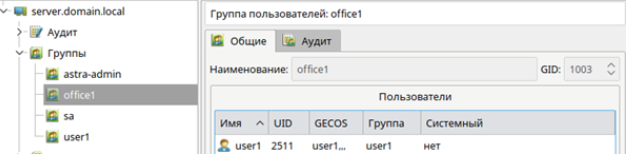


Рисунок 13

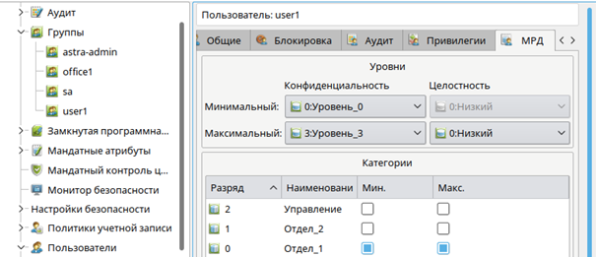


Рисунок 14

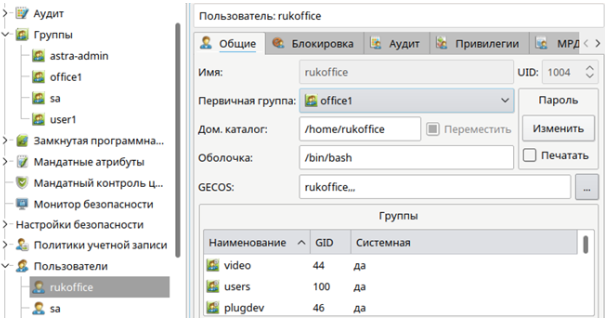


Рисунок 15

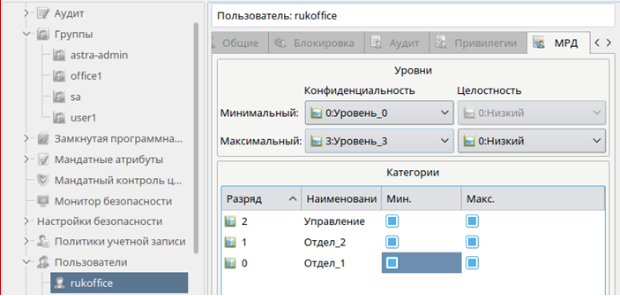


Рисунок 16

Создать общий каталог для работы от имени учётных записей пользователей userl, user2, rukoffice1 в каталоге /home/work. Для работы от имени учётных записей пользователей с наборами неиерархическими категорий «Отдел\_1», «Отдел\_2» и «Управление» выделить отдельные каталоги «otdel1», «otdel2» и «upr» соответственно.

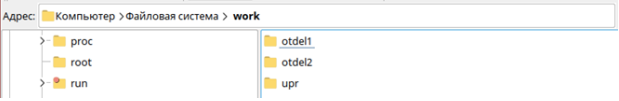


Рисунок 17

При этом обеспечить хранение файлов с различными уровнями конфиденциальности в каталогах с использованием специального атрибута CCNR, для чего осуществить следующие действия:

* 1. запустить терминал Fly в «привилегированном» режиме командой sudo fly-term;
  2. Прочесть информацию по командам mkdir, chown и chmode в приложении для практической работы.
  3. создать каталог work и задать параметры мандатного и дискреционного управления доступом командами:

*mkdir /home/work*

*chown user /home/work -v (назначение user владельцем каталога)*

*ls –lh (посмотрите изменения параметра “владелец” для каталога work)*

*chown :office1 /home/work –v (назначение группы)*

*ls –lh (посмотрите изменения параметра “группа” для каталога work)*

*chmod 750 /home/work*

*pdp-flbl 3:0:Отдел\_1,Отдел\_2,Управление:ccnr /home/work –v*

* 1. создать каталог для работы от имени учётных записей пользователей с набором неиерархических категорий «Отдел\_1» и задать параметры мандатного и дискреционного управления доступом командами:

*cd /home/work*

*mkdir otdel1*

*chown user1:office1 otdel1*

*chmod 770 otdel1*

*pdp-flbl 3:0:Отдел\_1:ccnr otdel1*

* 1. создать каталог для работы от имени учётных записей пользователей с набором неиерархических категорий «Отдел\_2» и задать параметры мандатного и дискреционного управления доступом командами:

*mkdir otdel2*

*chown user2:office1 otdel2*

*chmod 770 otdel2*

*pdp-flbl 3:0:Отдел\_2:ccnr otdel2*

* 1. создать каталог upr для работы от имени учётных записей пользователей с набором неиерархических категорий «Управление» командами:

*mkdir upr*

*chown rukoffice1:office1 upr*

*chmod 770 upr*

*pdp-flbl 3:0:Управление:ccnr upr*

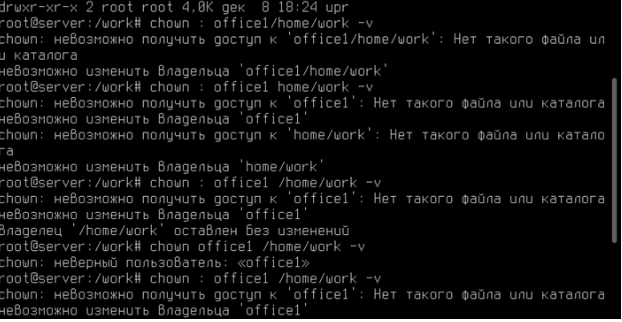


Рисунок 18

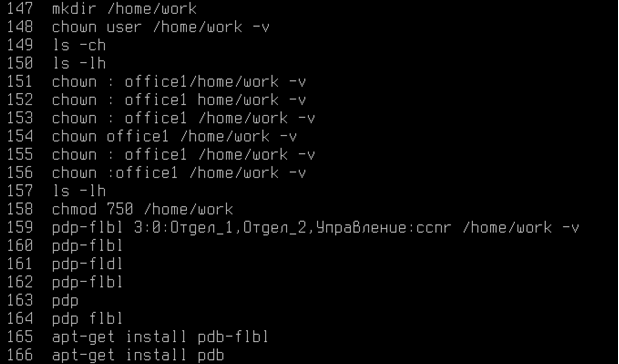


Рисунок 19

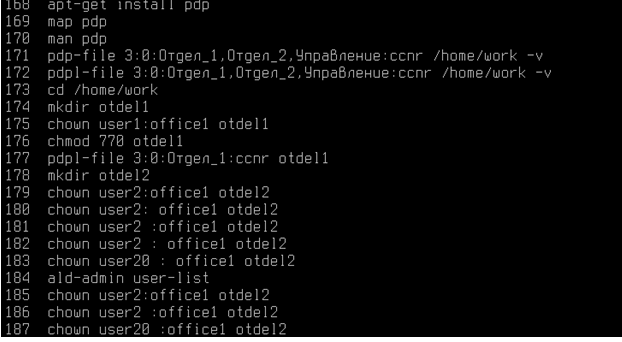


Рисунок 20

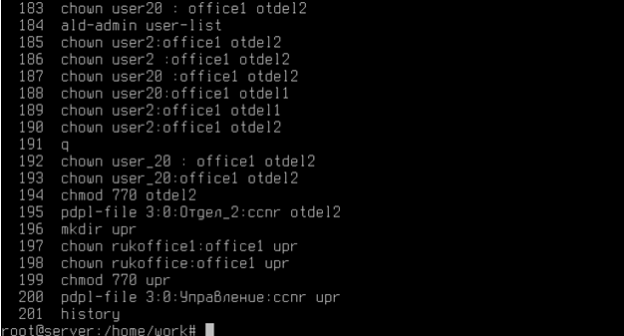


Рисунок 21

Задание 4.

Мандатный контроль целостности (Mandatory Access Control, MAC) в операционной системе Astra Linux представляет собой механизм, который обеспечивает контроль доступа к ресурсам на основе уровня секретности и целостности. В контексте Astra Linux мандатный контроль целостности осуществляется с помощью специальной подсистемы, называемой LSM (Linux Security Modules), которая позволяет определять и применять политики безопасности на уровне ядра операционной системы. Основные принципы мандатного контроля целостности Astra Linux: 1. Метки безопасности: Astra Linux использует метки безопасности для классификации объектов (файлов, процессов, ресурсов) и субъектов (пользователей, приложений) в соответствии с установленными правилами безопасности. 2. Правила доступа: Администраторы могут определять правила доступа на основе меток безопасности, задавая, кто и в каких условиях имеет право на чтение, запись или выполнение определенных ресурсов. 3. Целостность данных: Мандатный контроль целостности в Astra Linux обеспечивает защиту от несанкционированных изменений данных путем контроля и ограничения доступа к ним на основе уровня целостности. 4. Аудит: Astra Linux предоставляет возможность аудита действий пользователей и приложений, что позволяет отслеживать и анализировать события, связанные с доступом к ресурсам. Применение мандатного контроля целостности в Astra Linux способствует обеспечению высокого уровня безопасности операционной системы путем строгого контроля доступа к ресурсам на основе установленных правил безопасности и меток безопасности.

**Задание 5**

Лабораторная работа №8 «Настройка механизма замкнутой программной среды» В лабораторной работе продемонстрирована организация замкнутой программной среды для пользователей на примере "user1" и "user2". Пользователям на компьютере организуется запуск ограниченного набора программ:

1) Проводник;

2) LibreOffice Writer;

3) LibreOffice Calc;

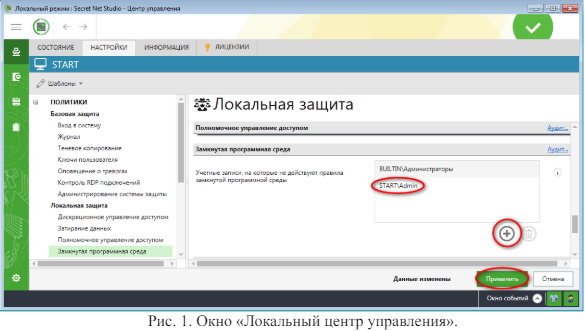
4) Internet Explorer;

5) Корзина.

Система Secret Net Studio позволяет сформировать модель данных для механизма замкнутой программной среды на основе сведений о запускавшихся программах из своего журнала безопасности. На основании этих данных формируются задания замкнутой программной среды для субъектов.

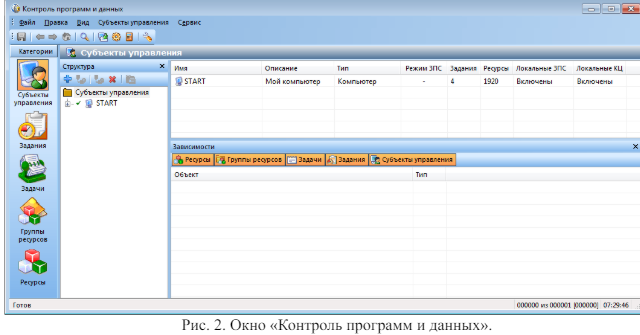
ХОД РАБОТЫ

1. Откройте меню настроек "Замкнутая программная среда" («Пуск → Все программы → Код безопасности → Secret Net Studio→ Локальный центр управления») и откройте раздел «НАСТРОЙКИ → Замкнутая программная среда» ". Добавьте администратора в группу привилегированных пользователей.

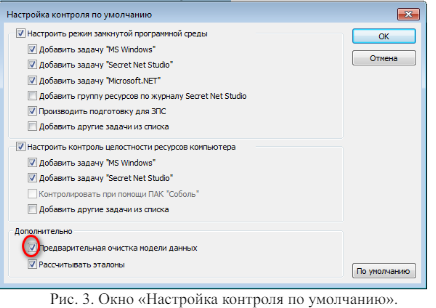


Нажмите кнопку «Применить».

2. Настройка механизма ЗПС осуществляется в программе "Контроль программ и данных". Запустить программу "Контроль программ и данных" можно через "Пуск → Все программы → Код Безопасности → Secret Net Studio→ Контроль программ и данных".

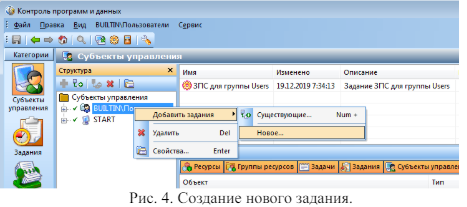


1. В меню "Файл" выберите пункт "Новая модель данных". На экране появится диалог "Настройка контроля по умолчанию". Добавьте выполнение пункта "Предварительная очистка модели данных" и нажмите кнопку "ОК".

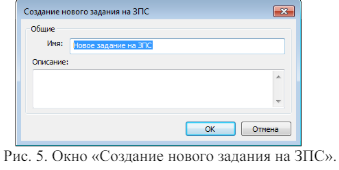


4. Программа запустит подготовку ресурсов для использования в ЗПС. По окончании автоматически запустится расчет эталонов для ресурсов. Дождитесь окончания процедуры расчета.

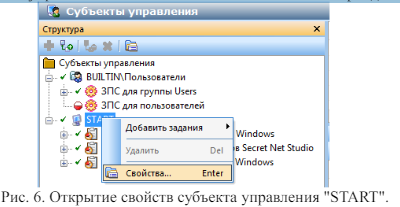
5. Создайте новое задание. Для этого, используя контекстное меню в поле "Субъекты управления", выберите "Добавить задание → Новое".



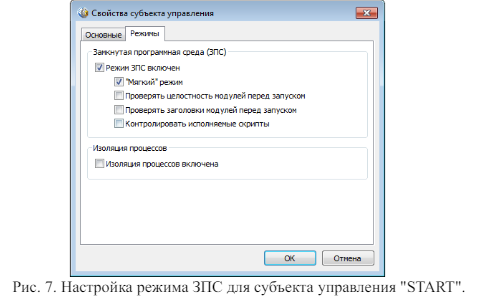
6. Введите название ЗПС для пользователей и нажмите кнопку "ОК".



7. Выберите субъект управления "START" и в контекстном меню перейдите в "Свойства".



8. Включите "Мягкий режим" ЗПС и нажмите кнопку "ОК".



9. Сохраните модель и перезагрузите компьютер.

10.Выполните вход в систему под учетной записью администратора и откройте журнал Secret Net Studio.

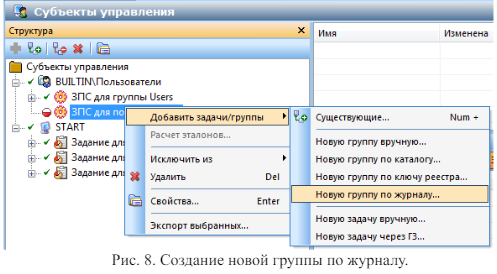
11.Выполните экспорт журнала во внешний файл с удалением после экспорта.

12.Перезагрузите компьютер и выполните вход в систему под учетной записью «user1».

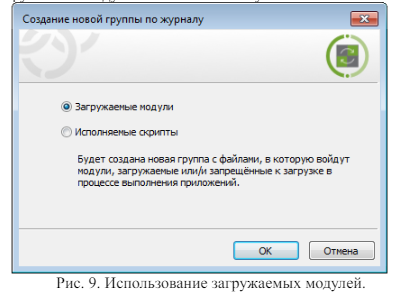
13.Запустите все программы, которые будут разрешены в дальнейшем пользователям.

14.Завершите сеанс и войдите в систему под учетной записью администратора.

15.Откройте программу "Контроль программ и данных". Для задания "ЗПС для пользователей" в контекстном меню выберите "Добавить задачи/группы →Новую группу по журналу".



16.Выберите загружаемые модули и нажмите кнопку "ОК".

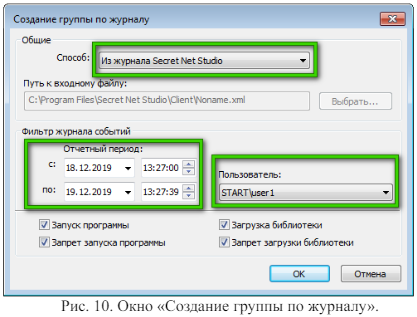


17.Выполните необходимые настройки в окне "Создание группы по журналу":

1) в поле способ укажите: "Из журнала Secret Net Studio";

2) в поле "Пользователь" нажмите кнопку "Найти" и выберите "user1";

3) в поле "Отчетный период" укажите период запуска разрешенных программ для "user1". Поскольку в данной лабораторной работе журнал Secret Net Studio был предварительно очищен, то поле "Отчетный период" можно оставить без изменения.



18.Нажмите кнопку "ОК" и сохраните модель.

19.Включите режим ЗПС. Снимите галочку в поле "Мягкий режим". Нажмите кнопку "OK" и сохраните модель данных (аналогично пунктам 7-8).

20.Войдите в систему под учетной записью пользователя "user1" и убедитесь, что пользователь может запускать только ограниченный набор программ:

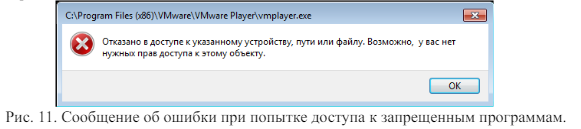
6) Проводник;

7) LibreOffice Writer;

8) LibreOffice Calc;

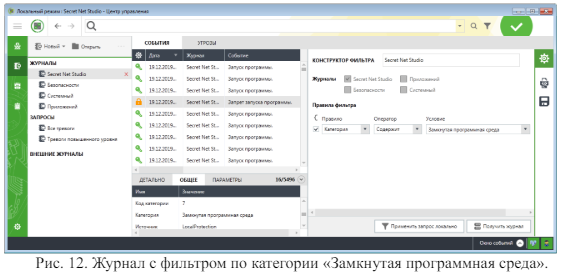
9) Internet Explorer;

10) Корзина.



21.Войдите в систему под учетной записью пользователя "user2" и убедитесь, что замкнутая программная среда работает и для других пользователей.

22.Войдите в систему под учетной записью администратора и откройте журнал Secret Net Studio. Ознакомьтесь с записями журналами, имеющими категорию "Замкнутая программная среда".



23.Отключите механизм ЗПС. Откройте программу "Контроль программ и данных" и снимите галочку в поле "Режим ЗПС включен". Закройте окно программы и перезагрузите компьютер.

**Задание 7**

# Лабораторная работа 1.3. Установка и настройка сервера *DNS*

**Цель:**научиться устанавливать сервер имён, добавлять зоны расширения имён, включать автоматическое обновление зон.

**Средства для выполнения работы:**

*∙***аппаратные**: компьютер с установленной ОС ***Windows XP***;

*∙***программные**: приложение ВМ: ***VirtualBox***; виртуальные машины: ***VM-1, VM-2***; установочные образы ОС: ***win98.iso.***

***Теоретические сведения***

*Система доменных имен (****DNS****)*была исходно определена в документах ***RFC*(*R****equest****F****or****C****omments***) *1034 и 1035***. Эти документы определяют следующие элементы, общие для всех реализаций программного обеспечения ***DNS***:

*∙*пространство доменных имен ***DNS***, которое задает структурированную иерархию доменов, используемую для организации имен;

*∙*записи ресурсов, сопоставляющие доменные имена ***DNS***определенным типам информации о ресурсах, которые используются при регистрации и разрешении имен в пространстве имен;

*∙****DNS***-серверы, которые сохраняют записи ресурсов и отвечают на запросы клиентов;

*∙****DNS***-клиенты, которые также называют системами разрешения имен, запрашивающие серверы для поиска и разрешения имен по типам записей ресурсов, указанным в запросе.



Пространство доменных имен ***DNS***, как показано на *рисунке 3*, базируется на концепции дерева именованных доменов. Каждый уровень дерева может представлять ветвь или лист дерева. Ветвь представляет уровень, на котором используется несколько имен, определяющих семейство именованных ресурсов. Лист представляет единственное имя, которое используется на этом уровне для указания конкретного ресурса.

В процессе разрешения имен существенно, что ***DNS***-серверы часто действуют как ***DNS***-клиенты, запрашивая другие серверы с целью полного разрешения имени в запросе. Любое доменное имя ***DNS***в дереве технически представляет домен. Однако принято считать, что имена идентифицируются одним из пяти способов на основании уровня и способа использования имени.

Например, доменное имя ***DNS***, зарегистрированное для образовательных учреждений (***edu.ru***), представляет домен второго уровня. Это имя состоит из двух частей (называемых метками), показывающих, что оно находится на втором уровне сверху от корня или вершины дерева. Большинство доменных имен ***DNS***содержат две или большее число меток, каждая из которых задает новый уровень в дереве. Точки используются в именах для разделения меток.

***DNS***представляет способ интерпретации полного пути к доменному имени ***DNS***аналогично интерпретации полного пути к файлу или каталогу в окне командной строки. Например, путь в дереве каталогов помогает указать на точное расположение файла, сохраненного на компьютере. Для компьютеров с операционной системой ***Windows***обратная косая черта (**\**) указывает каждый новый каталог, ведущий к точному расположению файла. Эквивалентным символом в ***DNS***является точка (**.**), указывающая каждый новый уровень домена в имени.

Для ***DNS***примером имени с несколькими уровнями может служить следующее полное доменное имя узла: ***host-a.mspu.edu.ru***

В отличие от имен файлов, при чтении полного доменного имени узла ***DNS***слева направо осуществляется переход от наиболее конкретной информации (имя ***DNS***компьютера ***host-a***) к наиболее общей (завершающая точка (**.**), которая указывает корень в дереве имен ***DNS***). Этот пример демонстрирует четыре уровня доменов ***DNS***, которые ведут от конкретного расположения ***host-a***:

∙домен ***mspu***, в котором зарегистрировано для использования имя компьютера ***host-a***;

∙домен ***edu***, который соответствует родительскому домену, являющемуся корнем поддомена ***mspu***;

∙домен ***ru***, который соответствует домену верхнего уровня, предназначенному для использования организациями из России, который является корнем для домена ***edu***;

∙завершающая точка (**.**), представляющая стандартный символ разделителя, которая используется, чтобы сделать полным доменное имя ***DNS***в дереве пространства имен ***DNS***.

**Работа запросов *DNS***

Когда ***DNS***-клиенту требуется найти имя, используемое в программе, он запрашивает ***DNS***-серверы для сопоставления имени. Каждое сообщение с запросом, отправляемое клиентом, содержит информацию трех типов, определяющую вопрос, на который отвечает сервер:

∙указанное доменное имя ***DNS***в виде полного доменного имени узла (***FQDN***);

∙указанный тип запроса, в котором задается либо тип записей ресурсов, либо тип операции запроса;

∙указанный класс доменного имени ***DNS***.

Для ***DNS***-серверов ***Windows***этот класс всегда должен быть указан как класс Интернета (***IN***).Например, указанное имя может представлять полное доменное имя узла для компьютера, такое как ***host-a.mspu.edu.ru***и тип запроса на поиск записей ресурсов адреса (A) для этого имени. Запрос ***DNS***можно представить как вопрос клиента, состоящий из двух частей, например:*«Имеются ли записи ресурсов A для компьютера с именем hostname.mspu.edu.ru?»*Когда клиент получает ответ от сервера, он читает и интерпретирует содержащуюся в ответе запись ресурса A, узнавая ***IP***-адрес компьютера, запрошенного по имени.

Запросы ***DNS***используют несколько способов сопоставления имен. Клиент может иногда ответить на запрос с помощью локальной кэшированной информации, полученной в предыдущем запросе. ***DNS***- сервер может использовать собственный кэш информации о записях ресурсов для ответа на запрос. ***DNS*-**сервер может также запросить или обратиться к другим ***DNS***-серверам в интересах запрашивающего клиента для полного сопоставления имени, а затем отправить ответ клиенту. Этот процесс называют *рекурсией*.

В дополнение к этому, клиент может самостоятельно пытаться установить контакт с дополнительными ***DNS***-серверами для сопоставления имени. При этом клиент использует отдельные дополнительные запросы, базирующиеся на ссылочных ответах от серверов. Этот процесс называют *итерацией*. Процесс запроса ***DNS***выполняется в две стадии:

1.Запрос к имени начинается на клиентском компьютере и передается в систему сопоставления имен службы ***DNS***-клиент;

2.Когда не удается ответить на запрос на локальном уровне, можно для сопоставления имени запрашивать ***DNS***-серверы по мере необходимости.

Обе стадии процесса подробнее рассматриваются далее.

**Локальная система разрешения имен**

На начальных этапах процесса в программе на локальном компьютере используется доменное имя ***DNS***. Затем запрос передается в

службу ***DNS***-клиент для сопоставления с помощью локальной кэшированной информации. Если удается разрешить запрошенное имя, поступает ответ на запрос и процесс завершается. Кэш локального сопоставления имен может включать информацию об именах из двух возможных источников:

∙если имеется локальный файл ***Hosts***, все сопоставления имен и адресов из этого файла предварительно загружаются в кэш при запуске службы ***DNS***-клиент;

∙записи ресурсов, полученные в ответах на запросы из

предыдущих запросов ***DNS***, добавляются в кэш и сохраняются в нем в течение определенного периода времени.

Если клиент не находит сопоставления в кэше, процесс продолжается с помощью запроса на разрешение имени от клиента к ***DNS***- серверу.

**Запрос к *DNS*-серверу**

Клиент запрашивает основной ***DNS***-сервер. Из глобального списка выбирается сервер, используемый на начальной стадии запроса от клиента к серверу. Когда ***DNS***-сервер принимает запрос, он сначала проверяет, можно ли дать удостоверяющий ответ на базе записей ресурсов, содержащихся в локальной зоне в конфигурации сервера. Если запрошенное имя соответствует информации в записи ресурса в локальной зоне, сервер дает удостоверяющий ответ, используя эту информацию для разрешения имени. Если в зоне нет информации для запрошенного имени, сервер проверяет, можно ли разрешить имя, используя информацию предыдущих запросов в локальном кэше. Если здесь обнаруживается совпадение, сервер отвечает с использованием этой информации. И в этом случае, если основной сервер может дать запрашивающему клиенту утвердительный ответ на сопоставление из собственного кэша, запрос завершается. Если на основном сервере не удается найти запрошенное имя

– ни в кэше, ни в зонах – процесс выполнения запроса может продолжаться с использованием рекурсии для полного разрешения имени. При этом другие ***DNS***-серверы помогают разрешить имя. Служба ***DNS***-клиент по умолчанию указывает серверу использовать процесс рекурсии для полного разрешения имен в интересах клиентов перед возвращением ответа. В большинстве случаев ***DNS***-серверы по умолчанию настраиваются на поддержку процесса рекурсии, как показано на *рисунке 4*.

Для правильного выполнения рекурсии ***DNS***-сервером ему необходимы сведения о контактах с другими ***DNS***-серверами в пространстве доменных имен ***DNS***. Такая информация обеспечивается в виде корневых ссылок, списка предварительных записей ресурсов, которые могут использоваться службой ***DNS***для обнаружения других ***DNS***- серверов, которые являются удостоверяющими для корня дерева пространства доменных имен ***DNS***. Корневые серверы являются удостоверяющими для корня доменов и доменов верхнего уровня в дереве пространства доменных имен ***DNS***.

Процесс заканчивается возвращением клиенту утвердительного ответа. Однако запросы могут возвращать и другие ответы, в частности: удостоверяющий, ссылочный и отрицательный.

*Утвердительный ответ*может содержать запрошенную запись ресурса или список записей ресурсов (который также называют набором записей), соответствующих запрошенному доменному имени ***DNS***и типу записи, указанному в сообщении запроса.

*Удостоверяющий ответ*представляет утвердительный ответ, возвращенный клиенту и доставленный с установленным битом полномочий в сообщении ***DNS***, указывающим, что ответ получен от сервера, имеющего прямые полномочия для запрашиваемого имени.

*Ссылочный ответ*содержит дополнительные записи ресурсов, не указанные по имени или типу в запросе. Ответ этого типа возвращается клиенту, если процесс рекурсии не поддерживается. Эти записи должны рассматриваться как справочные, которые могут использоваться клиентом для продолжения запроса с помощью итераций. Если клиент способен использовать итерации, он может выполнить дополнительные запросы в попытке полностью разрешить имя самостоятельно.

*Отрицательный ответ*от сервера может указывать на один из двух возможных результатов попытки сервера обработать и рекурсивно полностью и удостоверяющим образом сопоставить имя в запросе:

∙удостоверяющий сервер ответил, что запрошенное имя не существует в пространстве имен ***DNS***;

∙удостоверяющий сервер ответил, что запрошенное имя

существует, но для этого имени отсутствуют записи указанного типа.

Система сопоставления имен передает результаты запроса в виде утвердительного или отрицательного ответа в запрашивающую программу и кэширует ответ. Итерации представляют тип сопоставления имен, используемый ***DNS***-клиентами и серверами при выполнении следующих условий:

∙клиент запрашивает использование рекурсии, но рекурсия отключена на ***DNS***-сервере.

∙клиент не запрашивает использование рекурсии при запросе к

***DNS***-серверу.

Итерационный запрос от клиента сообщает ***DNS***-серверу, что клиент ожидает от ***DNS***-сервера наиболее точный ответ немедленно без обращения к другим ***DNS***-серверам. Когда используются итерации, ***DNS***- сервер отвечает клиенту о запрошенных именах на основании собственной информации о пространстве имен. Например, если ***DNS***-сервер в интрасети получает запрос от локального клиента для имени ***www.edu.ru***, он может возвратить ответ из кэша имен. Если в данный момент запрошенное имя не сохраняется в кэше сервера, то сервер может ответить предоставлением ссылки – т.е. списка записей ресурсов других ***DNS***- серверов, которые ближе к имени, запрошенному клиентом. Когда предоставляется ссылка, ***DNS***-клиент принимает на себя ответственность за продолжение итерационных запросов на сопоставление имени к другим указанным в конфигурации ***DNS***-серверам.

Например, в наиболее общем случае ***DNS***-клиент может расширить область поиска до серверов корневого домена в Интернете в попытках обнаружить удостоверяющие ***DNS***-серверы для домена ***ru***. После установления контакта с корневыми серверами Интернета клиент может получить от них дальнейшие итерационные ответы, указывающие на фактические ***DNS***-серверы Интернета для домена ***edu.ru***. Когда клиенту предоставляются записи для этих ***DNS***-серверов, он может отправить дальнейший итерационный запрос внешним ***DNS***-серверам ***edu***в Интернете, которые могут дать определенный и удостоверяющий ответ. При использовании итераций ***DNS***-сервер может также содействовать в запросе на сопоставление имени, предоставив клиенту собственный наиболее точный ответ. Для большинства итерационных запросов клиент использует локальный список ***DNS***-серверов для обращения к другим серверам имен в пространстве имен ***DNS***, если его собственный основной ***DNS***-сервер не может сопоставить имя в запросе.

По мере того как ***DNS***-серверы обрабатывают запросы клиентов с помощью рекурсии или итераций, они находят и накапливают значительный объем информации о пространстве имен ***DNS***. Эта информация *кэшируется*сервером. Кэширование дает возможность ускорить сопоставление часто используемых имен ***DNS***в последующих запросах и существенно снижает трафик запросов ***DNS***в сети.

При выполнении рекурсивных запросов ***DNS***-серверами для клиентов они временно кэшируют записи ресурсов. Кэшированные записи ресурсов содержат информацию, полученную от ***DNS***-серверов, которые являются удостоверяющими для доменных имен ***DNS***. Эта информация накапливается при выполнении итерационных запросов в процессе поиска

иполного ответа на рекурсивный запрос, выполняемый в интересах клиента. Когда затем другие клиенты размещают новые запросы на информацию, отвечающую кэшированным записям ресурсов, ***DNS***-сервер может использовать данные из кэшированных записей ресурсов для ответа.

При кэшировании информации значение срока жизни применяется ко всем кэшированным записям ресурсов. Пока не истек срок жизни кэшированной записи ресурса, ***DNS***-сервер может продолжать кэшировать

иснова использовать запись ресурса при ответах на соответствующие запросы клиентов. Значения срока жизни кэширования, используемые записями ресурсов в большинстве конфигураций зон, назначаются в параметре ***Мин. срок жизни TTL (по умолчанию)***, который задается в начальной записи зоны. По умолчанию задается значение минимального срока жизни 3600 секунд (1 час), но это значение может быть изменено, но могут также задаваться и отдельные значения срока жизни для каждой записи ресурса.

**Обратный просмотр**

В большинстве операций просмотра ***DNS***-клиенты обычно выполняют *прямой просмотр*, т. е. поиск, основанный на имени ***DNS***другого компьютера, сохраненного в записи ресурса адреса (A). В этом типе запроса в качестве данных для ответа на запрос ожидается ***IP***-адрес. ***DNS***также обеспечивает возможность *обратного просмотра*, в котором клиенты используют известный ***IP***-адрес для поиска имени компьютера по этому адресу. Обратный просмотр фактически является формой вопроса типа:*«Можете ли вы сказать мне имя****DNS****компьютера, который использует IP-адрес 192.168.1.20?».*

Система ***DNS***не разрабатывалась изначально для поддержки запросов этого типа. Одной из проблем при поддержке запросов обратного просмотра является различие в способах организации и индексации пространства имен ***DNS***и способов назначения ***IP***-адресов. Если бы единственным таким способом был бы поиск во всех доменах пространства имен ***DNS***, то для обработки обратного запроса потребовалось бы много времени, такой запрос оказался бы бесполезным.

Чтобы разрешить эту проблему, в стандартах ***DNS***был определен и зарезервирован специальный домен в пространстве имен ***DNS***Интернета, ***in-addr.arpa***, обеспечивающий практичный и надежный способ выполнения обратных запросов. Чтобы создать обратное пространство имен, поддомены в домене ***in-addr.arpa***формируются с помощью обратного упорядочения чисел в точечно-десятичной нотации ***IP***-адресов. Такое обратное упорядочение доменов для каждого октета необходимо, поскольку в отличие от имен ***DNS***, для которых ***IP***-адреса читаются слева направо, здесь интерпретация выполняется в обратном порядке. Когда ***IP***- адрес читается слева направо, информация анализируется от наиболее общей (***IP***-адрес сети в левой части адреса) до наиболее конкретной (***IP***- адрес узла в последнем октете). По этой причине порядок октетов ***IP***- адреса должен быть обращен при построении дерева домена ***in-addr.arpa***. ***IP***-адреса дерева ***DNSin-addr.arpa***могут делегироваться организациям, которым назначается ограниченный набор ***IP***-адресов в границах определенных для Интернета классов адресов. И, наконец, для дерева домена ***in-addr.arpa***, встроенного в ***DNS***, требуется определение дополнительного типа записей ресурсов – запись ресурса указателя (***PTR***). Такая запись ресурса используется для сопоставления в зоне обратного просмотра, обычно соответствующего записи ресурса именованного узла



(A) для имени ***DNS***компьютера в зоне прямого просмотра.

*Рисунок5*иллюстрирует обратный запрос, инициируемый ***DNS***- клиентом (***host-b***), которому требуется узнать имя другого узла (***host-a***) по его ***IP***-адресу 192.168.1.20.

**Рисунок 5. Обратный запрос**

Обратный запрос включает следующие этапы:

1.Клиент ***host-b***запрашивает ***DNS***-сервер о записи ресурса указателя (***PTR***), сопоставляющей IP-адрес 192.168.1.20 для

|  |  |
| --- | --- |
| имени | ***host-a.*** |
| *Поскольку* | *запрос относится к записям****PTR****, система* |

*сопоставления имен обращает адрес и добавляет имя домена****inaddr.arpa****в конец обращенного адреса. В результате образуется полное доменное имя узла (****20.1.168.192.in-addr.arpa.****), для которого будет проводиться поиск в зоне обратного просмотра;*

2.После обнаружения имени удостоверяющий ***DNS***-сервер для имени ***20.1.168.192.in-addr.arpa***может возвратить ответ с информацией записи ***PTR***. В этой информации содержится доменное имя ***DNS***узла ***host-a***, что приводит к завершению процесса обратного просмотра. *Необходимо помнить, что если запрошенное обратное имя не может быть возвращено****DNS****- сервером, можно использовать сопоставление имен****DNS****(либо рекурсию, либо итерации) для обнаружения****DNS****-сервера, который является удостоверяющим для зоны обратного просмотра и содержит запрашиваемое имя. В этом смысле процесс сопоставления имен при обратном просмотре аналогичен процессу прямого просмотра.*

**Инвертированные запросы**

*Инвертированные запросы*являются устаревшим средством, которое ранее было предложено как часть стандарта ***DNS***для поиска имени узла по его IP-адресу. В них используются нестандартные операции запросов ***DNS***, а их применение ограничено ранними версиями программы ***Nslookup***, которая является утилитой командной строки для устранения неполадок и тестирования службы ***DNS***.

Служба ***DNS***распознает и принимает сообщения инвертированных запросов и отвечает на них с имитацией ответа на запрос.

**Динамическое обновление**

*Динамическое обновление*позволяет компьютерам ***DNS***-клиентов регистрировать и динамически обновлять собственные записи ресурсов с помощью ***DNS***-сервера при каждом возникновении изменений. Это снижает необходимость администрирования записей зон вручную, в особенности для клиентов, которые путешествуют или часто меняют расположение и получают IP-адреса через ***DHCP***.

Клиентские и серверные службы ***DNS***поддерживают использование динамических обновлений, как описано в документе ***RFC 2136(Dynamic Updates in the Domain Name System)***. Служба ***DNS***-сервер поддерживает включение и отключение динамических обновлений отдельно для каждой зоны на каждом сервере, настроенном для загрузки либо стандартной основной зоны, либо зоны, интегрированной в каталоги. Служба ***DNS***- клиент будет по умолчанию динамически обновлять свои записи ресурсов узла (A) в ***DNS***, когда была выполнена настройка для ***TCP/IP***.

Динамические обновления обычно запрашиваются, когда изменяется имя ***DNS***или ***IP***-адрес компьютера. Например, для клиента с именем ***oldhost***в окне ***Свойства системы***заданы следующие имена:

|  |  |
| --- | --- |
| **Имя компьютера** | ***oldhost*** |
| **Доменное DNS-имя компьютера** | ***mspu.edu.ru*** |
| **Полное имя компьютера** | ***oldhost. mspu.edu.ru*** |

В этом примере в конфигурации компьютера нет доменных имен ***DNS***, специфических для подключения. В дальнейшем компьютер переименовывается из ***oldhost***в ***newhost***, в результате имена изменяются следующим образом:

|  |  |
| --- | --- |
| **Имя компьютера** | ***newhost*** |
| **Доменное DNS-имя компьютера** | ***mspu.edu.ru*** |
| **Полное имя компьютера** | ***newhost.mspu.edu.ru*** |

После изменения имени в окне ***Свойства системы***отображается

приглашение перезагрузить компьютер. Когда при перезагрузке компьютер запускает ОС, служба ***DHCP***-клиент выполняет следующие действия для обновления ***DNS***:

1.Служба ***DHCP***-клиент отправляет запрос для типа начальной записи зоны (**SOA**) с использованием доменного имени ***DNS***компьютера. Клиентский компьютер использует текущее полное доменное имя узла компьютера (в данном случае ***newhost.mspu.edu.ru***) как имя, указанное в этом запросе;

2.Удостоверяющий ***DNS***-сервер зоны, содержащей полное доменное имя узла клиента, отвечает на запрос типа ***SOA***;

3.После этого служба ***DHCP***-клиент пытается установить контакт с

основным ***DNS***-сервером:

*Клиент обрабатывает ответ на запрос****SOA****для его имени, чтобы определить****IP****-адрес****DNS****-сервера, удостоверенного как основной сервер, для принятия его имени. Далее он выполняет такую последовательность шагов, необходимых, чтобы установить контакт и динамически обновить его основной сервер.*

−клиент отправляет запрос на динамическое обновление основному серверу, определенному в ответе на запрос **SOA**.

*Если обновление выполняется успешно, другие действия не предпринимаются;*

−при отказе на обновление клиент отправляет запрос типа **NS**(о серверах имен) для зоны, имя которой указано в записи **SOA**;

−когда клиент получает ответ на этот запрос, он отправляет запрос **SOA**на первый ***DNS***-сервер, перечисленный в ответе;

−после разрешения имен в запросе ***SOA***клиент отправляет динамическое обновление серверу, указанному в возвращенной записи ***SOA***. *Если обновление выполняется успешно, другие действия не предпринимаются;*

−при отказе на обновление клиент повторяет запрос ***SOA***, отправляя его к следующему ***DNS***-серверу, перечисленному в ответе;

4.Как только находится основной сервер, который может выполнить обновление, клиент отправляет запрос на обновление, который обрабатывается сервером.

Содержимое запроса на обновление включает инструкции добавить

записи ресурсов A (и возможно ***PTR***) для имени ***newhost.mspu.edu.ru***и записи этих типов для ранее зарегистрированного имени ***oldhost.mspu.edu.ru***.

Сервер также выполняет проверку, разрешены ли обновления для запроса клиента. Для стандартных основных зон динамические обновления не являются безопасными, поэтому для клиентов должны выполняться любые попытки обновления. Для зон, интегрированных в службу каталогов ***Active Directory***, обновления являются безопасными и выполняются с помощью параметров безопасности, устанавливаемых на основе каталогов.

Динамические обновления отправляются или выполняются периодически. По умолчанию компьютер отправляет обновления каждые 7 дней. Если в результате обновления данные в зоне не изменяются, зона остается в текущей версии и никакие изменения не записываются. Обновления выполняются только при фактических изменениях имен и адресов в зоне или в результате добавочной зонной передачи.

**Безопасное динамическое обновление**

Безопасные обновления ***DNS***доступны только для зон, интегрированных в службу каталогов ***Active Directory***. После преобразования зоны в интегрированную становится возможным использование с консоли ***DNS***списков управления доступом. Можно добавлять пользователей и группы в списки или удалять их для указанной зоны или записи ресурса. Параметры безопасного динамического обновления для ***DNS***-серверов и клиентов по умолчанию обрабатываются следующим образом:

∙***DNS***-клиенты сначала предпринимают попытки выполнить небезопасные динамические обновления. При отказе на небезопасные обновления клиенты пытаются выполнить

безопасные обновления;

*Кроме того, клиенты используют политику обновления по умолчанию, которая позволяет им пытаться переписывать ранее зарегистрированную запись ресурса, если она специально не заблокирована условиями безопасности обновления.*

∙ после интегрирования зоны в службу каталогов ***Active Directory DNS***-серверам ***Windows Server 2003***по умолчанию разрешаются только безопасные динамические обновления.

*При использовании стандартного сохранения зон настройки по умолчанию службы****DNS****-сервер не разрешают динамические обновления зон. И для зон, интегрированных в каталоги, и для использующих стандартное сохранение в файлах можно изменить параметры зоны и разрешить динамические обновления. Это позволяет принимать любые обновления.*

При развертывании ***DNS***-серверов совместно с ***Active Directory***необходимо иметь в виду следующее:

*∙*служба ***DNS***требуется для обнаружения контроллеров доменов ***Windows Server 2003***. Служба сетевого входа в систему использует новые средства поддержки ***DNS***-серверов для обеспечения регистрации контроллеров доменов в пространстве доменных имен ***DNS***;

*∙****DNS***-серверы ***Windows Server 2003***могут использовать службу

каталогов ***Active Directory***для сохранения и репликации зон.

*При интегрировании зон в службу каталогов пользователи получают возможность использовать дополнительные средства****DNS****, такие как безопасные динамические обновления и средства устаревания и очистки записей.*

Способы интеграции ***DNS***со службой каталогов ***Active Directory***:

*∙*при установке ***Active Directory***на сервер выполняется повышение сервера до роли контроллера указанного домена. Когда данный процесс завершается, пользователю выводится приглашение указать доменное имя ***DNS***для домена ***Active Directory***, для которого выполняется присоединение и повышение сервера;

*∙*если в этом процессе удостоверяющий ***DNS***-сервер для

указанного домена либо не обнаруживается в сети, либо не поддерживает протокол динамического обновления ***DNS***, выводится приглашение установить ***DNS***-сервер. Такая возможность предоставляется, поскольку ***DNS***-серверу необходимо отыскать этот сервер или другие контроллеры домена для рядовых серверов домена ***Active Directory***.

После установки ***Active Directory***имеются две возможности сохранения и репликации зон при работе с ***DNS***-сервером на новом контроллере домена:

*∙*стандартное сохранение зоны с помощью файла в текстовом

*формате. Зоны, сохраняемые этим способом, размещаются в файлах с расширением****DNS****, которые сохраняются в папке****systemroot\System32\Dns****на каждом компьютере, на котором выполняется****DNS****-сервер. Имя файла зоны соответствует имени, которое пользователь выбрал для зоны при ее создании, например,****mspu.edu.ru.dns****, если именем зоны является****mspu.edu.ru.dns****;*

∙сохранение зон, интегрированных в службу каталогов, с помощью базы данных ***Active Directory***. *Зоны, сохраняемые таким образом, размещаются в дереве****Active Directory****под разделом каталога домена или приложения. Каждая зона, интегрированная в службу каталогов, сохраняется в контейнере****dnsZone****, который идентифицируется по имени, выбранному пользователем при ее создании.*

**Преимущества интеграции с *Active Directory***

В сетях с развертыванием ***DNS***для поддержки службы каталогов ***Active Directory***настоятельно рекомендуется использовать основные зоны, интегрированные в службу каталогов, которые предоставляют следующие преимущества:

∙обновление с несколькими главными серверами и расширенные средства безопасности, базирующиеся на возможностях ***Active Directory***. В модели стандартного сохранения зон обновления ***DNS***выполняются на основе модели с единственным главным сервером. В такой модели единственный удостоверяющий ***DNS***- сервер зоны обозначается как основной источник для зоны. Это сервер содержит главную копию зоны в файле на локальном диске. В этой модели основной сервер зоны представляет единственную фиксированную точку отказа. Если этот сервер недоступен, запросы на обновление зоны от ***DNS***-клиентов не обрабатываются. При сохранении зон, интегрированных в службу каталогов, динамические обновления ***DNS***выполняются с использованием модели с несколькими главными серверами. В этой модели любой удостоверяющий ***DNS***-сервер, например, контроллер домена, выполняющий службу ***DNS***-сервер, обозначается как основной источник для зоны. Поскольку главная копия зоны поддерживается в базе данных ***Active***

***Directory***, которая полностью реплицируется на все контроллеры домена, зона может обновляться любыми ***DNS***-серверами, выполняющимися на любом контроллере домена.При использовании модели ***Active Directory***с несколькими главными серверами любой из основных серверов для зоны, интегрированной в каталоги, может обрабатывать запросы от ***DNS***-клиентов на обновление зоны, пока контроллер домена является доступным по сети. Кроме того, при использовании зон, интегрированных в службу каталогов, можно с помощью списков управления доступом защитить объект-контейнер ***dnsZone***в дереве каталогов. Это средство обеспечивает дифференцированный доступ к зоне или к конкретной записи ресурса в зоне. Например, список управления доступом для записи ресурса в зоне можно ограничить так, чтобы разрешить динамические обновления только указанному компьютеру клиента или группе безопасности, например, группе администраторов домена. Это средство безопасности недоступно для стандартных основных зон. Необходимо отметить, что при преобразовании зоны к типу интегрированной

вслужбу каталогов настройка по умолчанию для обновлений зоны изменяется, и разрешаются только безопасные обновления. Кроме того, при использовании списков управления доступом на объектах ***Active Directory***, относящихся к ***DNS***, списки управления доступом могут применяться только к службе ***DNS***- клиент;

***∙***репликация и синхронизация зон с новыми контроллерами домена выполняется автоматически при каждом добавлении нового контроллера в домен ***Active Directory***. Хотя службу ***DNS***можно выборочно удалять с контроллеров домена, зоны, интегрированные в службу каталогов, всегда сохраняются на каждом контроллере домена. В результате сохранение и управление зонами не является дополнительным ресурсом. Кроме того, способы синхронизации информации, сохраняемой

вслужбе каталогов, обеспечивают повышение быстродействия по сравнению со стандартными способами сохранения обновлений зон, которые могут потенциально потребовать передачи зоны целиком;

∙за счет сохранения баз данных зон ***DNS***в ***Active Directory***имеется возможность рационализировать репликацию баз данных в сети. Когда пространство имен ***DNS***и домены ***Active Directory***сохраняются и реплицируются независимо, необходимо обеспечить планирование и администрирование каждого из них в отдельности. Например, при одновременном использовании стандартного сохранения зон ***DNS***и службы каталогов ***Active Directory***необходимо обеспечить структуру, реализацию, тестирование и управление для двух различных топологий репликации баз данных. Одна топология требуется для репликации данных из каталогов между контроллерами домена, а другая топология может потребоваться для репликации баз данных зон между ***DNS***-серверами.Это приведет к дополнительным трудностям при планировании и разработке структуры сети с учетом ее естественного роста. За счет интеграции сохранения информации ***DNS***появляется возможность унифицировать вопросы управления и репликации для ***DNS***и ***Active Directory***, объединяя их в единое административное целое;

∙репликация каталогов выполняется быстрее и эффективнее, чем стандартная репликация ***DNS***. Поскольку репликация ***Active Directory***выполняется на уровне отдельных свойств, распространяются только необходимые изменения. При этом для зон, интегрированных в службу каталогов, используется и отправляется меньший объем данных.

**Задание 8**

Служба каталогов ASTRA LINUX DIRECTORY

**Введение**

В ходе выполнения лабораторной работы, учащимся необходимо создать единое пользовательское пространство на основе службы Astra Linux Directory (ALD).

**Цель**

Создание единого пользовательского пространства ALD.

**Задачи**

1. Настройка проводного соединения между сервером и клиентом.
2. Установка пакетов серверной и клиентской части службы ALD и графических утилит администрирования.
3. Конфигурация службы ALD сервера и клиента.
4. Проверка работоспособности единого пользовательского пространства.

**Используемое программное обеспечение**

Для выполнения лабораторной работы используется VirtualBox и установленный дистрибутив ОС Astra Linux.

**Порядок выполнения задания:**

**Проверка имени компьютера**

Проверили имя компьютера через свойства (см. рис. 1, 2). Также можно проверить имя компьютера командой hostname через терминал.

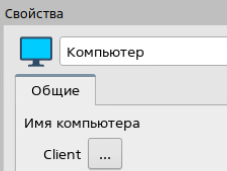


Рис. 1. Проверка имени на клиенте

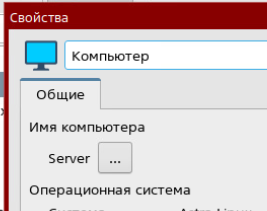


Рис. 2. Проверка имени сервера

**Установка ALD на клиенте**

До начала настройки сети необходимо установить пакеты из интернет репозитория, в противном случаем, после настройки сети не будет доступа до сети «Интернет».

Установка пакетов осуществляется по средствам менеджера пакетов Synaptic. Установили пакеты (см. рис. 3):

*ald-client-common*

*fly-admin-ald-client*

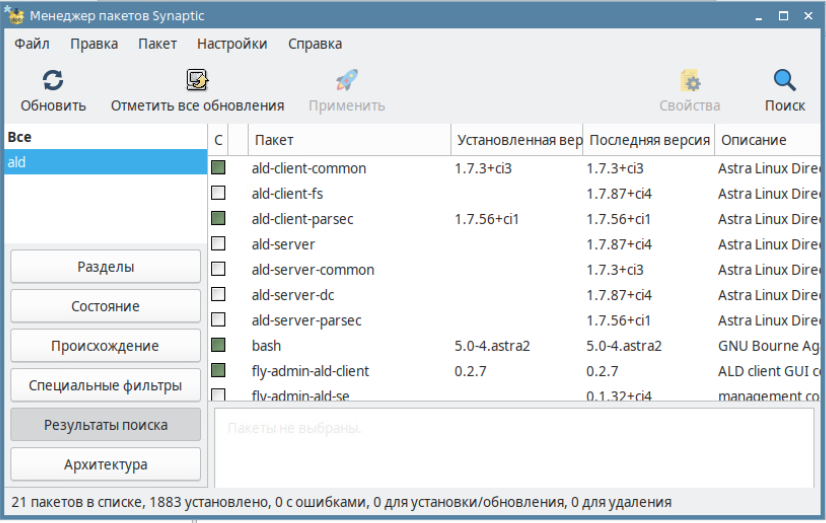


Рис. 3. Установленные пакеты на клиенте

**Настройка сети**

Настроили сеть на клиенте и сервере (см. рис. 4, 5).

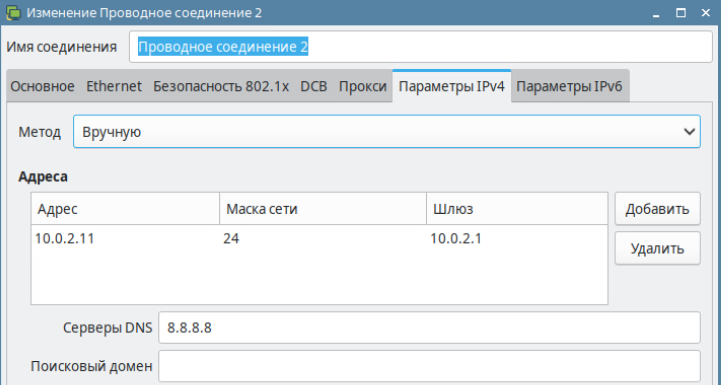


Рис. 4. Настройка на клиенте

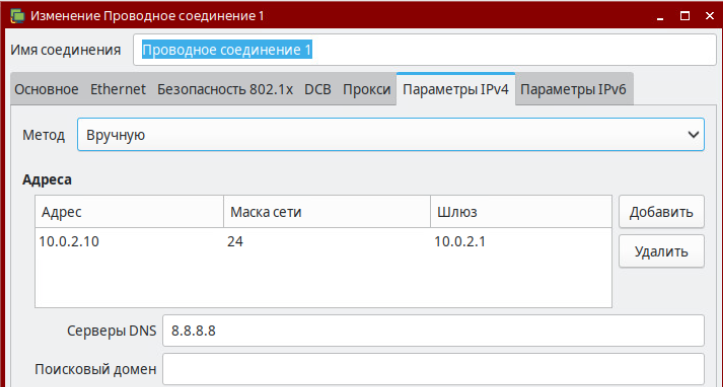


Рис. 5. Настройка на сервере

**Конфигурация файла hosts**

Конфигурирование файла hosts необходимо для взаимодействия виртуальных машин по их именам, без использования DNS.

В терминале выполнили следующие команды:

*sudo –i*

*kate /etc/hosts*

Команда *sudo –i*, меняет текущего пользователя на root.

Команда *kate /etc/hosts*, открывает файл hosts, и позволяет его редактировать.

После выполнения данных команд откроется файла hosts, в котором дописали строки привязки сервера и клиента (см. рис. 6, 7).

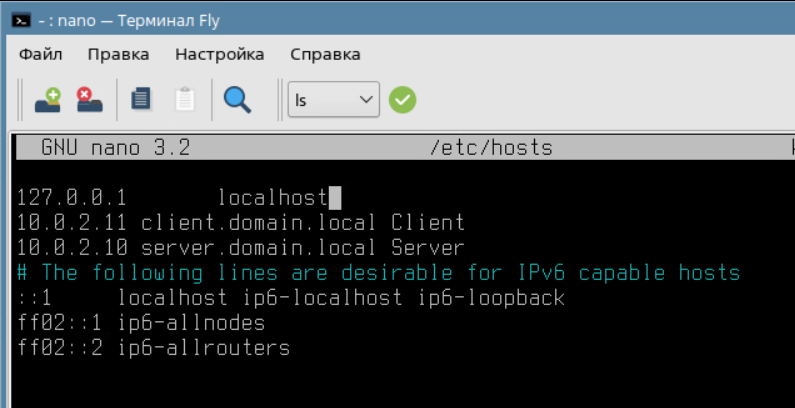


Рис. 6. Файл hosts на клиенте

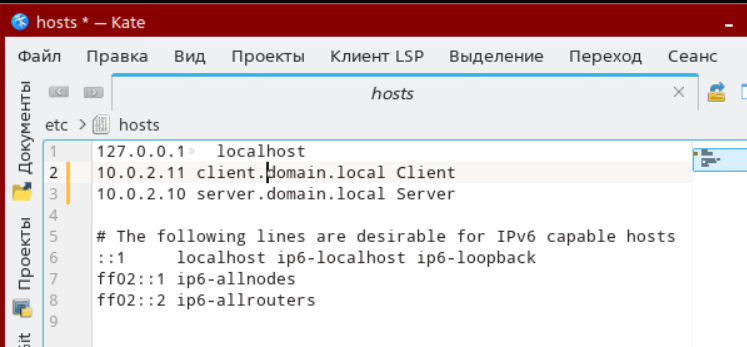


Рис. 7. Файл hosts на сервере

**Проверка работоспособности сети**

Выполнили команду (см. рис. 8, 9):

*ping client.domain.local* – на сервере

*ping server.domain.local* – на клиенте

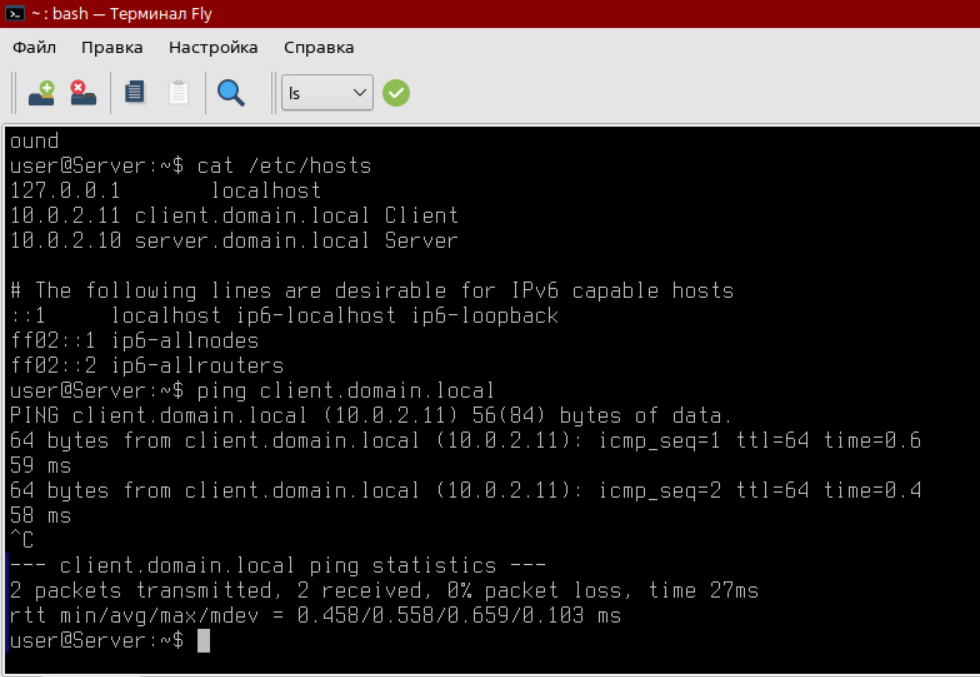


Рис. 8. Ping с сервера на клиента

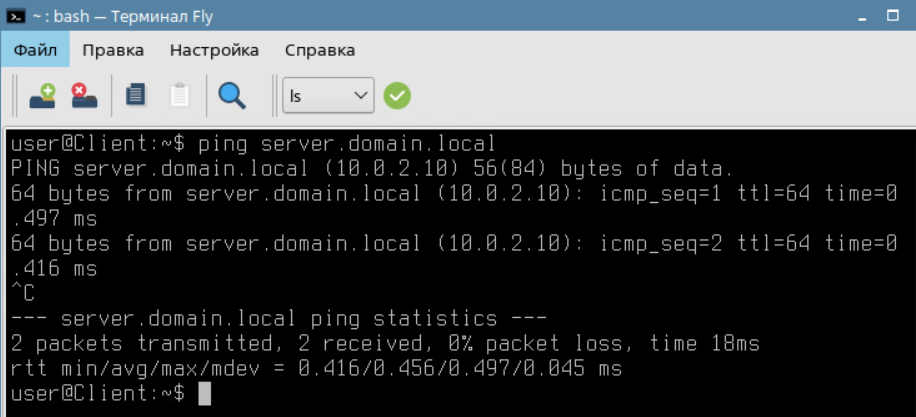


Рис. 9. Ping с клиента на сервер

**Установка ALD на сервер**

Запустили Менеджер пакетов Synaptic и установили следующие пакеты (см. рис. 10):

*ald-server-common*

*ald-admin-common*

*fly-admin-ald-server*

Пакет *ald-server-common* содержит набор программ и утилит для работы службы ALD. Пакет *ald-admin-common* содержит набор программ и утилит для администрирования службы ALD.

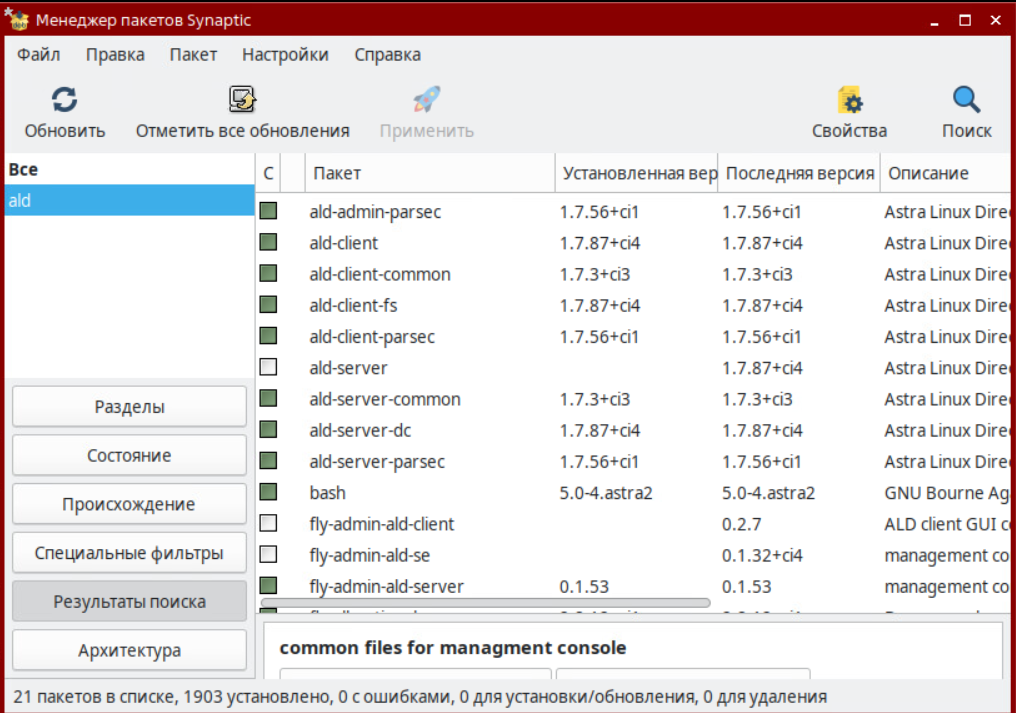


Рис. 10. Установленные пакеты на сервер

**Настройка службы ALD**

Установка и настройка ЕПП ALD, заключается в создании домена, создании пользователя домена и настройка привилегий пользователя. Для настройки службы ALD в панели управления перешли в раздел сеть и запустили доменную политику безопасности.

Поле запуска программы в открывшемся окне ввели пароль *admin*. Дале перешли в раздел Создание ALD сервера.

При создании ALD сервера указали данные как на рисунке 11.

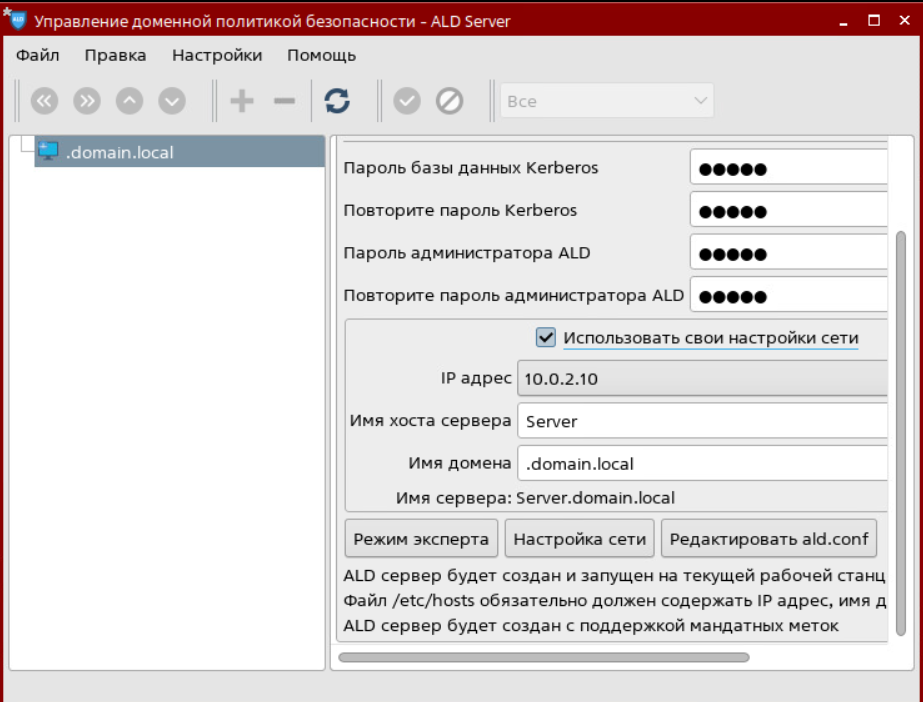


Рис. 11. Настройка ALD сервера

После происходит процесс инициализации ALD сервера (см. рис. 12).

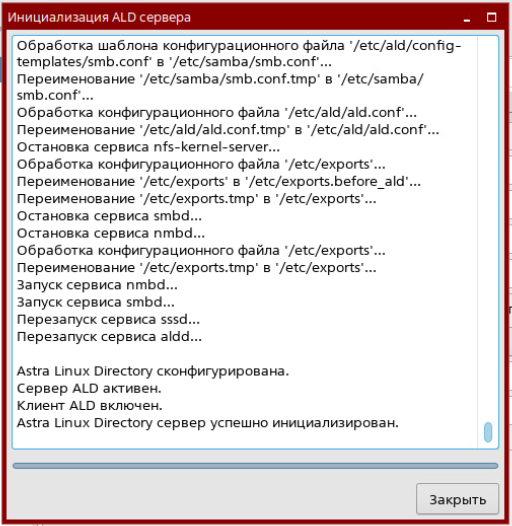


Рис. 12. Процесс инициализации ALD сервера

Подключаем домен. В левой части программы Доменной политики безопасности, отобразилась структура домена (см. рис. 13).

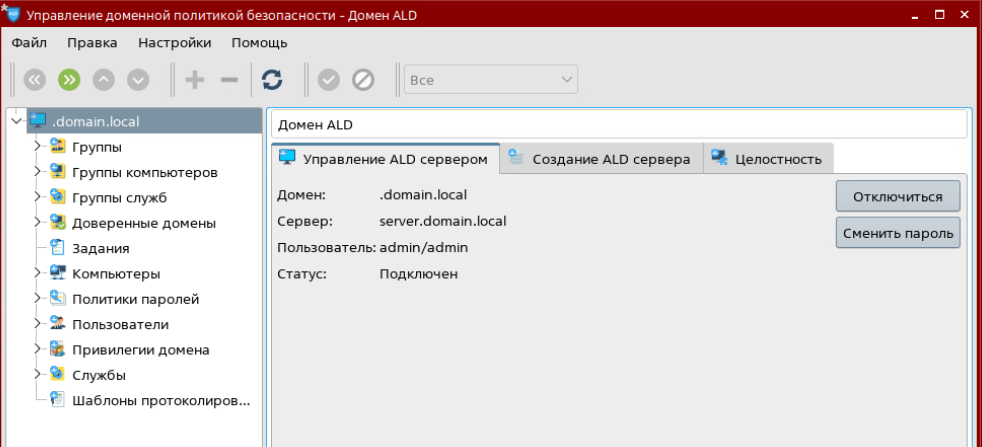


Рис. 13. Дерево домена

**Настройка ALD на клиента и подключение к домену**

Конфигурация служб ALD на компьютере клиента, а также утилит для подключения компьютера к домену используя графический интерфейс.

Запустили Настройку ALD клиента Fly и указали настройки, приведенные на рисунке 14.

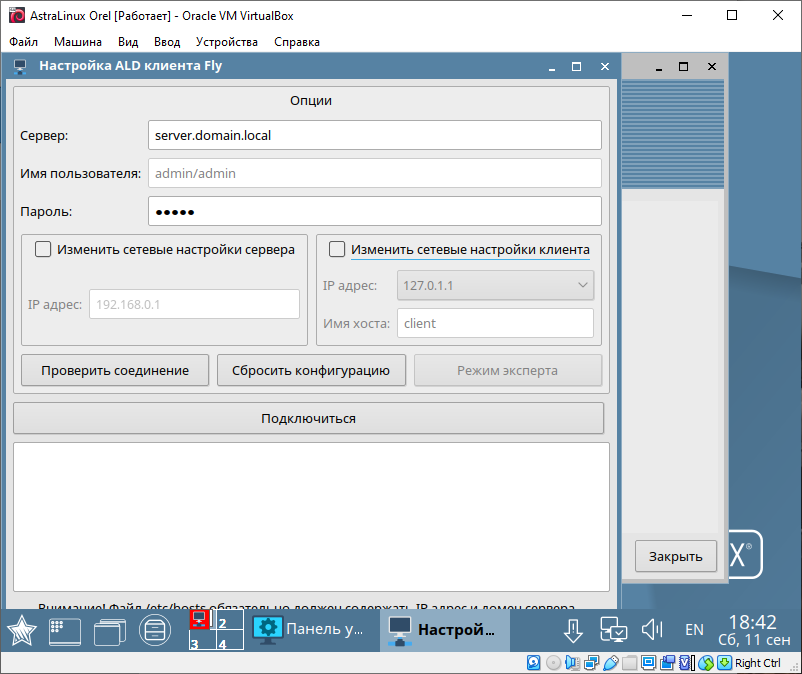


Рис. 14. Настройка ALD клиента Fly

Послу ввода настроек проверили соединение с сервером службы ALD, кнопкой Проверить соединение.

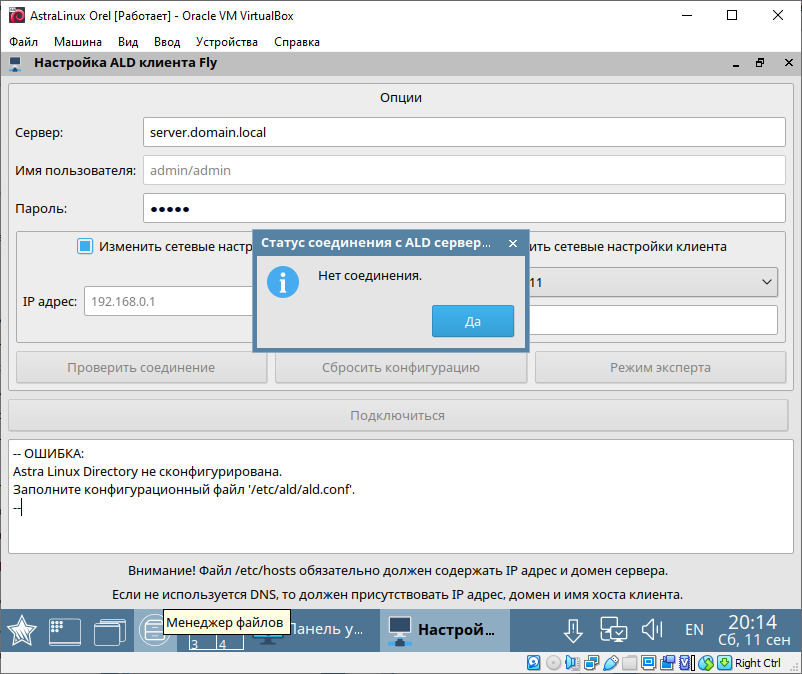


Рис. 10 Нет соединения

Настройка ALD клиента Fly, сообщает об ошибке, а именно заполните конфигурационный файл /etc/ald/ald.conf. Для заполнения файла ald.conf, запускаем терминал и с помощью графического текстового редактора Kate открываем данный файла.

В данном конфигурационном файле задали значения, соответствующие таблице 1.

Таблица 1. Конфигурационный файл ald.conf

|  |  |  |
| --- | --- | --- |
| № п.п. | Параметр | Значение |
| 1 | DOMAIN | .domain.local |
| 2 | SERVER | server.domain.local |
| 3 | SERVER\_ON | 1 |
| 4 | CLIEBT\_ON | 1 |

После изменения конфигурационного файла выполнили проверку соединения с сервером ALD (см. рис. 15).

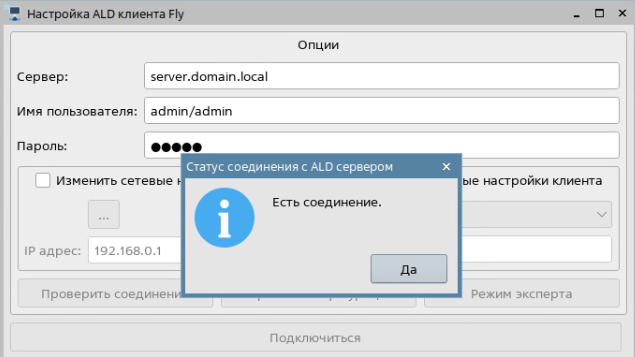


Рис. 15. Статус соединения с ALD сервером

После успешной проверки нажали на кнопку Подключиться (см. рис. 16).

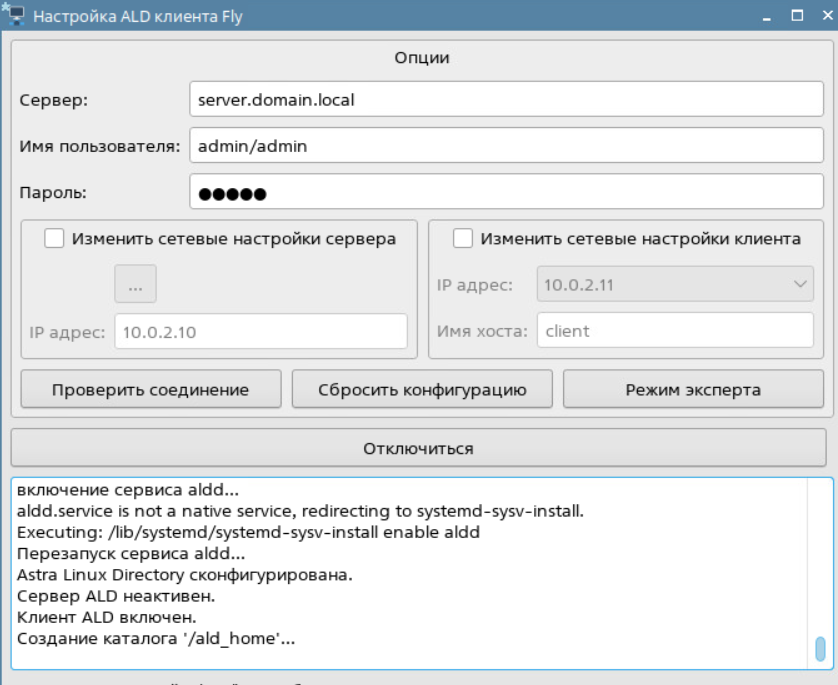


Рис. 16. Настройка ADL клиента

**Создание и настройка пользователя ALD**

Создание пользователя средствами управления доменной политикой безопасности и присвоение определенных привилегий домена.

Для создания пользователя на сервере ALD запустили Управления доменной политикой безопасности, в выпадающем списке .domain.local перешли во вкладку Пользователи и нажали на кнопку плюс. Далее заполнили поле Имя и выбрали Тип ФС local.

После создания пользователя изменили ему пароль (см. рис. 17).

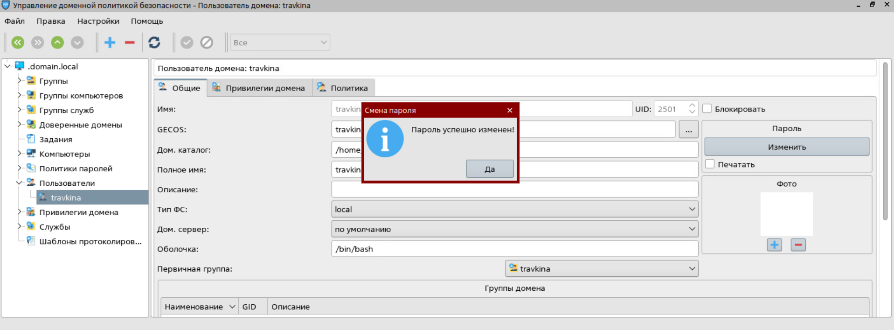


Рис. 17. Смена пароля

После успешного создания пароля перешли в раздел Привилегии домена и в блоке компьютеры добавили компьютер *client.domain.local*. Затем проверили наличие созданного пользователя (см. рис. 18).

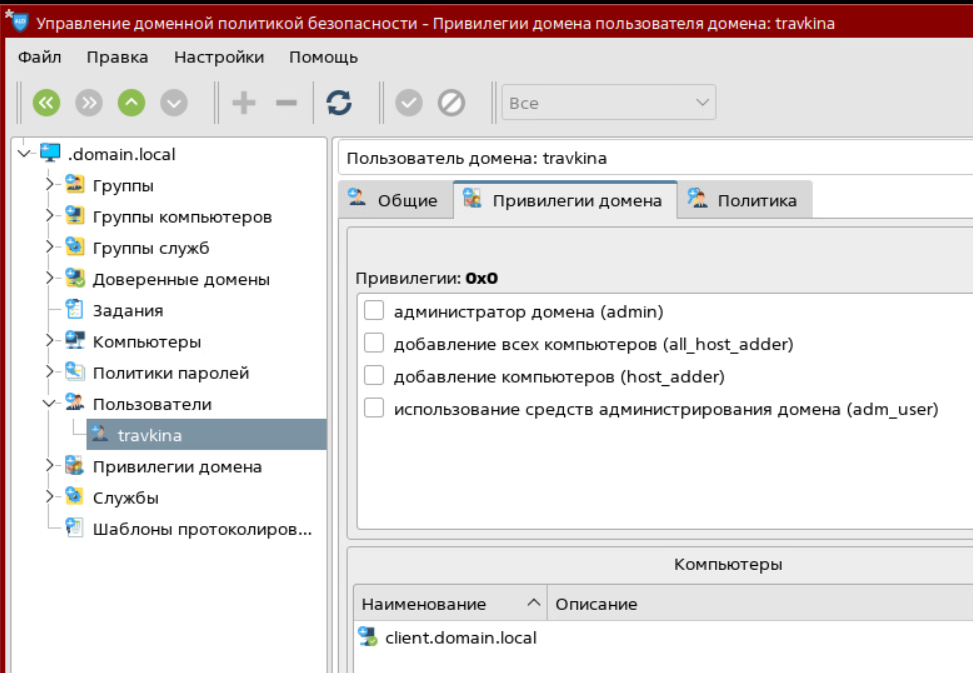


Рис. 18. Привилегии домена

**Проверка работоспособности ALD**

После выхода из системы, нажали кнопку Сессия, выбрали новую сессию.

После ввода учетных данных нового пользователя осуществили проверку учетных данных пользователя, для этого зашли в терминал Fly (см. рис. 19).

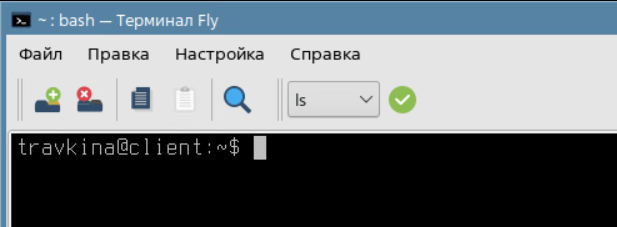


Рис. 19. Терминал Fly нового пользователя