МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

ОТЧЕТ

по лабораторной работе №1

«Сетевое взаимодействие Windows и Linux»

по дисциплине

«Администрирование систем и сетей»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_ Кочешков А. А.\_\_

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Игнаков К. М.

(подпись) (фамилия, и.,о.)

\_\_\_\_\_\_19-ВМ\_\_\_\_\_\_\_\_

(шифр группы)

Работа защищена «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород 2023

Цель работы

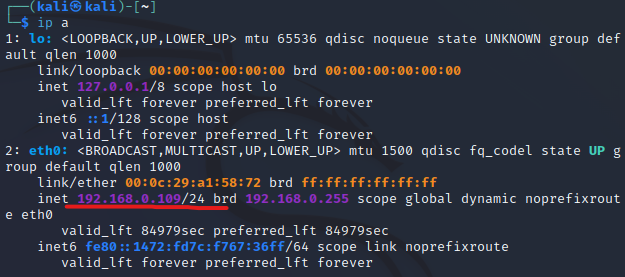
Изучение механизмов и средств интеграции Windows и Linux на базе сетевых файловых систем.

Ход работы

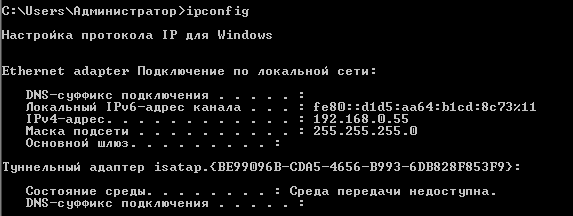
Взаимодействие на базе протокола SMB

Проверка сетевого доступа по протоколу TCP/IP.

Необходимо чтобы сервера находились в одной локальной сети или у них были “белые” ip-адреса. Так как компьютеры находятся в одной локальной сети, то следующим шагом необходимо узнать их ip-адреса. Это можно сделать с помощью команд: ipconfig для Windows, ifconfig или ip a[ddr] для Linux.



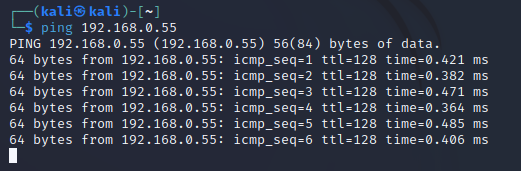
IP-адрес линукс сервера.



IP-адрес для Windows Server.

С помощью утилиты ping необходимо проверить “отвечают” ли друг другу сервера в сети.

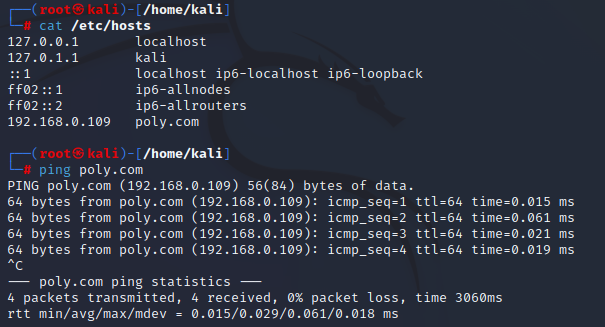




Сервера “пингуются” можно увидеть что их ip-адрес это 192.168.0.0/24.

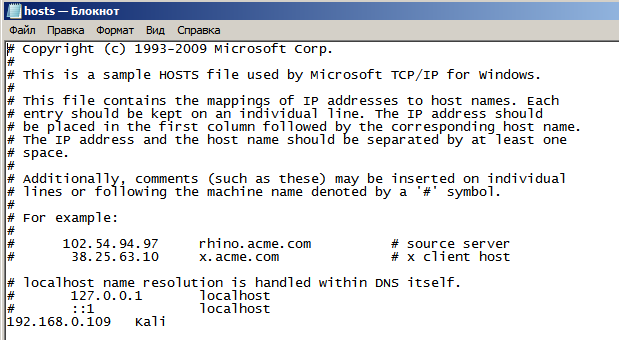
Настроить локальные файлы трансляции DNS-имен.

Для настройки DNS имен в Linux обычно служит файл /etc/hosts. Для того чтобы добавить запись необходимо открыть его от имени администратора и сделать запись вида: <ip> <domain-name>.

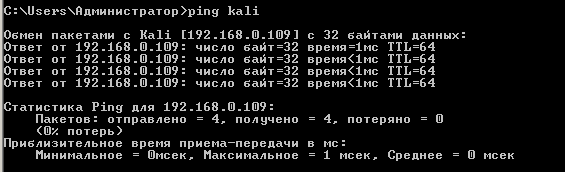


Как видим после добавление записи в /etc/hosts Windows сервер сразу пингуется по доменному имени.

Для настройки DNS записи в Windows служит файл %SYSTEMROOT%\system32\drivers\etc\hosts. Правила записи там такие же:



Проверим что linux сервер теперь так же доступен по доменному имени:



Реализация сервера файлового доступа SMB на Linux-системе.

Ознакомиться с составом Samba.

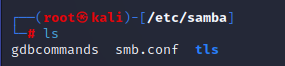
Samba - это бесплатный программный пакет с открытым исходным кодом, предоставляющий услуги по работе с файлами и печати для клиентов Windows. Он обеспечивает совместимость между серверами Linux/Unix и клиентами на базе Windows, делая возможным совместное использование и печать файлов в различных операционных системах. Samba использует протокол SMB/CIFS для предоставления этих услуг и позволяет интегрироваться с Active Directory для централизованной аутентификации и авторизации. Она также поддерживает такие функции, как блокировка файлов, управление доступом и совместное использование принтеров. Samba широко используется в корпоративных и домашних средах и является популярным выбором для создания файловых серверов, контроллеров домена и серверов печати.

Для установки необходимо выполнить команду: apt install samba.



При установке Samba в системе Linux, вместе с ней могут быть установлены несколько файлов и пакетов, в зависимости от дистрибутива Linux и конкретной конфигурации Samba. Ниже приведен обзор некоторых распространенных файлов и пакетов, которые могут быть установлены вместе с Samba:

1. Двоичные и конфигурационные файлы Samba: Эти файлы являются ядром Samba и необходимы для работы программы. Двоичные файлы обычно находятся в каталоге /usr/sbin или /usr/bin, а файлы конфигурации - в каталоге /etc/samba. Некоторые общие файлы в этом каталоге включают smb.conf (основной файл конфигурации Samba), smbusers (файл, сопоставляющий пользователей Samba с пользователями системы) и secrets.tdb (файл, хранящий зашифрованные пароли).





1. Клиентские утилиты Samba: это программы, которые позволяют пользователю получить доступ к общим ресурсам Samba с клиентского компьютера. Некоторые распространенные клиентские утилиты Samba включают smbclient (утилита командной строки, позволяющая просматривать общие ресурсы Samba и работать с ними), smbmount (программа, позволяющая монтировать общие ресурсы Samba как файловую систему) и smbpasswd (утилита для изменения паролей пользователей Samba).
2. Пакеты для сервера Samba: В зависимости от дистрибутива Linux, может быть установлено несколько пакетов, которые обеспечивают дополнительную функциональность сервера Samba. Например, пакет "samba-common" обычно устанавливается для обеспечения общих файлов и библиотек, используемых сервером и клиентом Samba. Пакет "samba-dsdb-modules" может быть установлен для обеспечения реализации Samba базы данных доменных служб Windows.
3. Пакеты аутентификации Samba: Эти пакеты используются для аутентификации пользователей Samba в различных системах аутентификации, таких как локальный файл паролей Linux, службы каталогов LDAP или Active Directory. Некоторые распространенные пакеты аутентификации Samba включают "samba-winbind" (который позволяет Samba использовать контроллеры домена Windows для аутентификации) и "samba-krb5" (который обеспечивает поддержку аутентификации Kerberos для Samba).
4. Пакеты печати Samba: Samba также может быть использована для предоставления услуг печати клиентам Windows. Чтобы включить эту функцию, можно установить несколько пакетов, связанных с печатью, таких как "cups" (общая система печати, используемая в Linux), "cups-client" (утилита для управления заданиями печати) и "samba-printing" (пакет, предоставляющий услуги печати Samba).

Изучить конфигурирование Samba.

Файл smb.conf является основным конфигурационным файлом для сервера Samba, который используется для обмена файлами и принтерами с клиентами Windows по протоколу SMB. Файл расположен в каталоге /etc/samba/ и содержит различные параметры конфигурации для сервера Samba.

Файл smb.conf разделен на секции, каждая из которых начинается с имени, заключенного в скобки ([]). Разделы содержат различные параметры, определяющие работу Samba. Наиболее распространенными разделами являются [global], [homes], [printers] и [shares].

Раздел [global] содержит глобальные настройки для сервера Samba, такие как имя рабочей группы, настройки безопасности и параметры протоколирования. Этот раздел необходим в каждом файле smb.conf.

Раздел [homes] используется для настройки домашних каталогов пользователей, позволяя им получать доступ к своим личным файлам на сервере Samba.

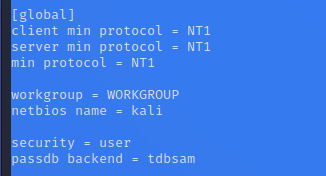
Раздел [printers] используется для настройки общего доступа к принтерам на сервере Samba.

Раздел [shares] используется для определения общих ресурсов для определенных каталогов или файлов на сервере Samba, что позволяет пользователям получать доступ к общим файлам и каталогам.

Каждый раздел содержит различные параметры, такие как путь к общему каталогу, разрешения на чтение и запись, права собственности пользователей и групп, а также параметры безопасности.

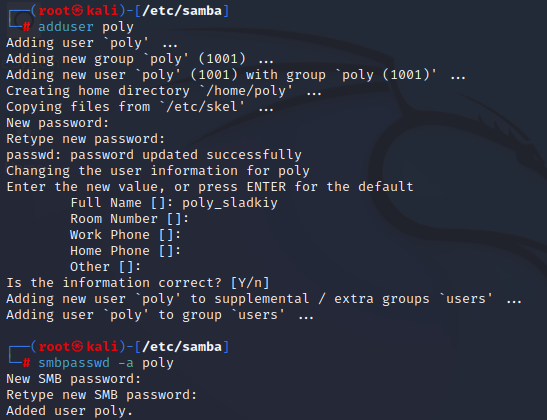
Файл smb.conf можно редактировать вручную с помощью текстового редактора, но рекомендуется использовать инструмент настройки сервера Samba, такой как SWAT (Samba Web Administration Tool) или редактор smb.conf, входящий в дистрибутив Linux.

Для тестовой настройки пропишем в глобальные правила netbios name, рабочую группу и аутентификацию на уровне пользователя:

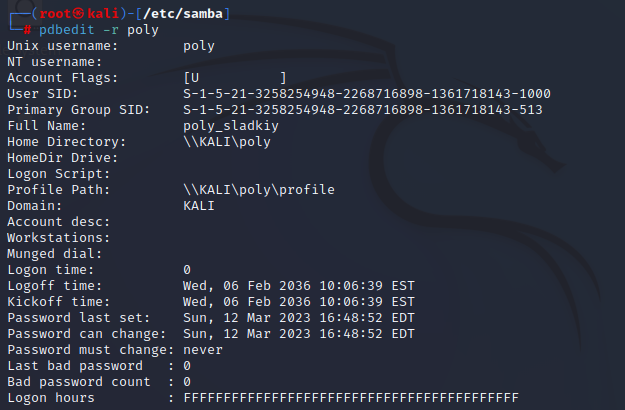


Создать учетную запись пользователя Samba.

Перед тем как создать пользователя Samba необходимо создать пользователя на самом сервер. Сделать это можно с помощью утилиты adduser <user\_name>. После это с помощью поставленной утилиты smbpasswd и флагом “-a” создаем пользователя с тем же именем.

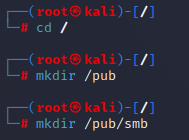


Чтобы посмотреть информацию об учетной записи можно воспользоваться командой pdbedit с флагом “-r”, например, SID, время последнего входа в систему,

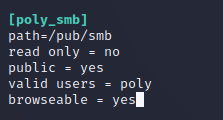


Внутри локального каталога /pub сформировать каталог для выделения в сетевой доступ по smb-протоколу

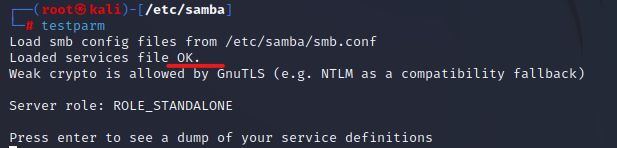
Создадим в корне директорию, к которой потом будет подключаться с клиентского smb.



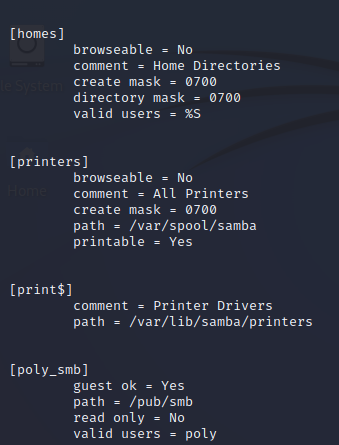
Опишем сетевой ресурс с именем ресурс в smb.conf с различными типами доступа.



Для проверки корректно ли введены настройки воспользуемся testparm:

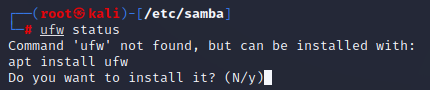


В выводе есть подсказка, что при нажатии на ‘Enter’ в консоль будет выведен дамп конфигурационного файла (в нем отсутствуют комментарии):

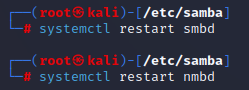


Подготовить условия для подключения smb-клиентов и стартовать сервер Samba.

Отключим межсетевой экран командой **ufw disable**, и проверим его состояние командой **ufw status**. Как видим межсетевой экран отсутствует в данной сборке linux дистрибутива.



Перезапустим сервер Samba:

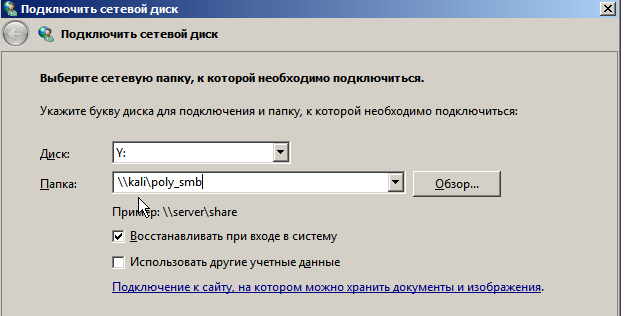


Проверим появились ли деманические процессы samba. Как видим все процессы успешно запустились.

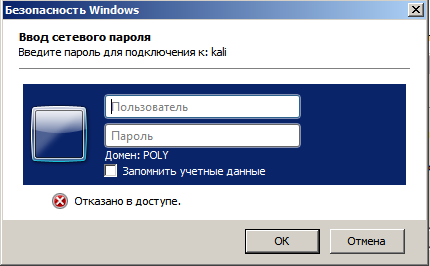


В Windows-клиенте проверить доступность и свойства сетевых ресурсов.

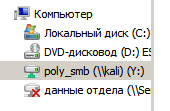
Попробуем подключить сетевой диск в проводнике на Windows:



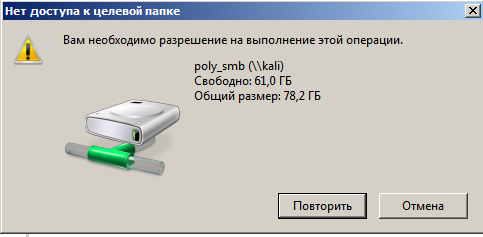
Вводим учетные данные пользователя, которые указывали при конфигурировании Samba на linux сервере.



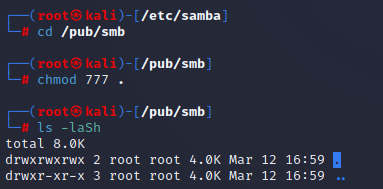
Сетевой диск успешно подключен:



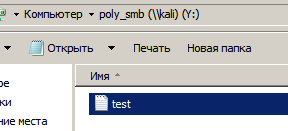
При попытке создать файл или директорию в сетевой диске возникает исключение с ошибкой “Нет доступа к целевой папке”. Эту проблему можно решить несколькими способами: выдать все права на сетевую директорию (777), сделать владельца директории poly, выдать права для пользователя poly и другие.



Перейдем в сетевую директорию и выдадим все права, а так же проверим их:

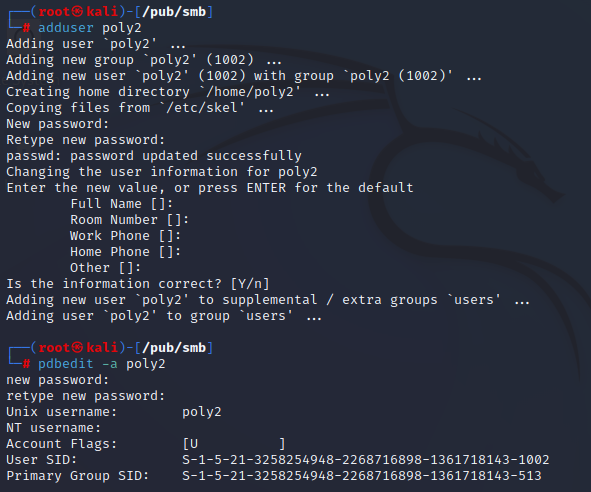


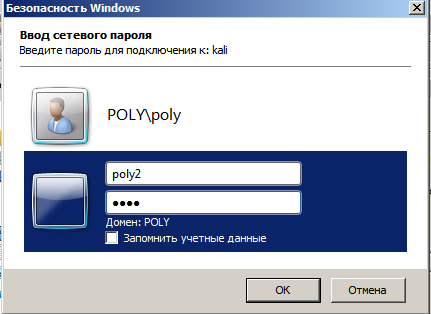
Повторим попытку и увидим, что ошибка пропала:



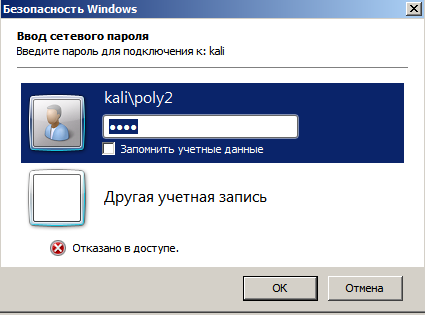
Проверка прав на подключение

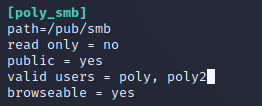
Создадим нового пользователя и попробуем подключиться от его имени на Windows smb клиенте:



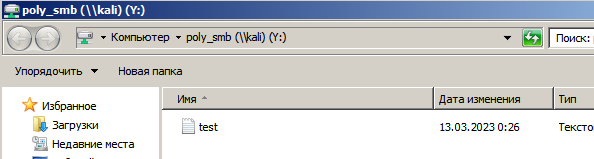


Получено сообщение об отказе в доступе, для этого нужно новую учетную запись надо добавить в конфигурационный файл Samba.

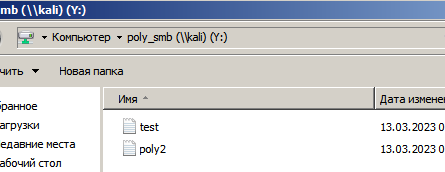




Перезапустим smb сервер как было сказано выше и попробуем присоединиться снова:

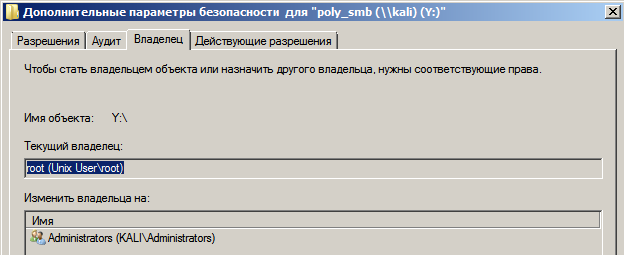


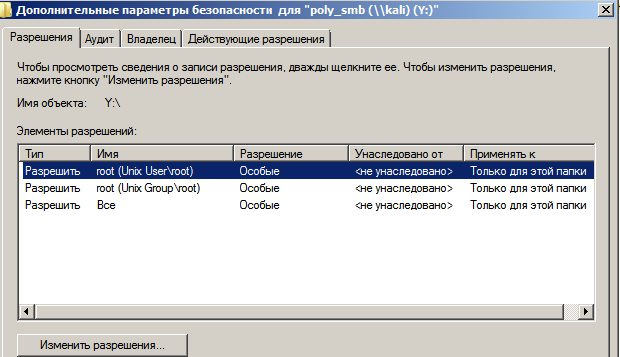
Попробуем создать текстовый документ poly2. Как видим, что все прошло успешно:



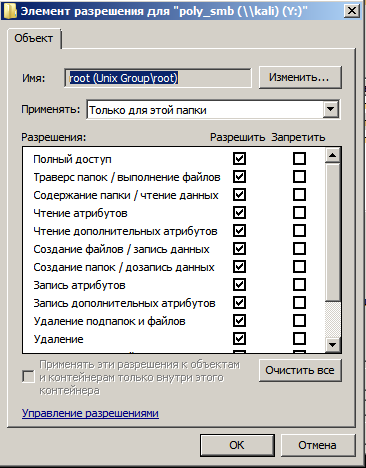
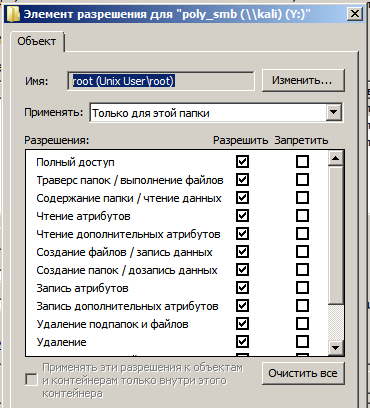
Разрешения доступа к каталогу

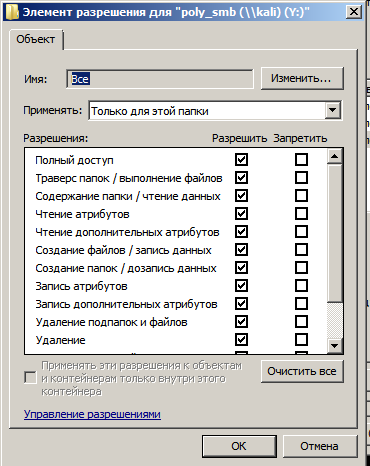
Посмотрим информацию о разрешениях доступа и владельце подключенного ресурса.





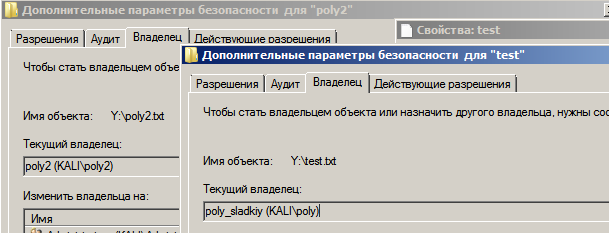
Как видим, владельцем ресурса является root. У всех пользователей и групп стоят «Особые разрешения», посмотрим элементы разрешения и увидим полный доступ.

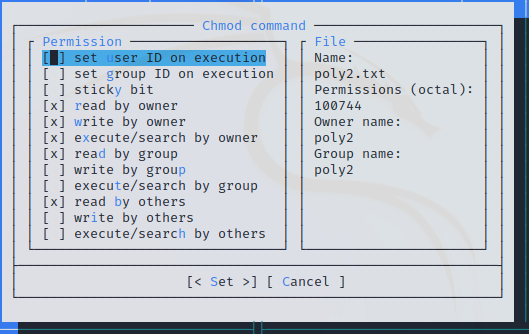
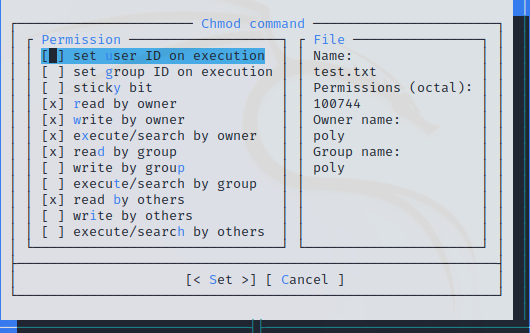




Можно предположить, что перед тем как мы выдали все права, доступ к файлам и папкам имел только root. А так как в конфигурации мы запретили подключаться от имени root, то никто не имел права на файлы и папки в этой директории.

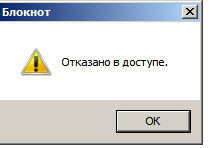
Посмотрим права и владельцев файлов, которые были созданы выше:



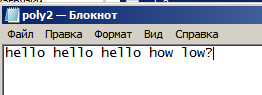


Как видим, владельцем проставился пользователь, от которого был присоединен сетевой диск.

Попробуем внести изменения в файл от пользователя poly2 в файл test, который принадлежит poly. При сохранении система предложила создать файл, если попробовать перезаписать существующий, то получим ошибку доступа:



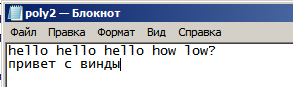
Попробуем изменить файл принадлежащий текущему пользователю:



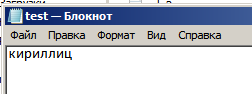
Сохранение прошло успешно, проверим изменения в файле на linux сервере:



Попробуем ввести кириллицу с Windows и Linux:

Посмотрим, что храниться в файлах, текст с линукс устпешно отобразился в виндовс:



Файл сохраненный с виндовс открылся с проблемами на линукс:

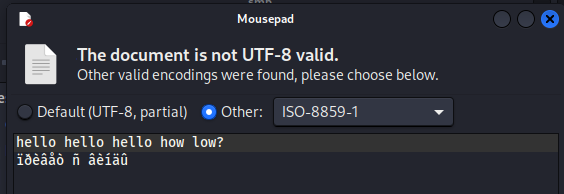
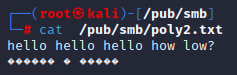
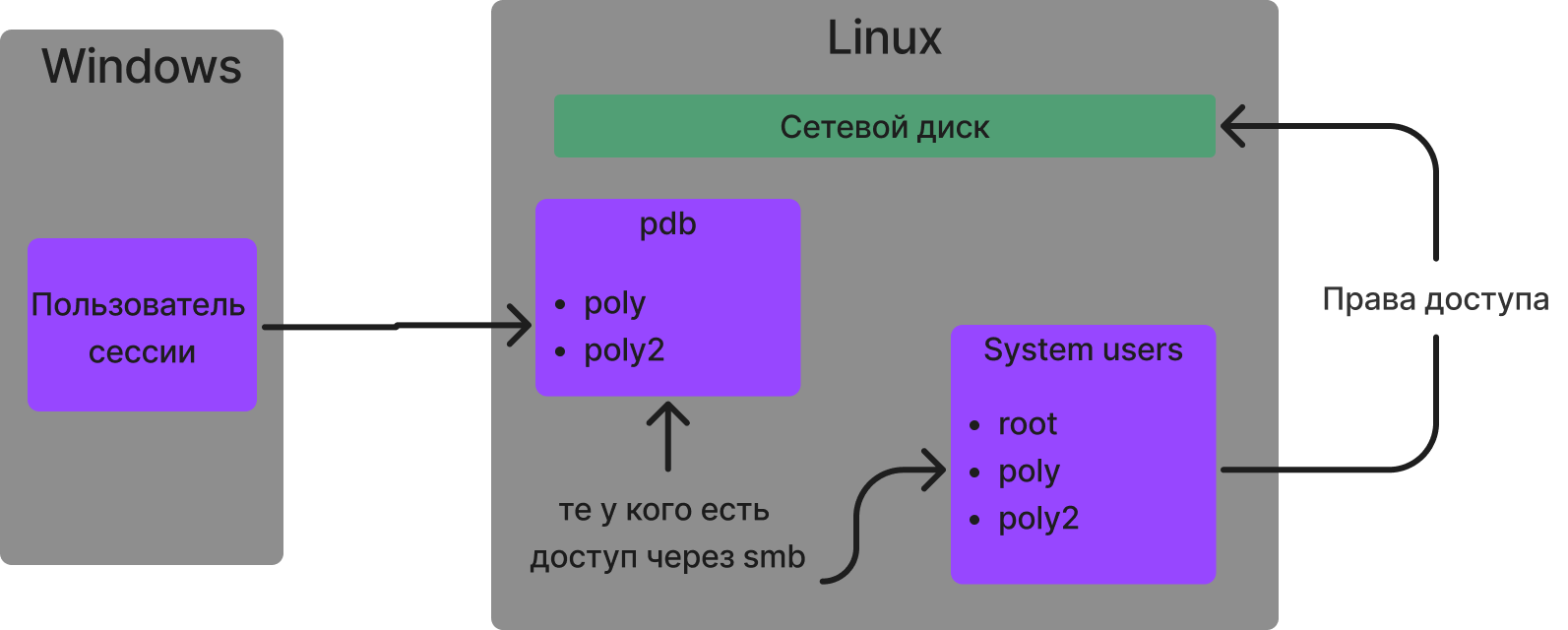


Схема сетевого взаимодействия



Когда Windows выступает в роли клиента, а Linux - в роли сервера в сетевой среде SMB, для авторизации происходит следующий процесс:

Пользователь на клиенте Windows пытается получить доступ к общему ресурсу на сервере Linux, используя путь UNC, например \\\linux-server\sharename.

Сервер Linux получает запрос на подключение от клиента и проверяет подлинность пользователя. Сервер проверяет, присутствует ли пользователь в файле /etc/passwd или во внешней системе аутентификации, такой как LDAP или Active Directory.

Если пользователь успешно прошел аутентификацию, сервер проверяет, имеет ли он право доступа к запрашиваемому ресурсу. Это определяется разрешениями, установленными для общей папки или файла на сервере Linux.

Если пользователь авторизован для доступа к ресурсу, сервер предоставляет клиенту доступ и разрешает доступ к запрашиваемому файлу или папке. Сервер Linux также отправляет ответ клиенту, подтверждающий, что соединение было установлено и пользователь авторизован.

После этого клиент Windows может получить доступ к общему ресурсу, как и к любому другому локально доступному файлу или папке.

Работа SMB-клиента в Linux.

Использовать утилиту smbclient. Отобразить доступные ресурсы на заданном smb-сервере

Smbclient – это инструмент командной строки, используемый для доступа к сетевым ресурсам SMB/CIFS на серверах под управлением Microsoft Windows или других операционных систем. Он позволяет пользователям выполнять различные операции, такие как просмотр и доступ к файловым ресурсам, службы печати и даже аутентификация в доменах Windows.

Инструмент smbclient является частью пакета Samba и предоставляет простой способ доступа и управления файлами и каталогами на удаленной машине Windows. Инструмент поддерживает широкий спектр команд, включая подключение к удаленному серверу, просмотр содержимого общих ресурсов, загрузку и выгрузку файлов и многое другое.

Smbclient также можно использовать для отправки и получения сообщений между системами Windows с помощью протокола Messaging Application Programming Interface (MAPI). Кроме того, его можно использовать для печати на серверах печати Windows, а также для управления принтерами в сети Windows.

Чтобы использовать smbclient, необходимо иметь базовые знания об инструментах командной строки и сетевых протоколах. Инструмент имеет широкий набор опций и параметров, которые позволяют пользователям настраивать взаимодействие с удаленными серверами. Вы можете указать множество параметров, включая параметры аутентификации, пути к общим ресурсам и каталогам и многое другое.

В целом, smbclient – это мощный и гибкий инструмент, который можно использовать для решения широкого круга задач сетевого администрирования.

Конечно, вот пример использования smbclient для подключения к файловому ресурсу Windows:

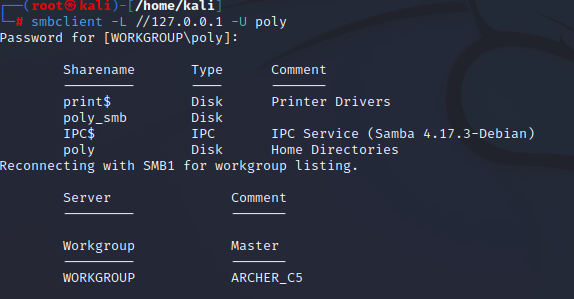
smbclient //192.168.1.100/share -U имя пользователя

Эта команда подключается к файловому ресурсу Windows, расположенному по IP-адресу 192.168.1.100, и запрашивает пароль для указанного имени пользователя.

После подключения вы можете использовать различные команды в smbclient для навигации и работы с файлами на удаленном ресурсе. Например:

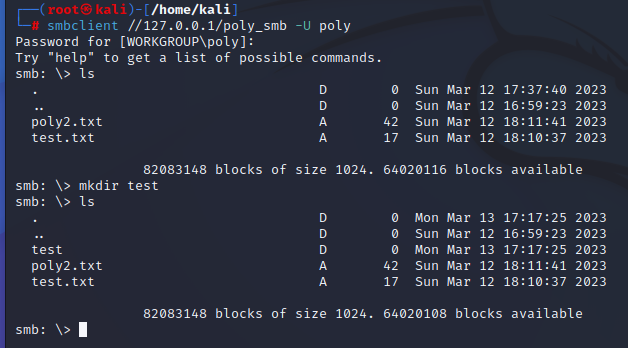
* **ls** выводит список содержимого текущего каталога на удаленном ресурсе
* **get** filename загружает файл с указанным именем с удаленного ресурса на локальную машину
* **put** filename загружает файл с указанным именем с локальной машины на удаленный ресурс
* **cd** directory изменяет текущий каталог на удаленном ресурсе
* **mkdir** directory создает новый каталог на удаленном ресурсе.
* Чтобы выйти из smbclient, просто введите **exit** или **quit**.

Попробуем подключиться с Linux клиента к Linux серверу:

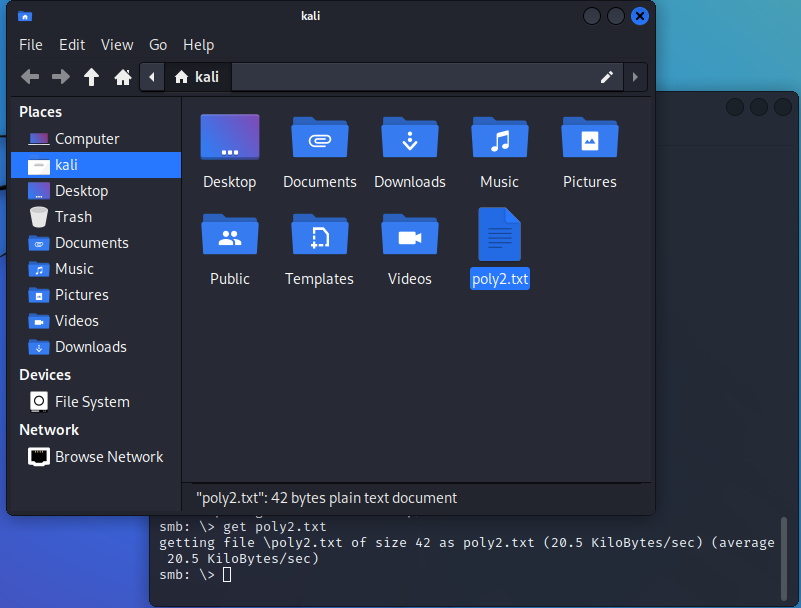


В выводе видно доступные ресурсы это диски: драйвер для принтера, директория, созданная для примера, рабочая директория пользователя poly, а также IPC - IPC$ — это скрытый ресурс в протоколе SMB (Server Message Block), который используется для межпроцессного взаимодействия между сетевыми компьютерами в сети на базе Windows. Он позволяет удаленным клиентам выполнять команды, передавать файлы и выполнять другие операции на сетевом компьютере. Название ресурса "IPC$" означает "Interprocess Communication Share", и он всегда присутствует на компьютерах Windows, даже если не создано никаких других ресурсов. Общий ресурс IPC$ используется различными сетевыми протоколами, такими как SMB, DCOM и RPC, для связи между процессами и службами, запущенными на разных компьютерах в сети.

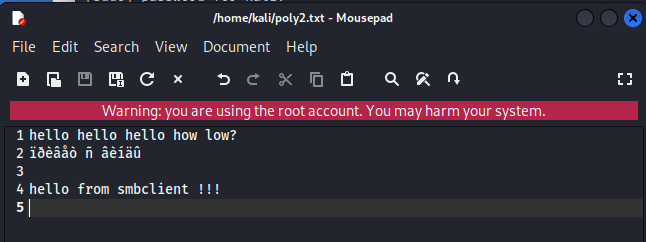
Для того чтобы выбрать к какому ресурсу, необходимо убрать флаг “-L”, который показывает доступных ресурсов:

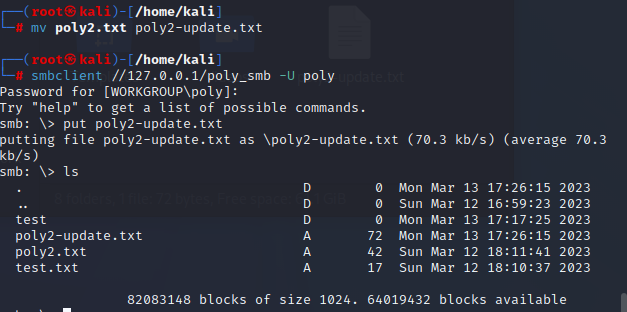


Для примера создали папку в корневой директории, попробуем загрузить файл poly2.txt в директорию, из которой было подключение по smbclient:

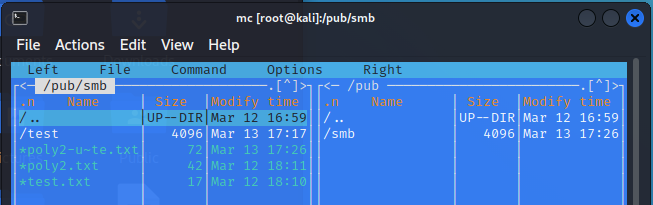


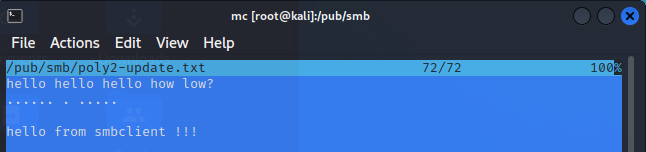
Теперь добавим информацию в файл, переименуем его и загрузим его обратно в ресурс:





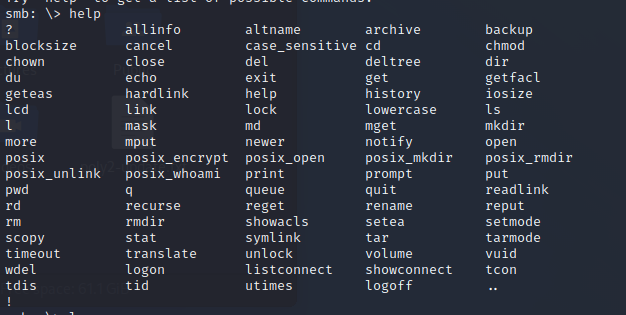
Перейдем в директорию в /pub/smb и посмотрим содержимое файлов:





Как видим, появился новый файл, который мы загрузили с помощью команды put.

Посмотрим основные команды интерфейса smbclient, посмотреть команды можно с помощью help:

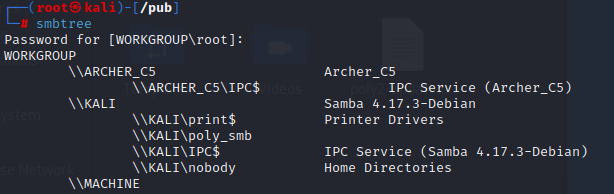


Попытаться выполнить обзор smb-ресурсов командой smbtree. Объяснить условия, необходимые для обзора.

Команда smbtree — это утилита Linux/Unix, используемая для составления списка всех доступных SMB-долей в сети. Она рекурсивно запрашивает SMB-серверы, чтобы отобразить топологию сети и доступные SMB-доли в древовидном формате.

Чтобы использовать smbtree, откройте терминал и выполните следующую команду: smbtree. Это выведет древовидное представление всех доступных SMB-долей в сети, а также имя сервера или IP-адрес.

Например:



Этот вывод показывает, что в сети "WORKGROUP" есть три сервера, "ARCHER\_C5", "KALI" и "MACHINE", каждый со своим набором общих ресурсов.

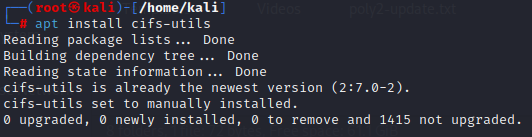
По умолчанию smbtree использует учетные данные текущего пользователя для аутентификации на серверах SMB. Однако также можно указать имя пользователя и пароль для аутентификации с помощью флага -U.

Использование команды mount.cifs.

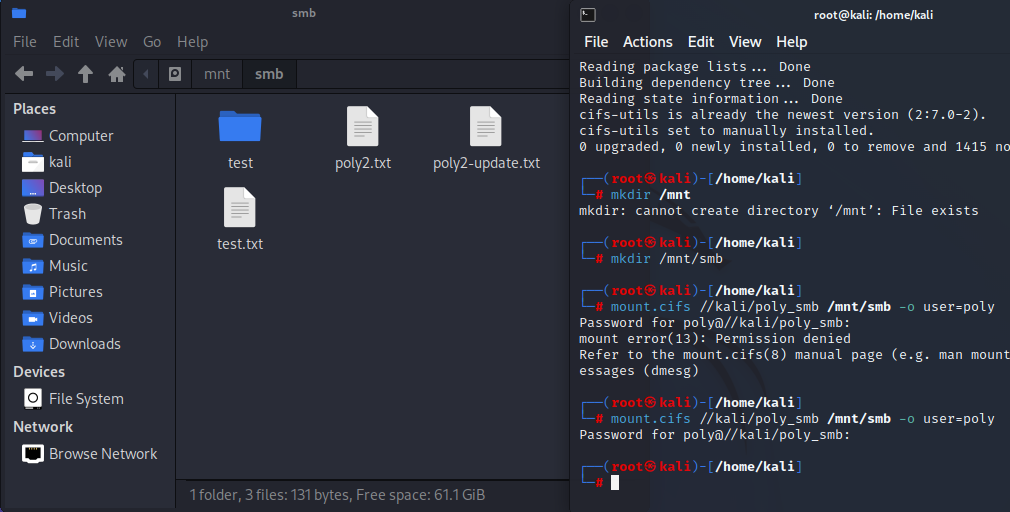
**mount.cifs** - это команда Linux, используемая для монтирования удаленных файловых систем, которые используют протокол Common Internet File System (CIFS). Она позволяет системам Linux получать доступ к общим сетевым дискам и каталогам на Windows и других системах, поддерживающих протокол SMB. Команда используется для указания имени удаленного сервера, имени общего ресурса и локальной точки монтирования. После монтирования удаленные файлы и каталоги могут быть доступны и управляемы как любая локальная файловая система.

Для выполнения команды mount.cifs требуются привилегии root, и, как правило, в системе Linux должен быть установлен пакет cifs-utils. Команда поддерживает различные опции, такие как указание имени пользователя и пароля для доступа к удаленному ресурсу, установка разрешений на файлы и включение поддержки старых версий протокола SMB.

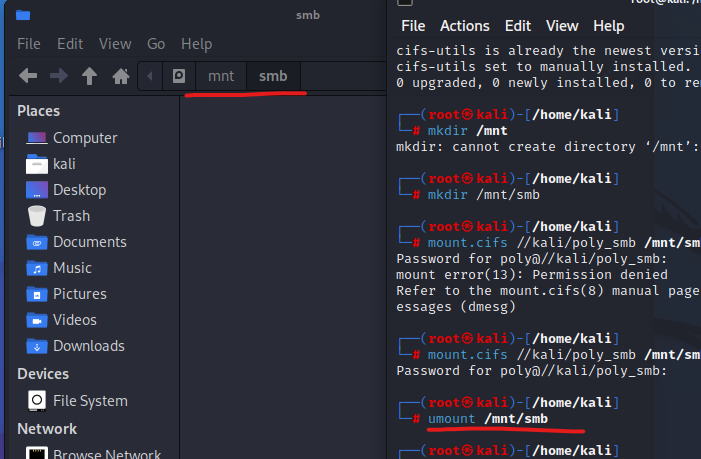
Установим пакет cifs-utils.



Создадим директии /mnt/smb в которую будем монтировать сетевой диск. Далее выполним команду монтирования с явным указанием пользователя:

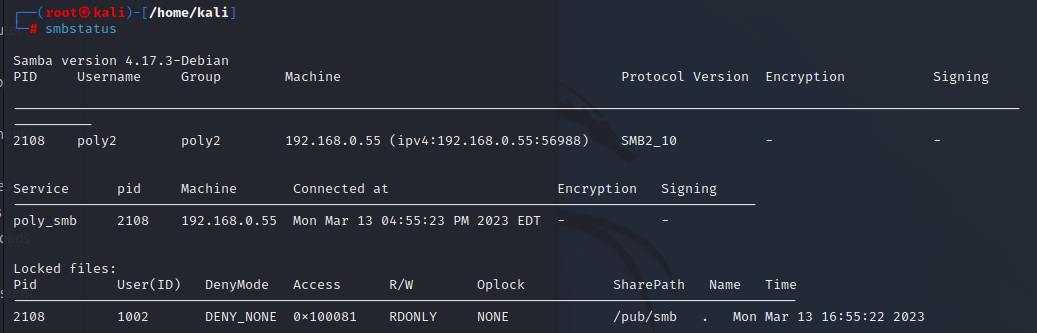


После конца работы необходимо размонтировать образ, выполнить это можно с помощью команды **umount <mount\_path>**, перед операцией размонтирования необходимо проверить что никакие файлы образа не используются:

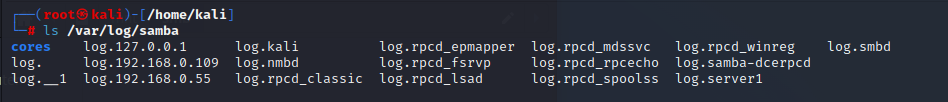


Использование команды smbstatus

**smbstatus** – это утилита командной строки в системах на базе Linux/Unix, которая позволяет пользователям просматривать информацию о текущих соединениях Samba и открытых файлах. Она отображает список активных соединений, включая пользователей, службы, к которым они обращаются, и открытые файлы. Она также предоставляет информацию о состоянии сервера Samba и его ресурсов. Smbstatus может быть полезен для мониторинга активности Samba и устранения проблем с соединениями Samba и общим доступом к файлам.

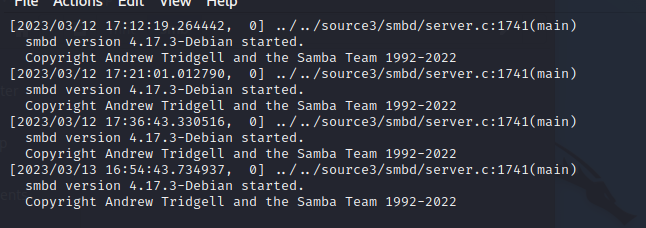


Рассмотрим протоколы демонов samba, которые хранятся в каталоге /var/log/samba.



На сервере Samba файлы log.smbd и log.nmbd содержат информацию об активности сервера и возникающих ошибках или предупреждениях.

**log.smbd**: Этот файл журнала содержит информацию, связанную с демоном Samba (smbd), который отвечает за совместное использование файлов и принтеров. Он содержит подробную информацию о подключениях пользователей, файловых операциях, печати и т.д.



**log.nmbd**: Этот файл журнала содержит информацию, связанную с демоном сервера имен NetBIOS (nmbd), который отвечает за предоставление услуг разрешения имен NetBIOS. Он содержит подробную информацию о регистрации и поиске имен, активности просмотра и т.д.

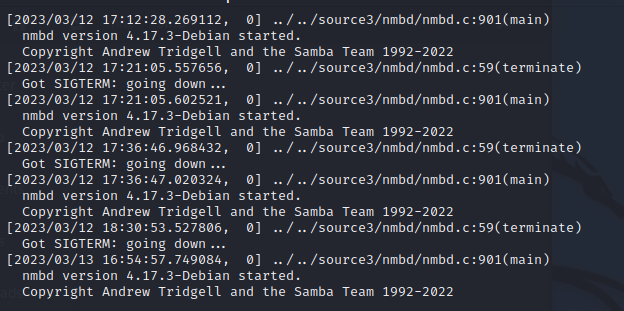
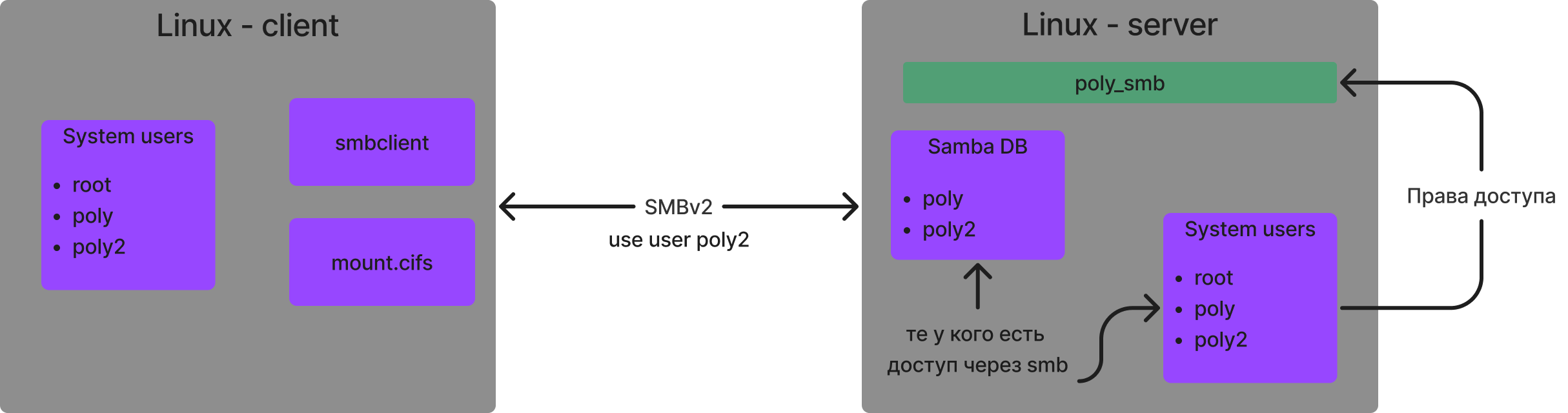


Схема сетевого взаимодействия (SMB, клиент –Linux, сервер - Linux)



Когда Linux является клиентом и сервером в сети SMB, процесс авторизации аналогичен процессу авторизации клиента Windows и сервера Linux. Клиент отправляет запрос на аутентификацию на сервер, а сервер в ответ запрашивает учетные данные клиента. Клиент отправляет свои учетные данные на сервер, который затем проверяет их по своей базе данных пользователей. Если учетные данные действительны, сервер предоставляет клиенту доступ к запрашиваемым ресурсам.

В Linux процесс авторизации включает проверку учетных данных клиента по учетным записям пользователей на сервере. Сервер поддерживает базу данных пользователей, которая включает в себя учетные записи пользователей и связанные с ними учетные данные. Когда клиент пытается получить доступ к ресурсу на сервере, сервер проверяет учетные данные клиента по базе данных пользователей, чтобы определить, имеет ли клиент право доступа к ресурсу.

Чтобы облегчить процесс аутентификации, клиенту и серверу необходимо согласовать набор протоколов безопасности и алгоритмов шифрования, которые будут использоваться для защиты аутентификационных данных при передаче по сети. Протокол SMB включает несколько механизмов безопасности, в том числе аутентификацию паролем, аутентификацию NTLM и аутентификацию Kerberos. Конкретные используемые механизмы безопасности зависят от конфигурации сервера и клиента и политики безопасности, действующей в сети.

Конфигурирование Samba-сервера утилитой net.

Рассмотреть возможности и особенности утилиты net для локального и удаленного конфигурирования.

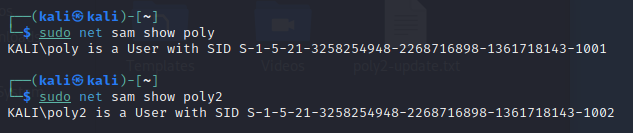
Утилита net в Linux – это мощный инструмент командной строки, используемый для локальной и удаленной настройки различных сетевых служб, включая Samba. Некоторые из возможностей и функций утилиты net для настройки сервера Samba в Linux следующие:

* Присоединение к домену и выход из него: Команда **net join** используется для присоединения Linux Samba-сервера к домену Windows, а **net leave** - для выхода из домена.
* Добавление и удаление пользователей и групп: **net user** используется для создания, изменения или удаления пользователей в Samba, а **net group** - для управления группами.
* Управление общими ресурсами: **net share** используется для создания, изменения или удаления общих ресурсов в Samba.
* Управление принтерами: **net printer** используется для управления принтерами в Samba.
* Управление доменными трастами: **net rpc trustdom** используется для управления доменными трастами.
* Управление аутентификацией: **net auth** используется для настройки параметров аутентификации в Samba.
* Просмотр информации о сервере Samba: **net serverinfo** используется для просмотра информации о сервере Samba.
* Список активных сессий: **net sessions** используется для просмотра списка активных сессий на сервере Samba.
* Управление разрешениями на файлы и каталоги: **net fileacl** и **net shareacl** используются для управления разрешениями на файлы и каталоги в Samba.

В целом, утилита net – это универсальный инструмент, который можно использовать для управления различными аспектами сервера Samba в Linux, как локально, так и удаленно.

Использовать информационные возможности утилиты net.

Команда **net sam show** – это инструмент, используемый в Samba, реализации сетевого протокола SMB/CIFS с открытым исходным кодом. Она позволяет отобразить информацию об учетных записях и группах пользователей, хранящихся в базе данных Samba, а также их свойства, такие как полные имена, SID и Unix ID. Эта информация обычно используется для управления учетными записями пользователей и групп в домене Samba, а также для интеграции Samba с другими службами каталогов, такими как LDAP.

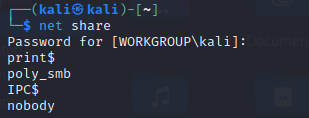


Посмотрим данные пользователей линукс с помощью команды **id**:

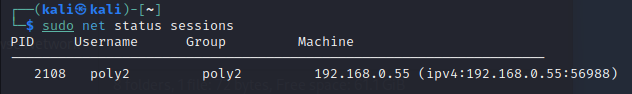


Видно, что пользовательский uid в линукс поставился в конец sid отображенный с помощью команды net.

В Linux Samba команда **net share** используется для перечисления и управления общими ресурсами Samba на сервере Samba. Она отображает информацию о каждом ресурсе, включая имя ресурса, путь и разрешения доступа. Команда может использоваться для создания новых, изменения существующих и удаления общих ресурсов. Она также может быть использована для отображения списка подключенных клиентов, активных ресурсов и их пользователей, а также других подробностей о сервере Samba.



Команда **net status sessions** используется на сервере Linux Samba для отображения информации о текущих сессиях, установленных между клиентами и сервером Samba. Она показывает такие детали, как имя пользователя, IP-адрес и время соединения для каждого активного сеанса. Это может быть полезно для мониторинга использования сервера и устранения проблем с подключением.



В момент выполнения по SMB протоколу был подключен Windows Server от имени poly2.

Команда **net getlocalsid** – это команда Linux Samba, которая используется для получения идентификатора безопасности (SID) локальной системы. Эта команда часто используется в сочетании с другими командами и утилитами Samba для управления и настройки серверов Samba в сетевой среде Windows.



Если посмотреть выше на вывод команды **net sam show <user>**, то увидим, что getlocalsid это SID данного сервера, а uid это конкретный пользователь.

Команда **net lookup <computer\_name>** — это команда, которая позволит по имени хоста получить его IP-адрес.



Взаимодействие на базе протокола NFS

Сетевая файловая система NFS в Linux.

NFS (Network File System) – это протокол распределенной файловой системы, разработанный компанией Sun Microsystems, который позволяет пользователю получать доступ к файлам на удаленном компьютере или сервере и обмениваться ими по сети. NFS позволяет нескольким клиентским машинам совместно использовать одни и те же файлы и каталоги по сети, как если бы они были непосредственно подключены к локальной машине. Он широко используется в операционных системах на базе Unix и Linux для совместного использования файлов и является альтернативой протоколу SMB/CIFS, используемому в среде Windows.

Основные компоненты, необходимые для работы NFS

Для установки NFS необходимо ввести apt install nfs-kernel-sever. Для установки клиента выполнить: apt install nfs-common.

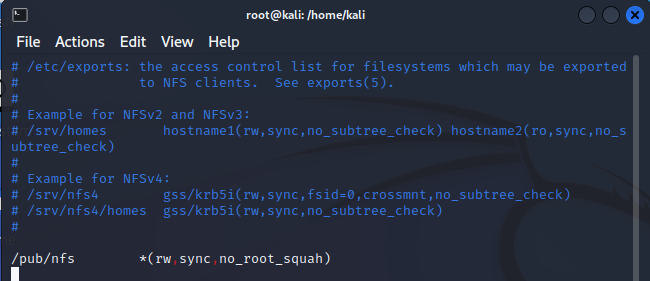
Основные компоненты, входящие в пакеты nfs-kernel-server и nfs-common, следующие:

* nfsd: Это основной процесс сервера NFS, который работает на сервере и обрабатывает запросы клиентов на доступ к файлам.
* mountd: Этот процесс отвечает за монтирование и размонтирование файловых систем NFS на стороне клиента.
* rpc.nfsd: Это демон, который обеспечивает функциональность сервера NFS для ядра Linux.
* rpc.mountd: Этот демон отвечает за управление монтированием NFS на стороне сервера.
* rpc.lockd: Это демон, который предоставляет услуги блокировки для NFS, обеспечивая безопасный и скоординированный доступ к файлам.
* rpc.statd: Этот демон отвечает за обработку уведомлений о сбоях и восстановление клиентов NFS.
* showmount: это утилита командной строки, используемая для отображения текущего экспорта NFS на сервере.
* rpcinfo: это еще одна утилита командной строки, которая позволяет запросить RPC-сервис, чтобы получить информацию о нем и версии протокола, который он поддерживает.
* nfsstat: Эта утилита предоставляет информацию о состоянии сервера и клиента NFS.
* nfs-common: Этот пакет предоставляет функциональность клиента NFS, которая позволяет системе Linux монтировать удаленные файловые системы NFS.

Выделение каталога в общий доступ

Чтобы открыть общий доступ к каталогу с помощью NFS в Linux, необходимо выполнить следующие шаги:

1. Экспортируйте каталог, к которому вы хотите предоставить общий доступ, путем редактирования файла /etc/exports на серверной машине. Например, если вы хотите совместно использовать каталог /pub/nfs/, добавьте следующую строку в файл /etc/exports:

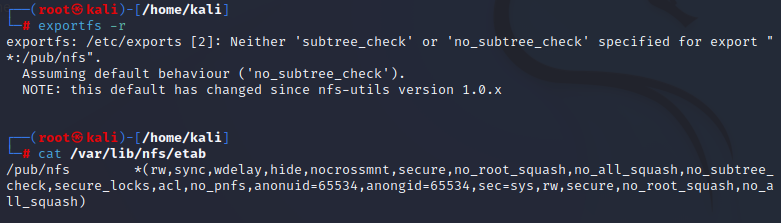


Символ "\*", стоящий перед открывающейся скобкой означает, что нет ограничений по ip или domain name.

1. Опция "rw" позволяет клиенту читать и записывать в общий каталог, опция "sync" обеспечивает запись изменений на диск перед подтверждением клиенту, а опция "no\_subtree\_check" отключает проверку поддеревьев для этого ресурса. Так же есть множество других опций:

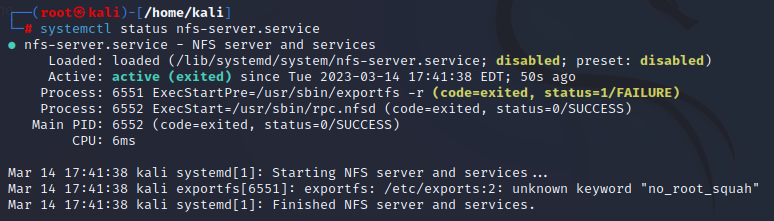
* rw: разрешает доступ на чтение и запись к общему каталогу
* sync: требует записи изменений на диск перед подтверждением операции записи.
* no\_subtree\_check: отключает проверку поддеревьев для этого общего ресурса, что может повысить производительность
* no\_root\_squash: позволяет пользователям root на клиенте иметь доступ на уровне root к общему каталогу.
* insecure: позволяет использовать непривилегированные порты для связи NFS, что может улучшить совместимость, но снизить безопасность.
* ro: разрешает только доступ на чтение к общему каталогу
* all\_squash: передает все запросы клиента анонимному пользователю и группе, что может помочь с безопасностью, но ограничивает функциональность.
* anonuid: устанавливает UID анонимного пользователя для общего ресурса
* anongid: устанавливает GID анонимного пользователя для общего ресурса

1. Обновим таблицу экспорта в ядре с помощью команды exportfs -r. В файле /var/lib/nfs/etab содержится полный дамп конфигурационного файла:



1. Перезапустите службу сервера NFS, чтобы применить изменения, внесенные в файл /etc/exports:



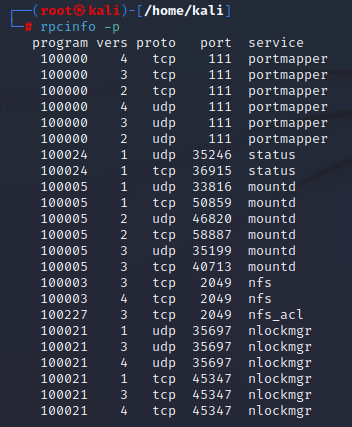


Запуск NFS.

После перезагрузки nsf-server посмотрим процессы, которые были запущены:



Для того, чтобы клиент знал какой порт прослушивает NSF сервер используется порт-маппер, который работает на 111 порту. Для того, чтобы вывести справку о текущих протоколах и портах можно воспользоваться командой rpcinfo -p:



Как видим, NFS сервер работает на 2049 порту. Причем по TCP и UDP.

Монтирование сетевой файловой системы

На клиентской машине создайте каталог точки монтирования, куда вы хотите смонтировать общий катал, например, /mnt/nfs/.

Смонтируйте общий каталог с помощью команды mount:

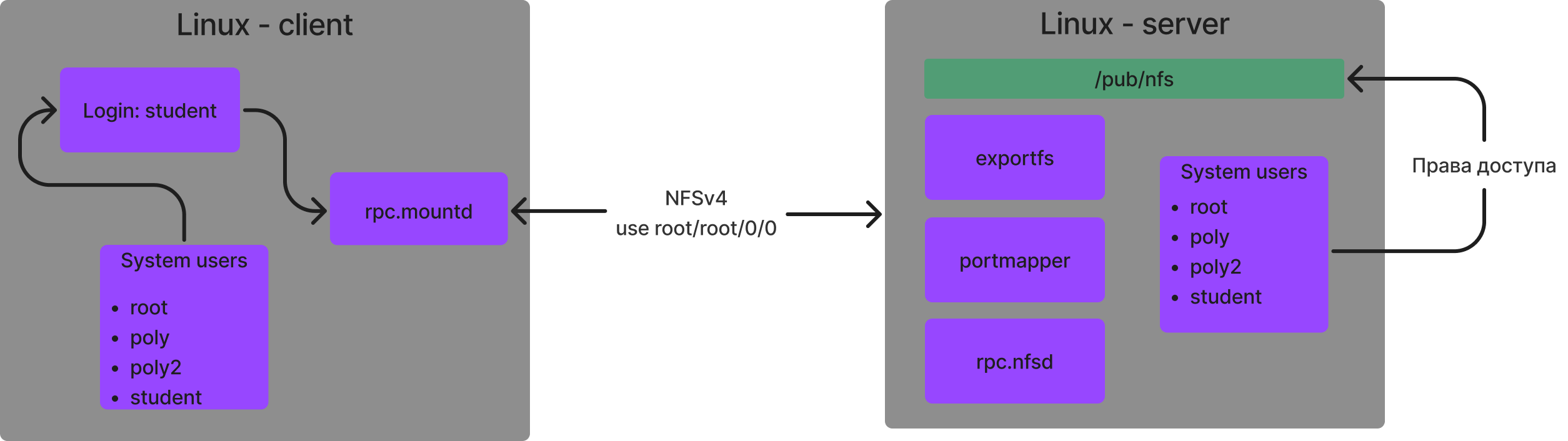


Посмотрим содержимое директории, в которое было выполнено монтирование, а также общей папке:



Как так в другой консоли я смонтировал от пользователя student и создал новый файл от его имени, то в директории указан владелец и группа student.

Схема сетевого взаимодействия



Соединение по протоколу NFS устанавливается между клиентом и сервером Linux следующим образом:

1. Клиент посылает запрос на монтирование на сервер, указывая файловую систему, которую он хочет смонтировать.
2. Сервер проверяет запрос клиента по списку экспорта в файле /etc/exports, чтобы убедиться, что клиент авторизован для монтирования запрашиваемой файловой системы.
3. Если клиент авторизован, сервер отвечает на запрос монтирования и отправляет точку монтирования и дескриптор файла клиенту.
4. Затем клиент создает точку монтирования на своей файловой системе и устанавливает соединение с сервером с помощью протокола Network Lock Manager (NLM).
5. После установления соединения клиент может читать и записывать файлы в файловой системе сервера, как если бы это были локальные файлы.
6. Когда клиент завершает работу с файловой системой, он размонтирует ее, что приводит к разрыву соединения с сервером.

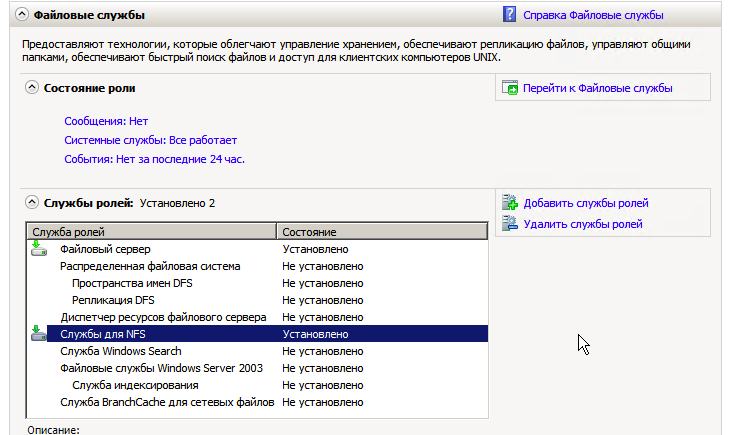
В целом, соединение по протоколу NFS устанавливается в процессе авторизации, монтирования, установления соединения и размонтирования.

Использование в Windows Server служб Service for NFS для организации сетевого доступа по протоколу NFS.

Настроить файл-сервер в Windows Server

Перед началом работы на машине SERVER1 был настроен как контроллер домена poly.com.

Откроем "Диспетчер сервера" и добавим роль файлового сервера, а так же службы NFS:

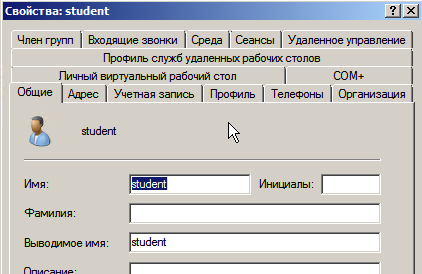


После установки появятся две оснастки: "Диспетчер ресурсов файлового сервера" и "Службы для NFS".

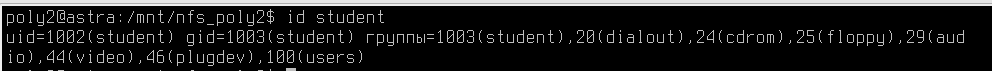
Рассмотреть средства согласования учетных записей и свойства службы отображения имен.

Для отображения пользователей linux на пользователей Windows используется механизм согласования имен uid и gid к SID.

Для примера создадим пользователя student:



Так как на Linux такой пользователь уже существует, его создавать не надо, но нужно узнать его UID и GID:



Видим, что uid=1002, а gid=1003, для того чтобы сопоставить пользователя на Windows откроем оснастку “Редактирование ADSI” и там в пользователях найдем нужно и откроем его свойства:



Необходимо найти атрибуты, отвечающие за uid и gid на linux – это gidNumber и uidNumber. Значение их будет таким же, как и у пользователя на linux:





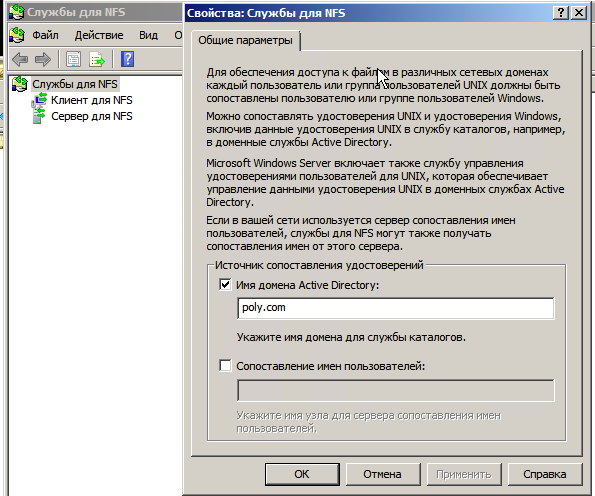
Оснастка «Службы для NFS»

Служба "Server for NFS" – это компонент Windows Server, который позволяет серверу работать в качестве сервера NFS, предоставляя доступ к файлам и каталогам клиентам NFS. Свойства службы "Server for NFS" можно настроить с помощью следующих опций:

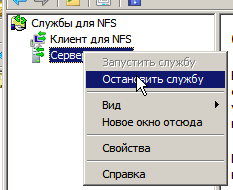
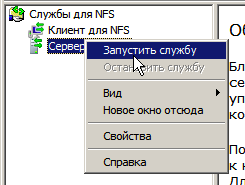
1. **Конфигурация сервера**: В этом разделе можно настроить параметры NFS сервера, включая NFS домен сервера, UID и GID по умолчанию для анонимного доступа, а также порты NFS, используемые сервером.
2. **Общие папки NFS**: В этом разделе вы можете указать, какие папки на сервере являются общими с помощью NFS. Вы можете указать путь к папке, имя общего ресурса, разрешения и сопоставление между разрешениями Windows и Unix.
3. **Сопоставление имен пользователей**: Этот раздел позволяет сопоставить пользователей и группы Windows с пользователями и группами Unix. Вы можете указать отображение по умолчанию для всех пользователей и групп или создать индивидуальные отображения для определенных пользователей и групп.
4. **Параметры клиента**: В этом разделе можно настроить параметры для клиентов NFS, которые подключаются к серверу, включая версию NFS по умолчанию, размер буфера чтения и записи NFS, а также максимальное количество одновременных соединений NFS.
5. **Производительность**: Этот раздел позволяет оптимизировать производительность сервера NFS путем настройки таких параметров, как количество потоков NFS и таймауты чтения и записи NFS.
6. **Ведение журнала**: Этот раздел позволяет настроить параметры ведения журнала для сервера NFS, включая уровень журнала и расположение файлов журнала.

В целом, служба "Сервер для NFS" предоставляет полный набор опций для настройки сервера NFS на Windows Server, позволяя ему интегрироваться с системами на базе Unix и предоставлять услуги совместного доступа к файлам клиентам NFS.

Возвращаясь к предыдущему пункту: теперь в службе на NFS, надо указать на сервер согласования имен, так как это и есть наш контроллер, то укажем его:

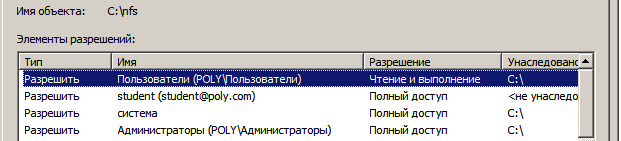


Для того, чтобы изменения вступили в силу, необходимо перезагрузить службу:

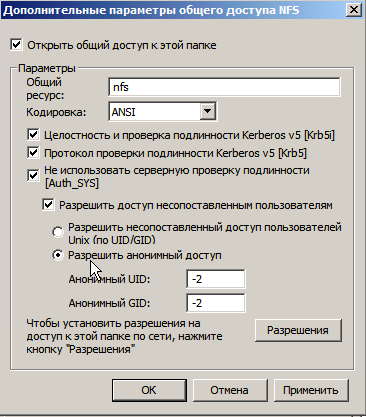
 

На Windows-сервере создать каталог, определить разрешения доступа NTFS. Средствами эксплорера в свойствах каталога настроить параметры экспорта каталога по NFS.

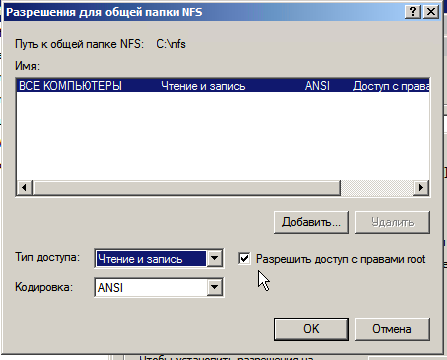
Создадим каталог C:\nfs, откроем настройки разрешения доступа NTFS и откроем полный доступ для Администратора и student, а для остальных чтени и исполнение.



На вкладке «Совместный доступ NFS» настроим разрешения NFS. Разрешим не сопоставленный доступ в варианте «Разрешить анонимный доступ», при таком разрешении учетные записи, которым не сопоставлена учетная запись Windows будут получать UID, GID равный «-2».



На вкладке «Разрешения» дадим доступ на подключение всем компьютерам для записи и чтения, а также подключаться с root-правами:



На Windows-сервере с помощью консоли администрирования перезапустить сервер NFS. Командой showmount отобразить состояние ресурсов NFS.

Команда showmount используется в среде NFS (Network File System) для отображения списка клиентов, смонтировавших общий каталог NFS, или для отображения списка общих каталогов на сервере NFS.

При использовании опции -e, showmount отобразит список каталогов, находящихся в общем доступе на сервере NFS, а при использовании опции -a, он отобразит список клиентов, которые в настоящее время смонтировали любой общий каталог с указанного сервера NFS.

Некоторые общие опции для showmount следующие:

-e: отображает список каталогов, совместно используемых NFS-сервером.

-a: отображает список клиентов, которые в настоящее время смонтировали любой общий каталог с указанного сервера NFS.

-d: отображает список каталогов на сервере, которые в настоящее время используются клиентами NFS.

Для демонстрации работы посмотрим какие есть каталоги на Линукс сервере, который использовался ранее:



Папки доступные на текущем Windows сервере:



Ознакомиться с командой nfsshare для управления ресурсами NFS.

Команду nfsshare можно использовать для управления ресурсами NFS на сервере Windows Server. Ниже приведены примеры использования этой команды:

1. Предоставление общего доступа к папке в качестве общего ресурса NFS:

nfsshare -o rw,crossmnt C:\share /exports/share

он предоставит общий доступ к папке C:\share как к NFS ресурсу с параметрами rw и crossmnt, и смонтирует ее на удаленном NFS клиенте по адресу /exports/share.

1. Перечислите все общие ресурсы NFS на сервере:

nfsshare -o show

1. Удалите общий ресурс NFS:

nfsshare -o unshare /exports/share

1. Изменение параметров существующего общего ресурса NFS:

nfsshare -o rw,ro /exports/share

Это изменит параметры общего ресурса NFS, смонтированного в /exports/share, на rw и ro.

Обратите внимание, что команду nfsshare нужно выполнять с правами администратора на сервере Windows Server.

Схема сетевого взаимодействия (NFS, Клиент – Linux, Сервер – Windows, сопоставленный доступ)

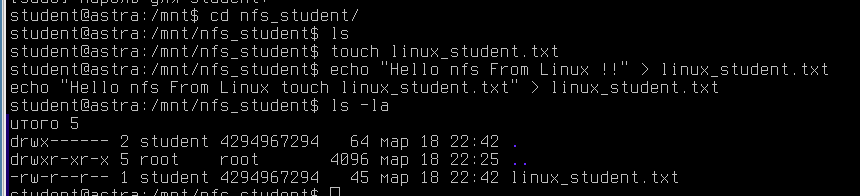
Смонтируем на клиенте Linux от пользователя student:



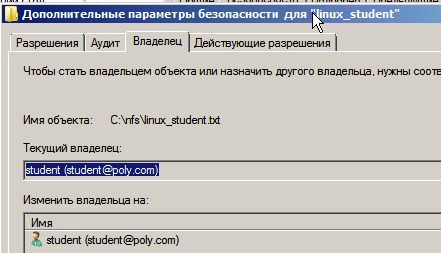
Посмотрим на файл сервере какие подключения появились:



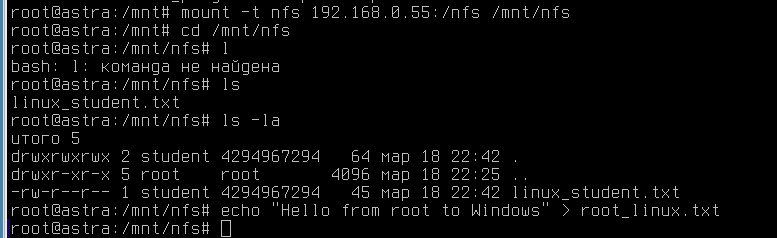
Как видим каталог успешно смонтировался. Создадим на клиенте файл и посмотрим на сервере его свойства:

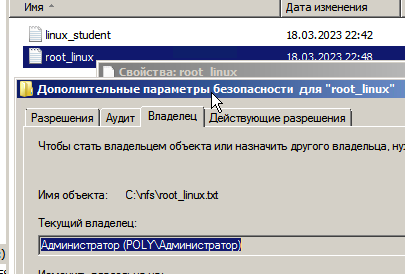


Как видим сопоставление имен отработало корректно и файл принадлежит пользователю student

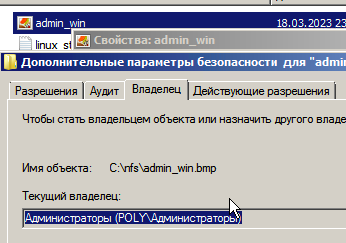


На Windows сервере мы так же сопоставляли root и Админстратора, попробует смонтировать образ от root пользователя и создать файл с его правами:

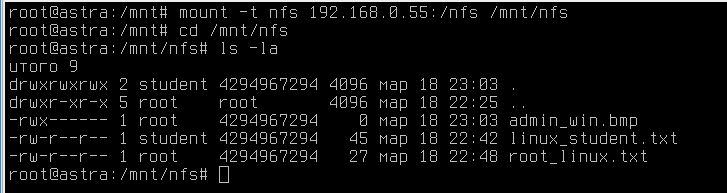




Сопоставление прошло успешно. Попробуем создать на сервере от имени Администратора точечный рисунок и посмотрим на клиентах владельца и разрешения:



Посмотрим на клиенте и увидим, что владельцем рисунка является root:

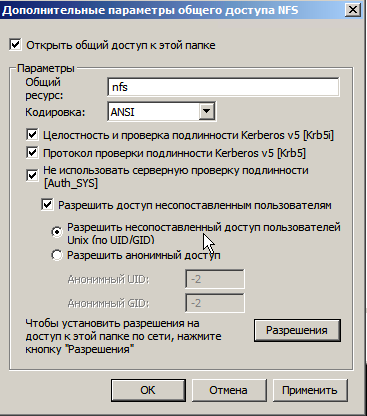


Для того, чтобы закончить работу с NFS сервером необходимо размонтировать его, как и раньше с помощью команды umount. Так же необходимо закрыть все файлы и каталоги, которые используют этот смонтированный образ:



Реализовать несопоставленный доступ.

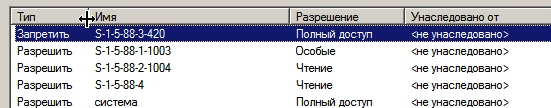
Для несопоставленного доступа необходи в свойствах совместного доступа NFS поменять выбор с “Разрешить анонимный доступ” на “Разрешить несопоставленный доступ пользователям UNIX (по UID/GID)”.



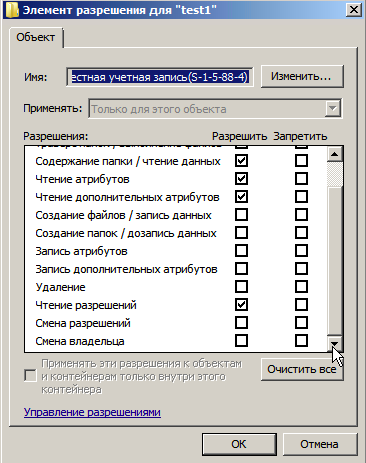
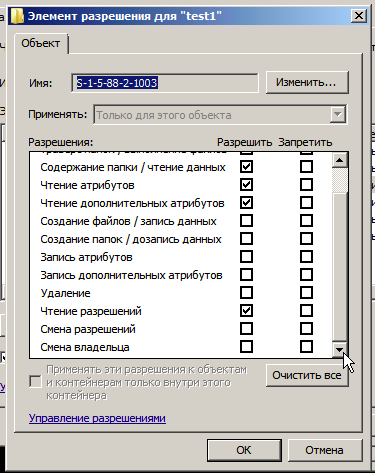
Применяем изменения и попробуем смонтировать образ на Linux клиенте от пользователя student:

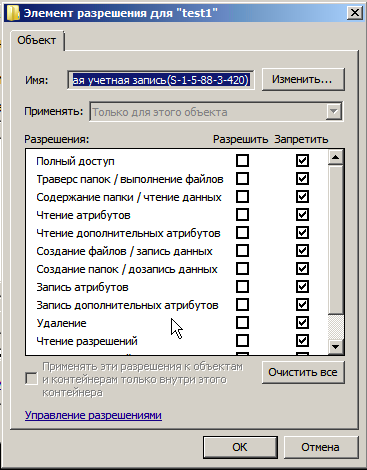
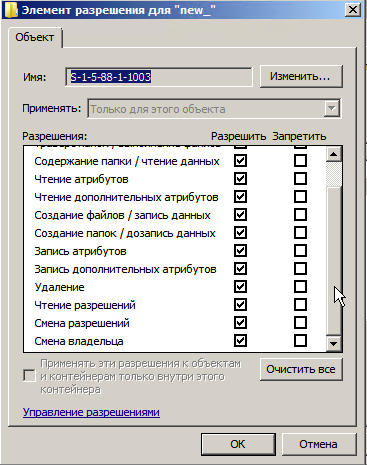


Монтирование прошло успешно, так же создали файл test.txt, посмотрим его права доступа на сервере:



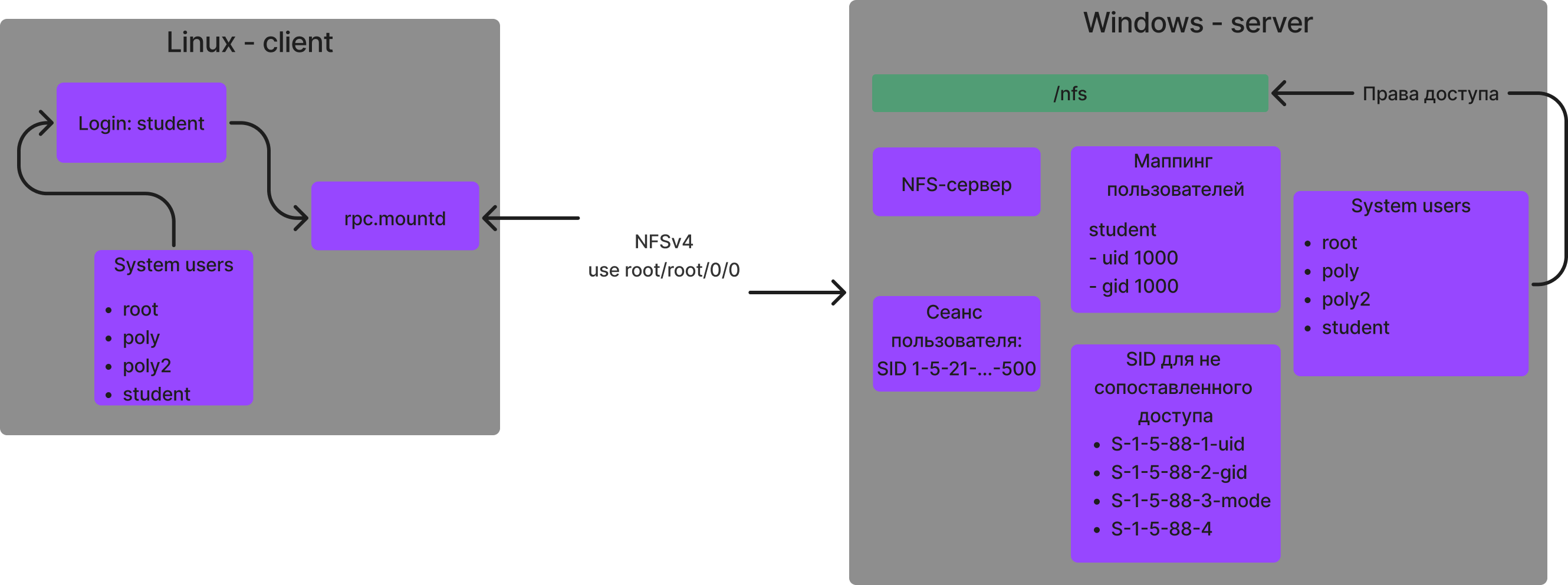
Со следующими правами:



* SID владельца на основе UID (например, "S-1-5-88-1-<uid>")
* SID группы на основе GID (например, "S-1-5-88-2-<gid>")
* SID режима, основанный на битах режима UNIX (например, "S-1-5-88-3-<mode>")
* Другой SID, постоянное значение (например, "S-1-5-88-4")

Схема сетевого взаимодействия (Linux-клиент, Windows-сервер):



Сопоставленный доступ:

1. Клиент Linux отправляет запрос серверу NFS на доступ к файловой системе с помощью команды mount. Запрос включает IP-адрес сервера NFS и имя общего каталога, который необходимо смонтировать.
2. Сервер NFS отправляет ответ клиенту Linux с информацией о монтировании, включая дескриптор файла и идентификатор файловой системы.
3. Клиент Linux отправляет запрос на сервер Windows для сопоставления идентификаторов пользователей и групп в стиле UNIX с их эквивалентами в Windows, используя службу Name Mapping.
4. Сервер Windows отправляет ответ клиенту Linux с сопоставлениями пользователей и групп.
5. Теперь клиент Linux может получить доступ к общему каталогу на сервере Windows, используя протокол NFS, с соответствующими сопоставлениями пользователей и групп.

Не сопоставленный доступ:

1. Клиент Linux отправляет запрос на монтирование ресурса NFS с сервера Windows, указывая IP-адрес или имя хоста сервера и имя ресурса.
2. Сервер Windows получает запрос и проверяет свою конфигурацию NFS, чтобы узнать, разрешено ли монтировать указанный ресурс указанному клиенту. Если ресурс разрешен, сервер отправляет клиенту ответ, указывающий на то, что ресурс был успешно смонтирован.
3. Клиент Linux получает ответ от сервера и создает точку монтирования в своей локальной файловой системе для ресурса NFS.
4. Клиент Linux отправляет запросы на чтение и запись на сервер Windows по протоколу NFS для доступа к файлам на смонтированном ресурсе.
5. Сервер Windows получает запросы на чтение и запись и обрабатывает их, читая или записывая данные в файловую систему на сервере.
6. Сервер Windows отправляет запрошенные данные обратно клиенту Linux по протоколу NFS.
7. Клиент Linux получает данные и обрабатывает их, либо отображая их пользователю, либо записывая их в файл на локальной файловой системе.
8. Когда Linux-клиент заканчивает работу с смонтированным ресурсом, он отправляет запрос на размонтирование на сервер Windows, чтобы отключиться от ресурса.
9. Сервер Windows получает запрос на размонтирование и удаляет смонтированный ресурс из файловой системы на сервере.

Использование NFS-клиента Windows

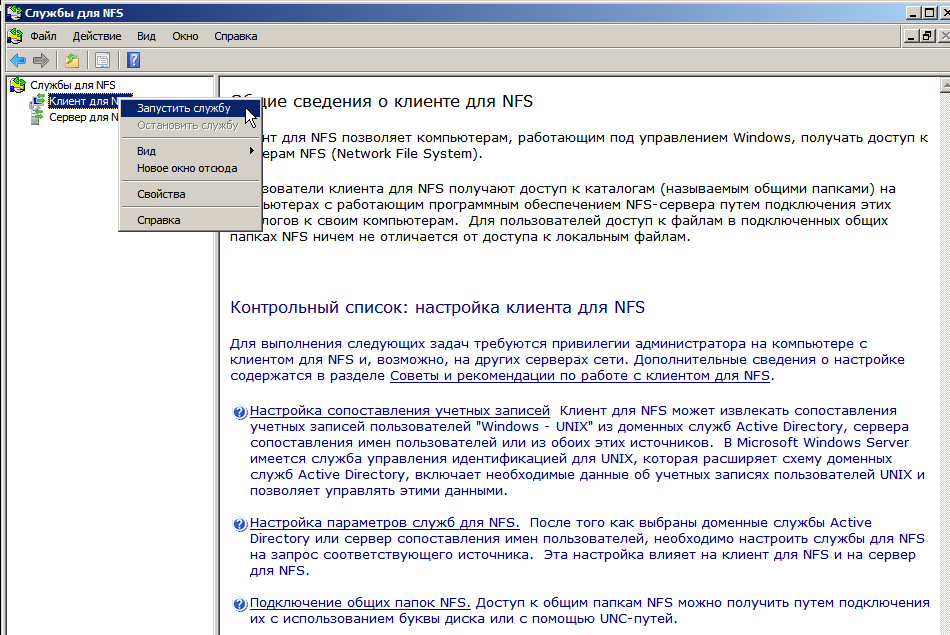
Служба "Client for NFS" - это компонент операционных систем Windows, который позволяет компьютерам Windows получать доступ к файлам и каталогам, расположенным на серверах NFS. Некоторые свойства службы "Client for NFS" следующие:

* Поддержка протоколов NFSv3 и NFSv4
* Сопоставление идентификаторов пользователей для разрешения имен пользователей и групп между средами Windows и NFS
* Поддержка вариантов безопасности NFS, включая AUTH\_SYS и KRB5 (Kerberos версии 5)
* Поддержка монтирования общих ресурсов NFS с помощью команды mount или функции Map Network Drive в проводнике Windows
* Поддержка разрешений и прав собственности на файлы и каталоги, а также символических и жестких ссылок
* Интеграция с Windows Active Directory для аутентификации и авторизации пользователей.

Эти свойства позволяют клиентам Windows беспрепятственно получать доступ к ресурсам NFS и обеспечивают совместную работу серверов Linux и Windows в гетерогенной сетевой среде.

Применить NFS-клиент на Windows для монтирования ресурса на NFS-сервере Linux.

Рассмотрим свойства службы “Клиент для NFS” и запустим её:

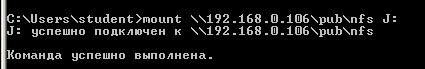


Служба "Client for NFS" – это компонент операционных систем Windows, который позволяет компьютерам Windows получать доступ к файлам и каталогам, расположенным на серверах NFS. Некоторые свойства службы "Client for NFS" следующие:

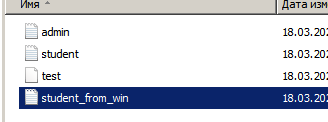
* Поддержка протоколов NFSv3 и NFSv4
* Сопоставление идентификаторов пользователей для разрешения имен пользователей и групп между средами Windows и NFS
* Поддержка вариантов безопасности NFS, включая AUTH\_SYS и KRB5 (Kerberos версии 5)
* Поддержка монтирования общих ресурсов NFS с помощью команды mount или функции Map Network Drive в проводнике Windows
* Поддержка разрешений и прав собственности на файлы и каталоги, а также символических и жестких ссылок.
* Интеграция с Windows Active Directory для аутентификации и авторизации пользователей.

Эти свойства позволяют клиентам Windows беспрепятственно получать доступ к ресурсам NFS и обеспечивают совместную работу серверов Linux и Windows в гетерогенной сетевой среде.

Применим команду mount для монтирования сетевого ресурса на сервере Linux:



Попробуем создать файл от пользователя student, для которого до этого настраивали сопоставление имен в ASDI:



Посмотрим свойства файла в атрибутах NFS:

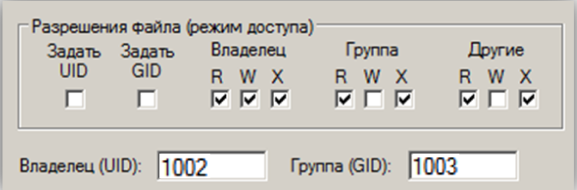
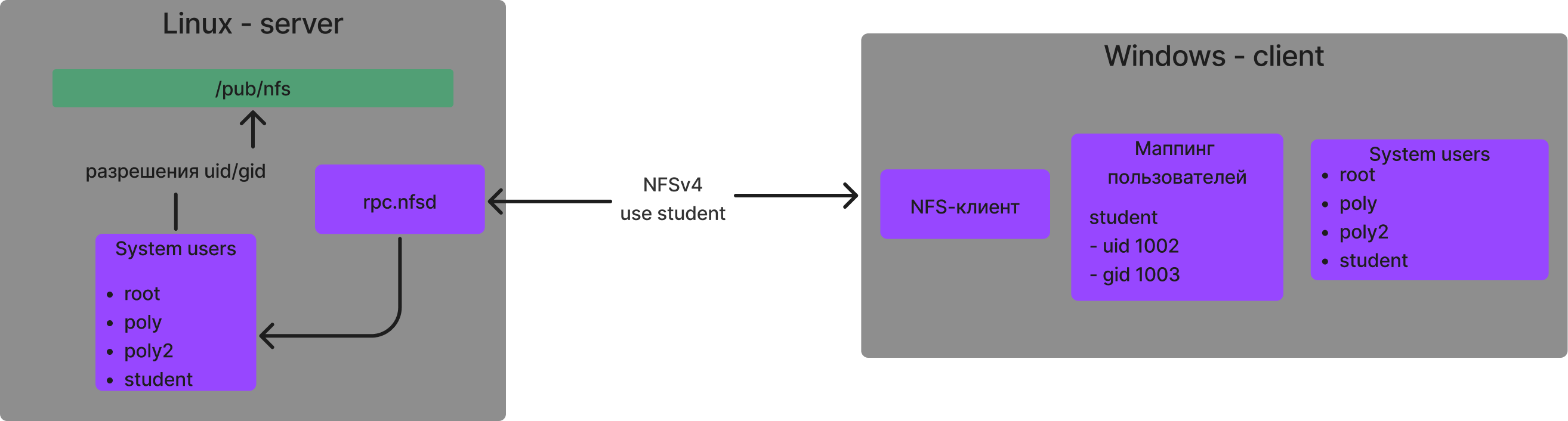


Схема взаимодействия NFS протокола (Windows – клиент, Linux – сервер)



1. Чтобы установить соединение, клиент NFS аутентифицируется на хосте Windows с помощью учетной записи, хранящейся в базе данных AD.
2. Служба клиента NFS на клиенте Windows использует протокол NFS для доступа к серверу NFS на хосте Linux.
3. Во время доступа к серверу NFS клиент отправляет UID/GID учетной записи, указанной в качестве параметра соединения (1002/1003).
4. Полученный UID/GID сопоставляется с именем пользователя в базе данных passwd, создавая сессию пользователя с соответствующим uid/gid.
5. Если разрешения на доступ к сетевому ресурсу в /etc/exports существуют, клиенту предоставляется доступ к ресурсу. Разрешения локального каталога /pub/nfs определяют, разрешена или запрещена конкретная операция.

Вывод

В ходе работы были изучены сетевые протоколы доступа к файлам: SMB и NFS. Так же произведена настройка конфигурирования файл сервера на двух ОС: Linux и Windows, которые в ходе работы выполняли в роли сервера и клиента.

Сходства протоколов:

* SMB и NFS используются для обмена файлами и ресурсами между компьютерами.
* Оба протокола позволяют нескольким пользователям одновременно получать доступ к файлам и обмениваться ими.
* Оба поддерживают механизмы безопасности и контроля доступа для ограничения доступа к общим ресурсам.

Различия:

* SMB – это протокол Microsoft, в то время как NFS - это протокол на базе Unix.
* SMB используется в основном в системах на базе Windows, а NFS - в системах на базе Unix.
* SMB поддерживает как TCP, так и NetBIOS, в то время как NFS использует только TCP.
* SMB поддерживает совместное использование файлов и принтеров, в то время как NFS используется в основном для совместного использования файлов.
* SMB имеет встроенную поддержку аутентификации и шифрования, в то время как NFS полагается на внешние механизмы безопасности, такие как Kerberos или SSL.
* NFS обеспечивает лучшую производительность на системах на базе Unix благодаря встроенной интеграции с операционной системой.

Плюсы и минусы SMB:

* Простота настройки и использования в системах на базе Windows, поддерживает совместное использование файлов и принтеров, встроенную аутентификацию и шифрование, может использоваться как с TCP, так и с NetBIOS.
* Более низкая производительность на системах на базе Unix, отсутствие встроенной интеграции с системами на базе Unix.

Плюсы и минусы NFS:

* + Более высокая производительность в системах на базе Unix, встроенная интеграция с системами на базе Unix, встроенная поддержка TCP, более простая настройка и использование в системах на базе Unix.
* Не так широко используется и поддерживается, как SMB, зависит от внешних механизмов безопасности для аутентификации и шифрования.

В целом, SMB и NFS являются полезными протоколами для обмена файлами и ресурсами между компьютерами. SMB в основном используется в системах на базе Windows, а NFS - в системах на базе Unix. Оба протокола имеют свои сильные и слабые стороны, и выбор между ними будет зависеть от конкретных потребностей пользователей и среды, в которой они используются.