Министерство науки и высшего образования Российской Федерации

Московский государственный университет геодезии и картографии (МИИГАиК)

Факультет геоинформатики и информационной безопасности

**Лабораторная работа** по дисциплине «Администрирование информационных и автоматизированных систем»

**Тема:**

«Настройка и установка FTP-сервера»

Выполнил: студент

data

data

Проверил:

data

**Москва –** data

**Введение:**

FTP (RFC-959) обеспечивает файловый обмен между удаленными пользователями. Протокол FTP формировался многие годы. Первые реализации в МТИ относятся к 1971. (RFC 114 и 141). RFC 172 рассматривает протокол, ориентированный на пользователя, и предназначенный для передачи файлов между ЭВМ. Позднее в документах RFC 265 и RFC 281 протокол был усовершенствован. Заметной переделке протокол подвергся в 1973, и окончательный вид он обрел в 1985 году. Таким образом, данный протокол является одним из старейших. Для реализации обмена между двумя персональными ЭВМ в пределах сети (программные пакеты PCTCP, и т.д.) можно резидентно загрузить FTPSRV или другую эквивалентную программу.

Работа FTP на пользовательском уровне содержит несколько этапов:

1. Идентификация (ввод имени-идентификатора и пароля).

2. Выбор каталога.

3. Определение режима обмена (поблочный, поточный, ASCII или двоичный).

4. Выполнение команд обмена (get, mget, dir, mdel, mput или put).

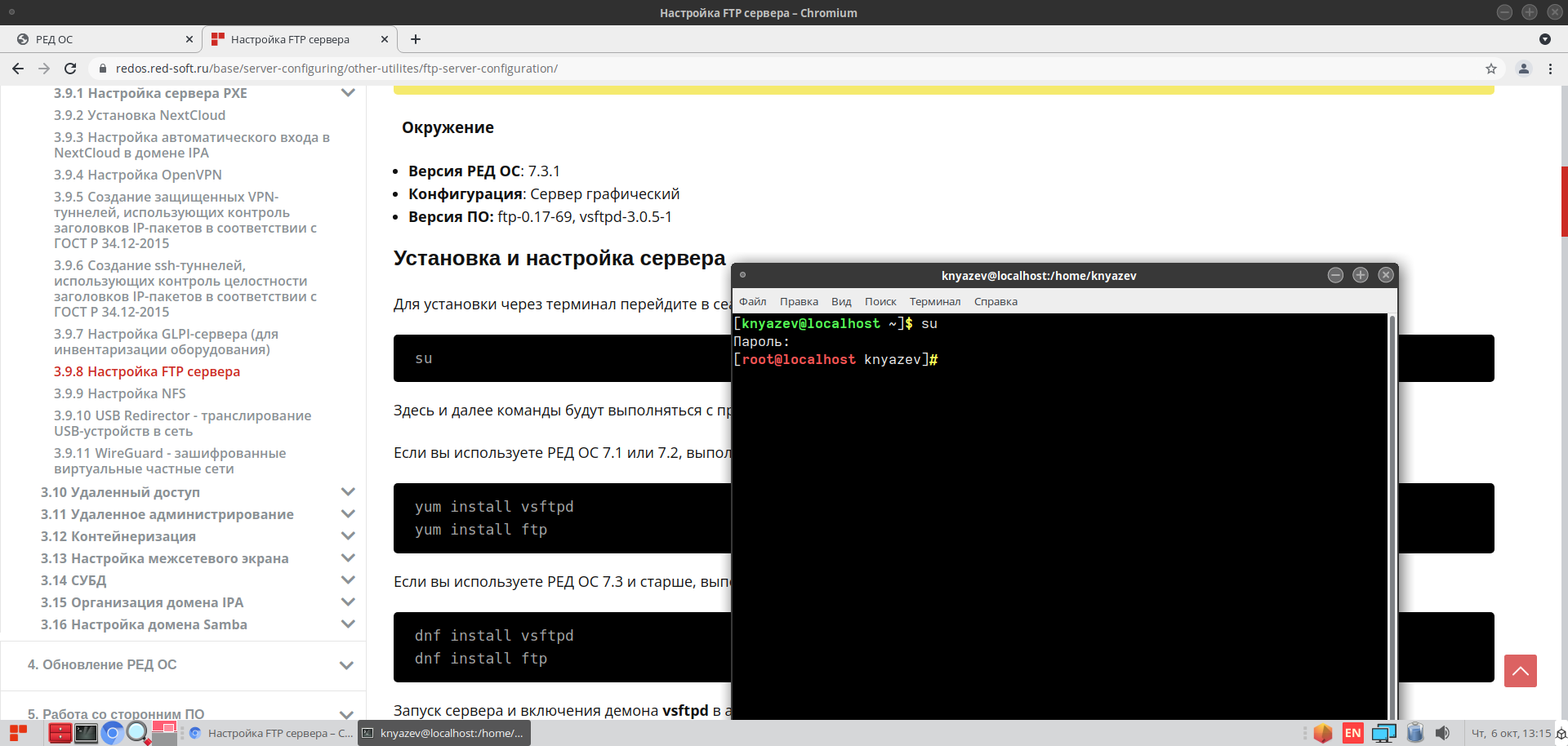
5. Завершение процедуры (quit или close).

FTP довольно необычная процедура, так как поддерживает две логические связи между ЭВМ. Одна связь служит для удаленного доступа и использует протокол Telnet. Другая связь предназначена для обмена данными. Сервер производит операцию passive open для порта 21 и ждет соединения с клиентом. Клиент осуществляет операцию active open для порта 21. Канал остается активным до завершения процедуры FTP. TOS (тип IP-сервиса) соответствует минимуму задержки, так как этот канал используется для ручного ввода команд. Канал для передачи данных (TCP) формируется каждый раз для пересылки файлов. Канал открывается перед началом пересылки и закрывается по коду end\_of\_file (конец файла). IP-тип сервиса (TOS) в этом случае ориентирован на максимальную пропускную способность.

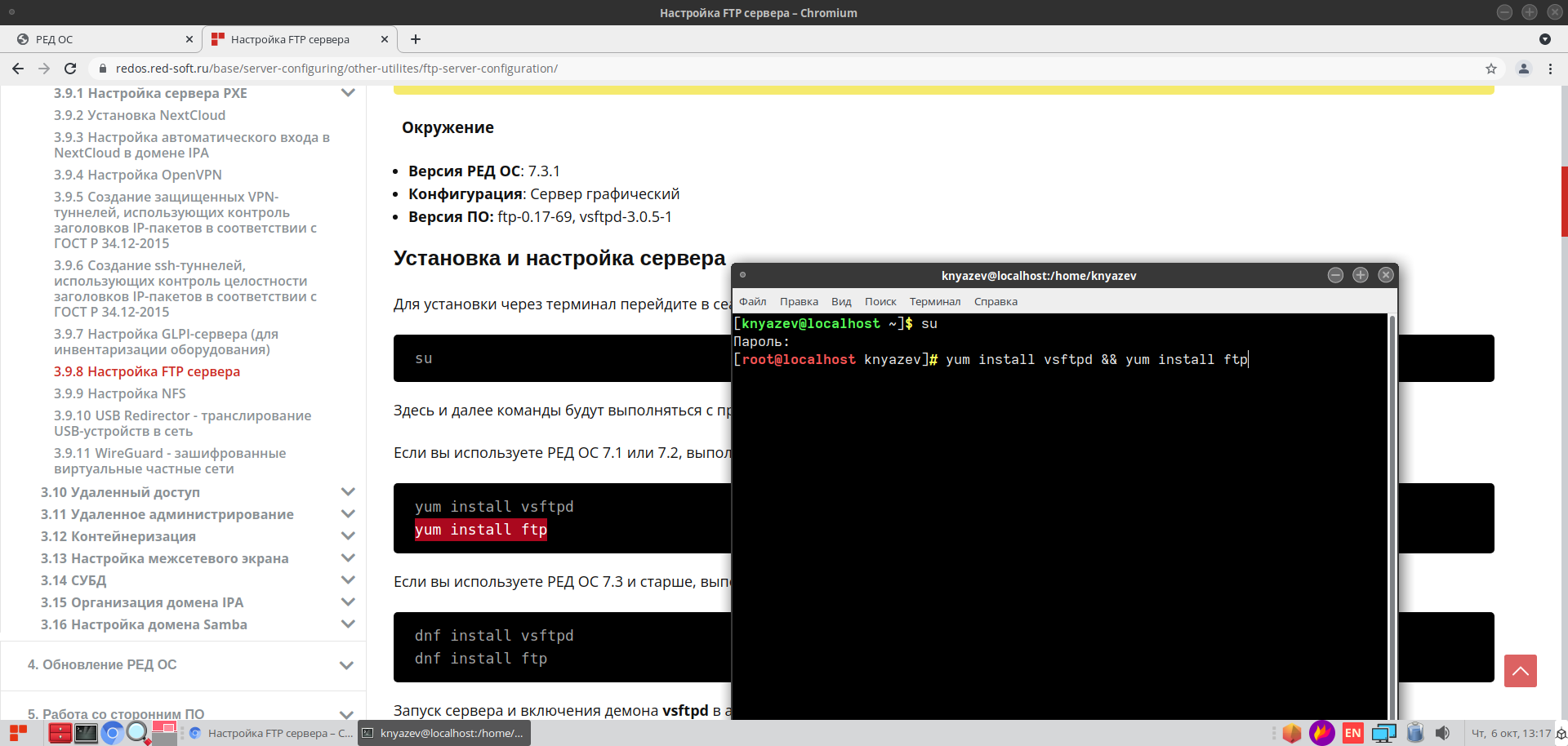
Конечный пользователь взаимодействует с протокольным интерпретатором, в задачи которого входит управление обменом информацией между пользователем и файловой системой, как местной, так и удаленной. Сначала по запросу клиента формируется канал управления, который в дальнейшем используется для передачи команд от клиента и откликов от сервера. Информационный канал формируется сервером по команде клиента, он не должен существовать постоянно на протяжении всей FTP-сессии и может формироваться, и ликвидироваться по мере необходимости. Канал управления может быть закрыт только после завершения информационного обмена. Для канала управления используется протокол Telnet. После того как управляющий канал сформирован, клиент может посылать по нему команды. Сервер воспринимает, интерпретирует эти команды и передает отклики.

**Основная часть:**

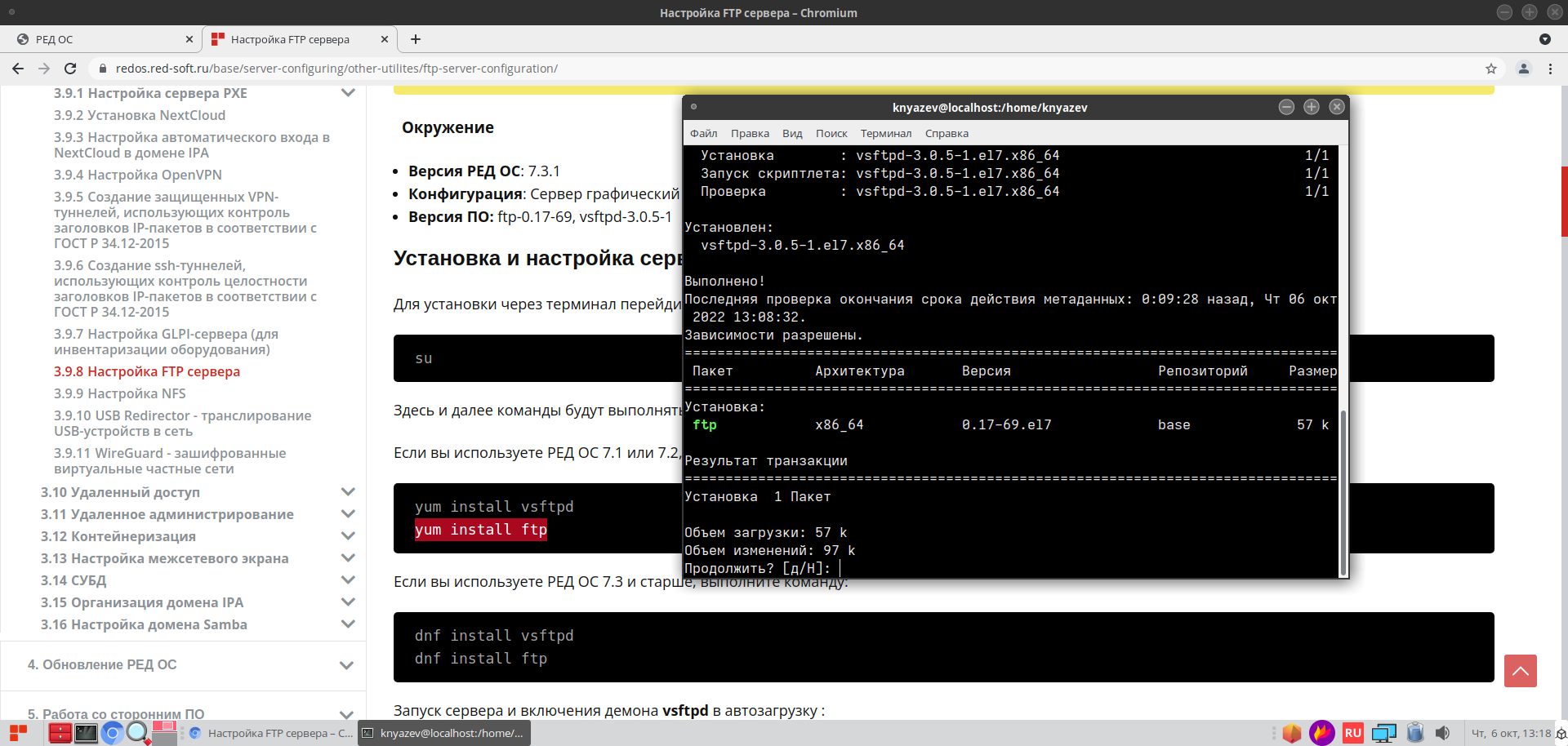
Для того, чтобы иметь доступ и права ко всем функциям операционной системы, пропишем команду **su** в терминале, авторизируемся и войдём, как суперпользователь: change photo



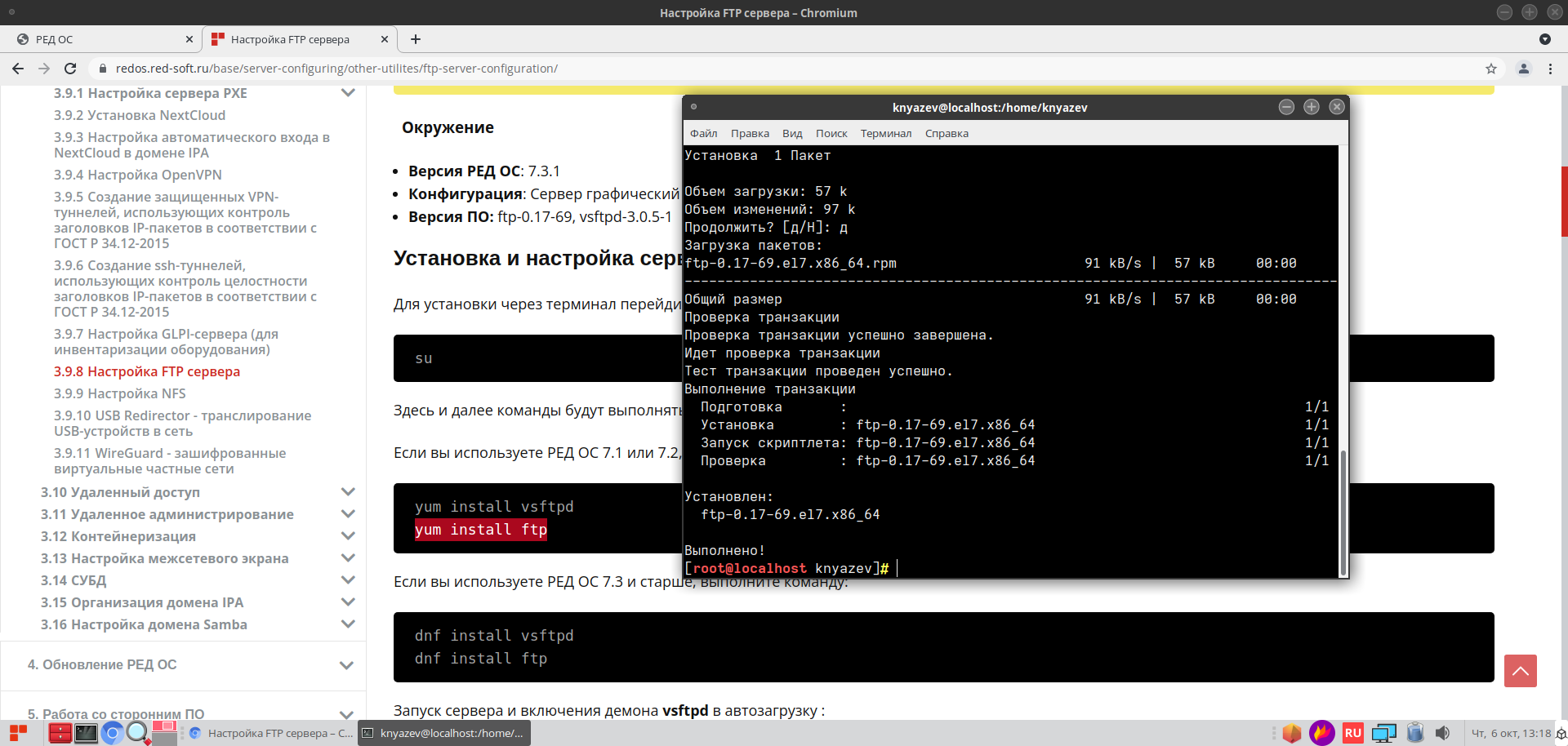
Используя пакетный менеджер **yum**, установим утилиты по работе с FTP-сервером командой **yum install vsftpd && yum install ftp**, где **&&** - этологическое «И»: change photo



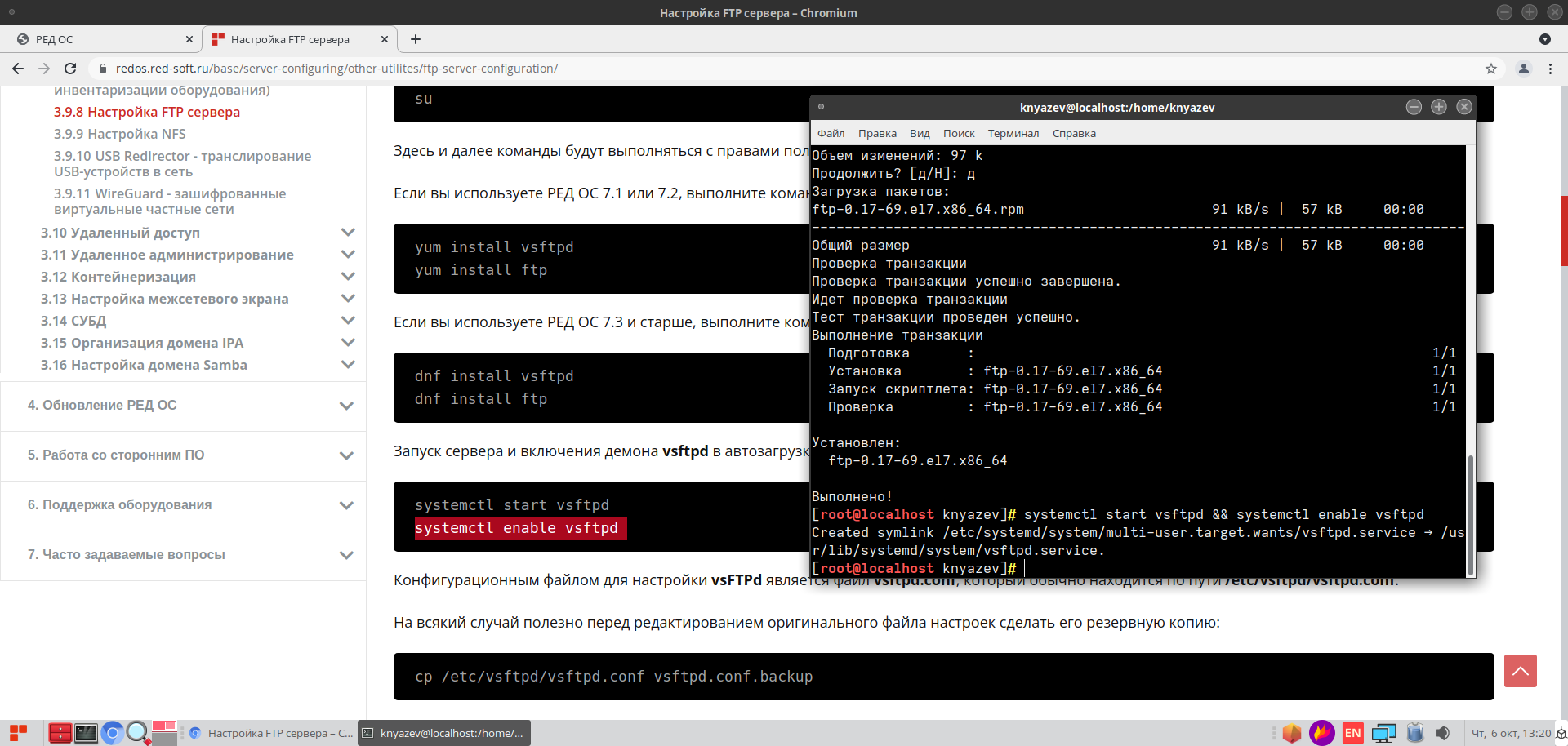
Соглашаемся с условиями загрузки: change photo



И ждём выполнения: change photo

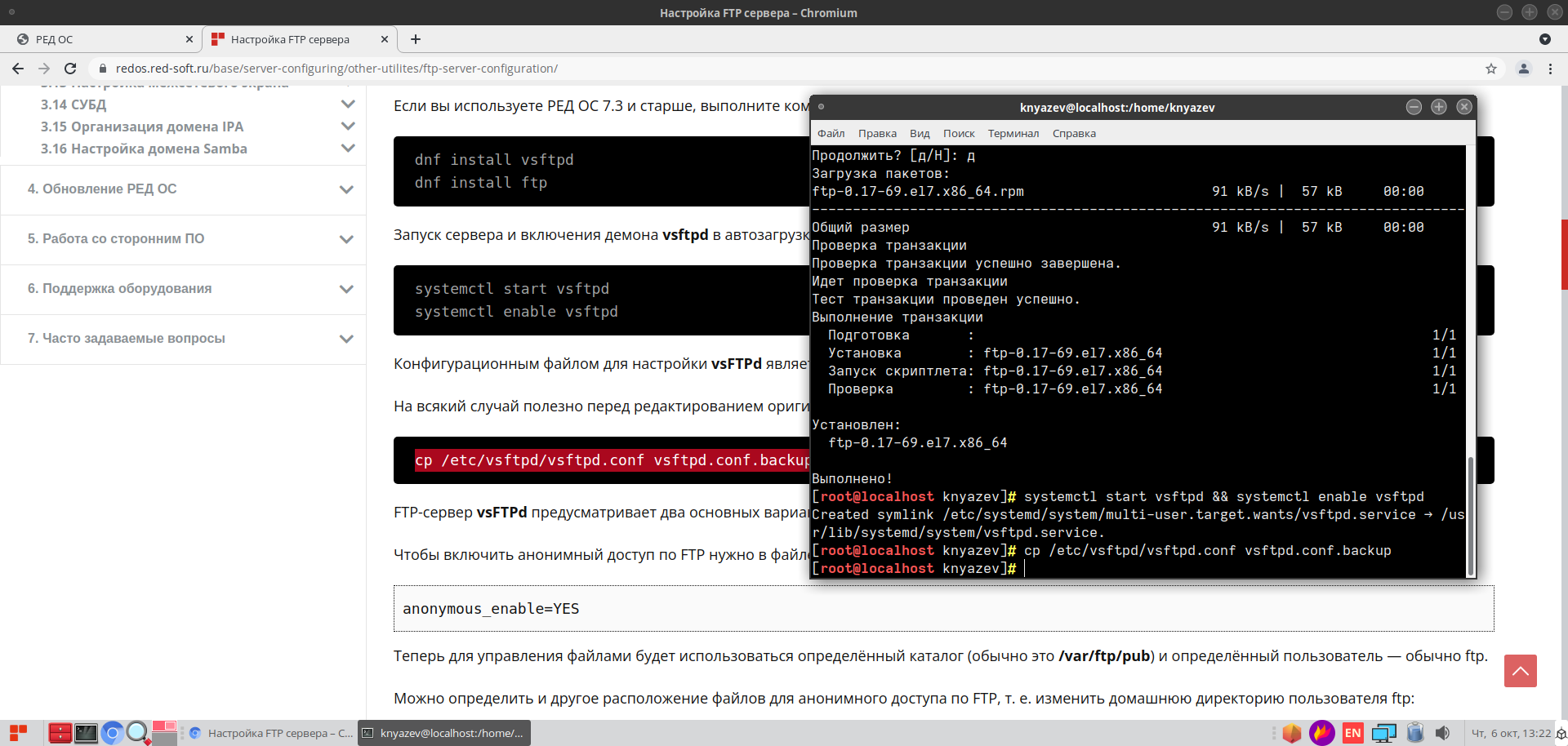


С помощью системного диспетчера **systemctl** (система инициализации), который стал новым стандартом для дистрибутивов Linux, запустим сервер и включим **vsftpd** в автозагрузку. Команда **systemctl start vsftpd && systemctl enable vsftpd**:

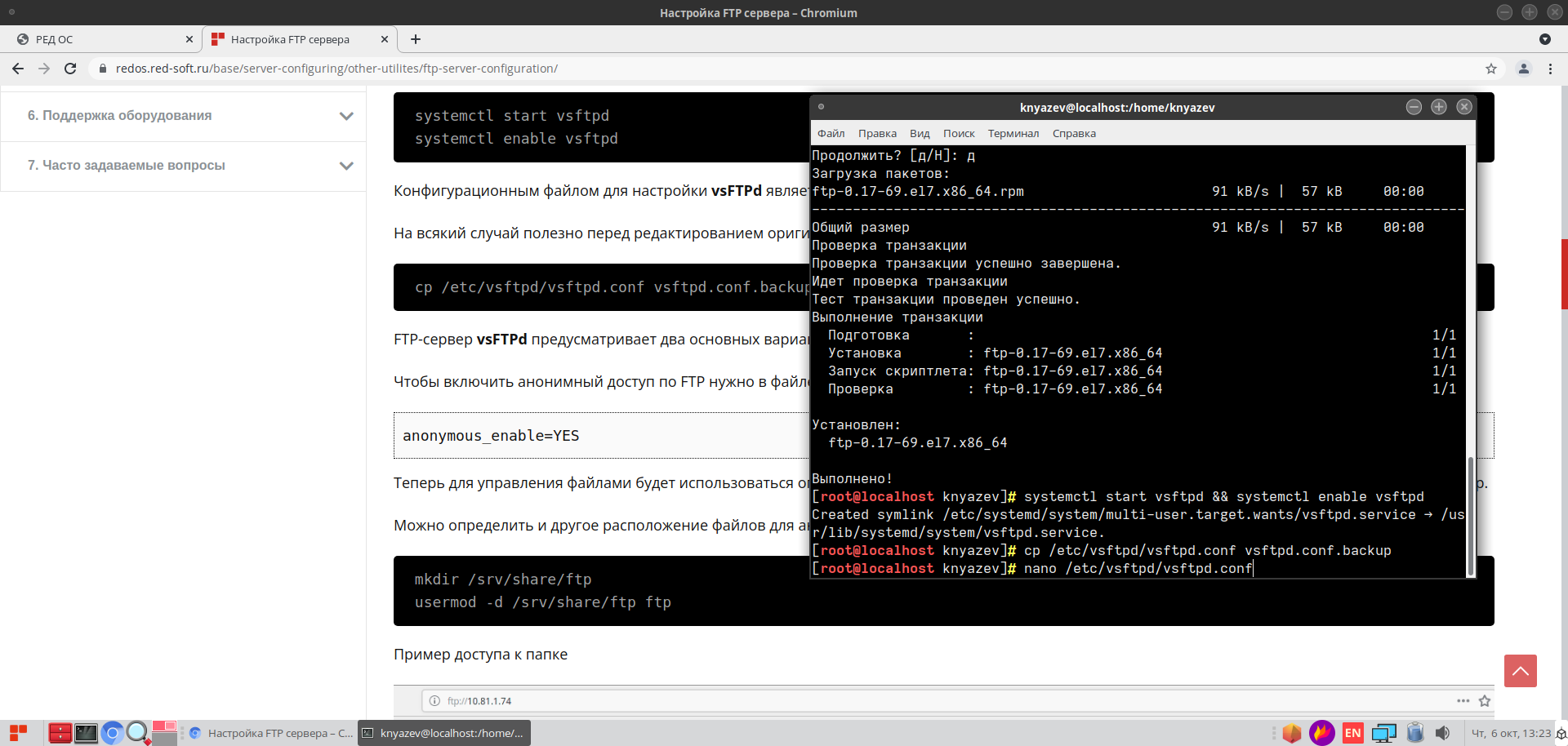


Конфигурационным файлом для настройки **vsFTPd** является файл **vsftpd.conf**, который находится по пути **/etc/vsftpd/vsftpd.conf**. change photo

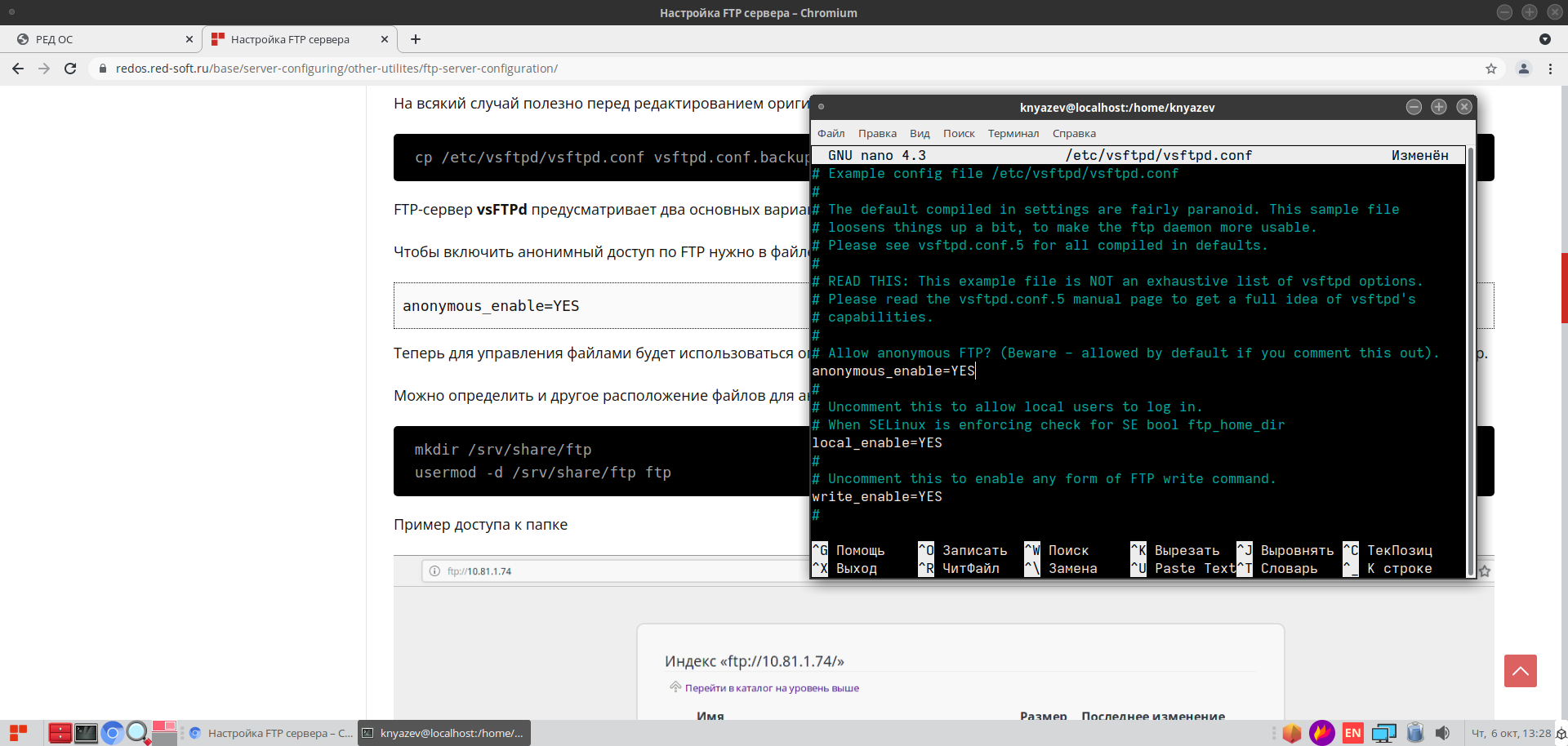
Чтобы иметь возможность откатиться к начальной версии конфигурационного файла, с помощью команды **cp** создадим резервную копию. Полная команда **cp /etc/vsftpd/vsftpd.conf** *(путь)* **vsftpd.conf.backup** *(бэкап)*: change photo



Через встроенный текстовый редактор **nano**, открываем файл конфигурации **vsftpd.conf** для редактирования: change photo

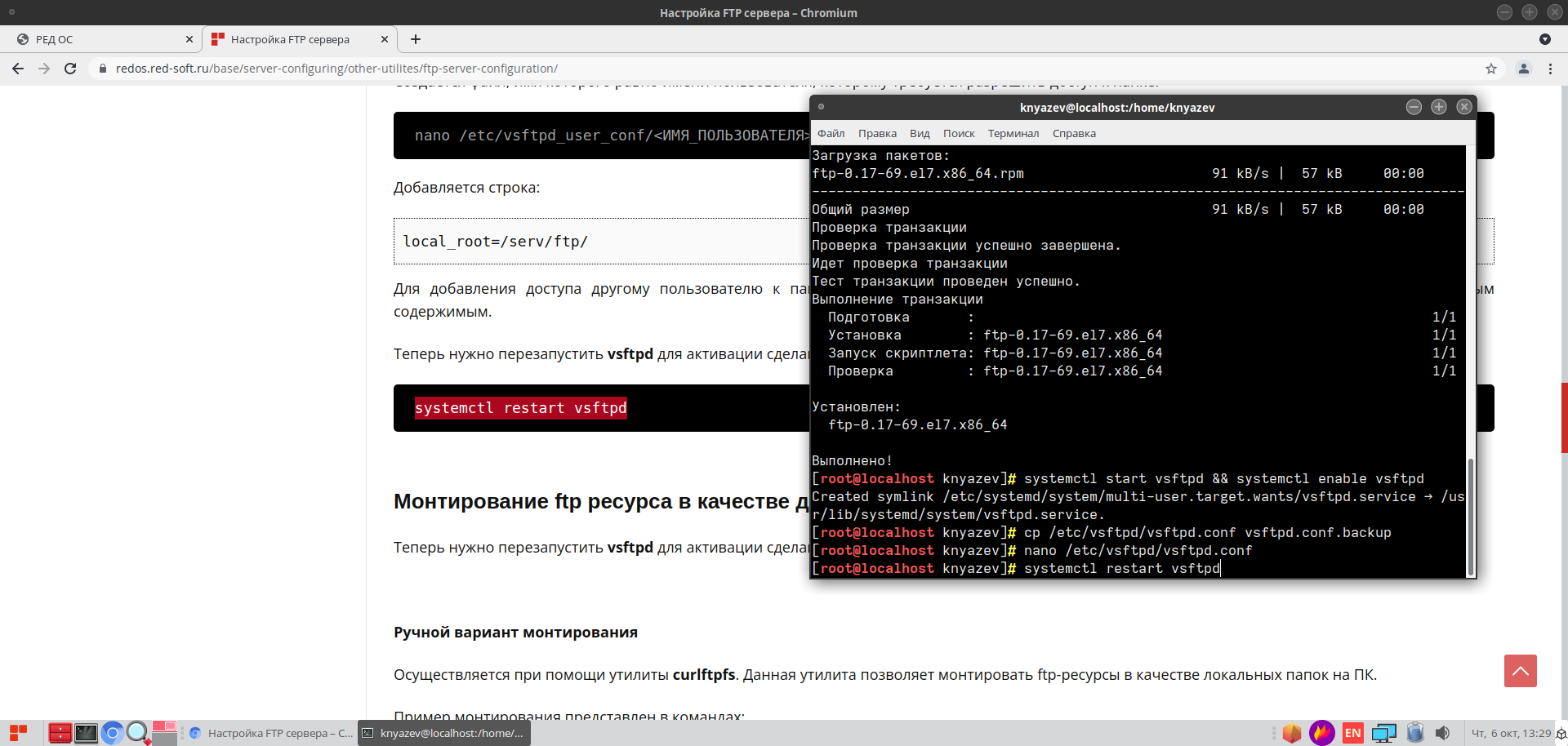


После ввода команды, появляется редактор. Меняем параметр **anonymous\_enable** на YES, чтобы включить анонимный доступ по FTP:

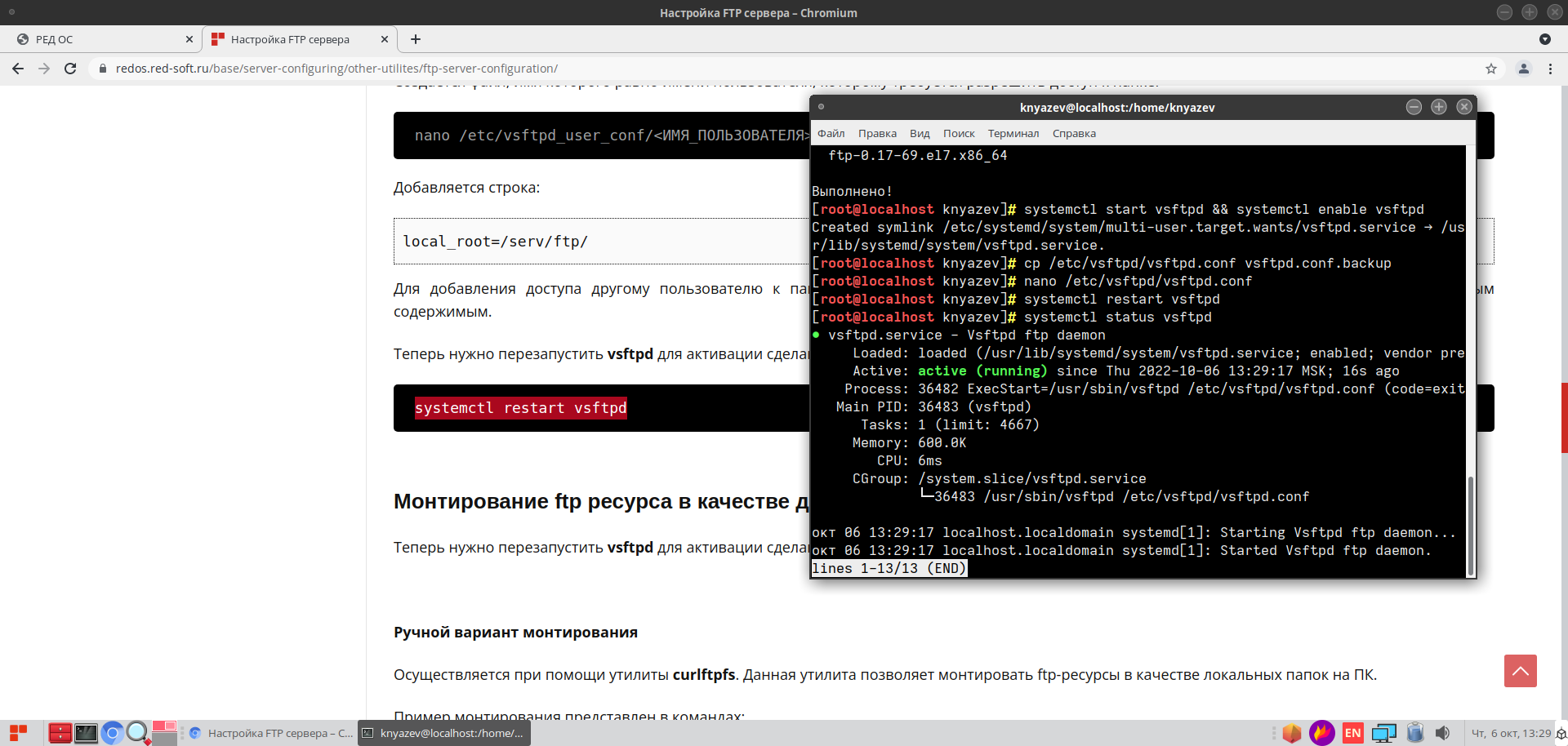


Для того чтобы сохранить изменения после завершения работы с файлом необходимо нажать сочетание клавиш Ctrl+O. change photo

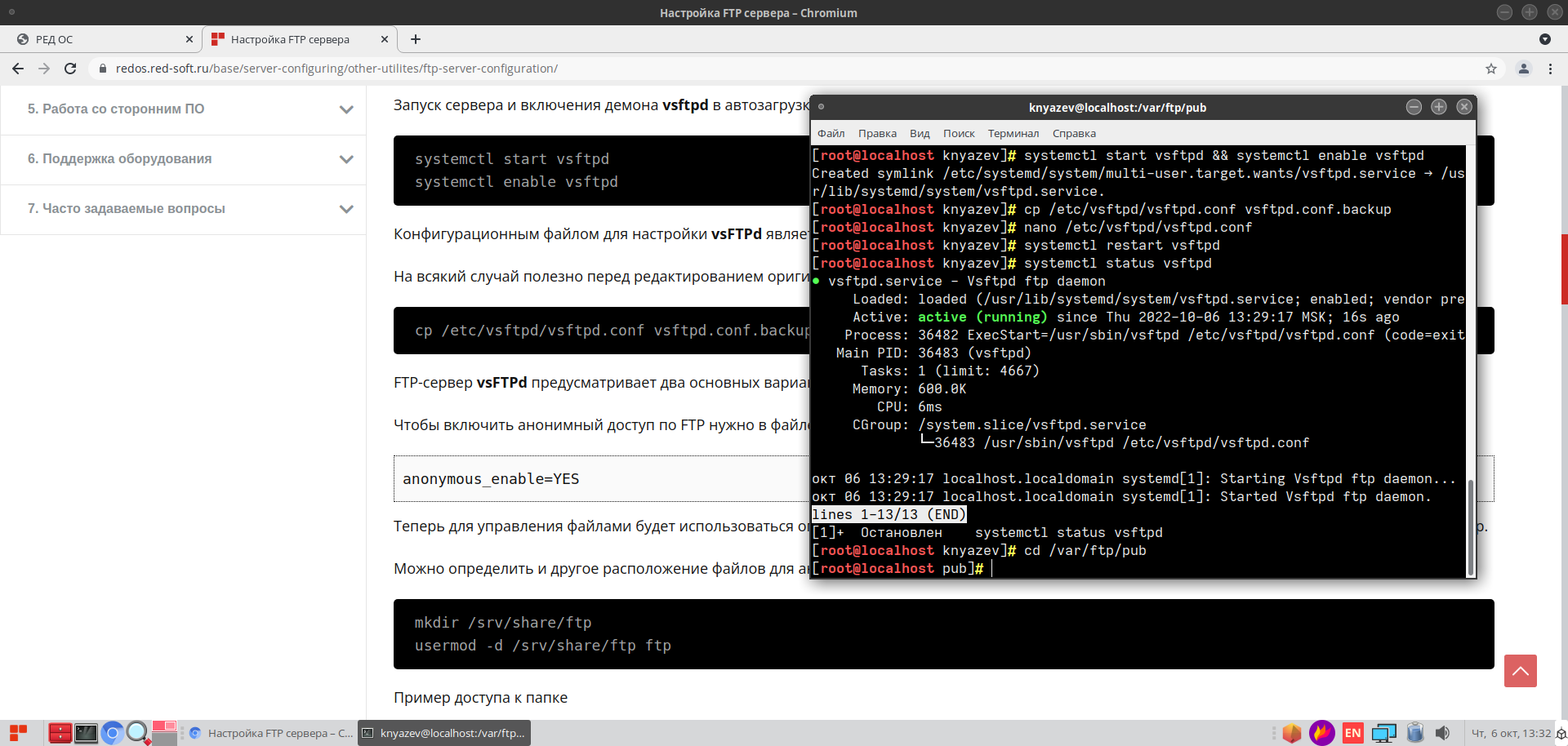
Теперь перезапустим vsftpd командой **systemctl restart vsftpd** для активации сделанных изменений: change photo



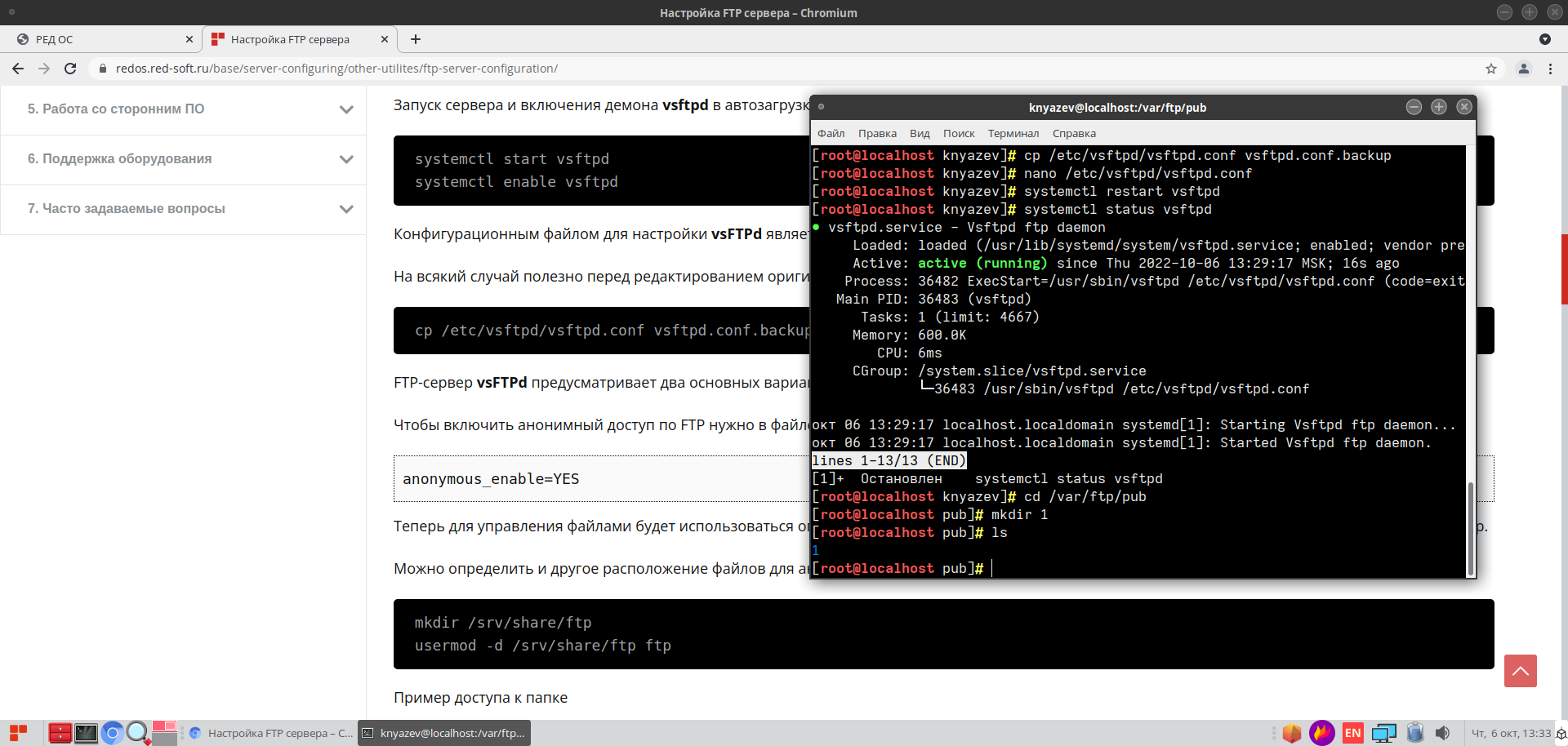
Командой **systemctl status vsftpd** проверим статус FTP-сервера, он должен быть активным: change photo



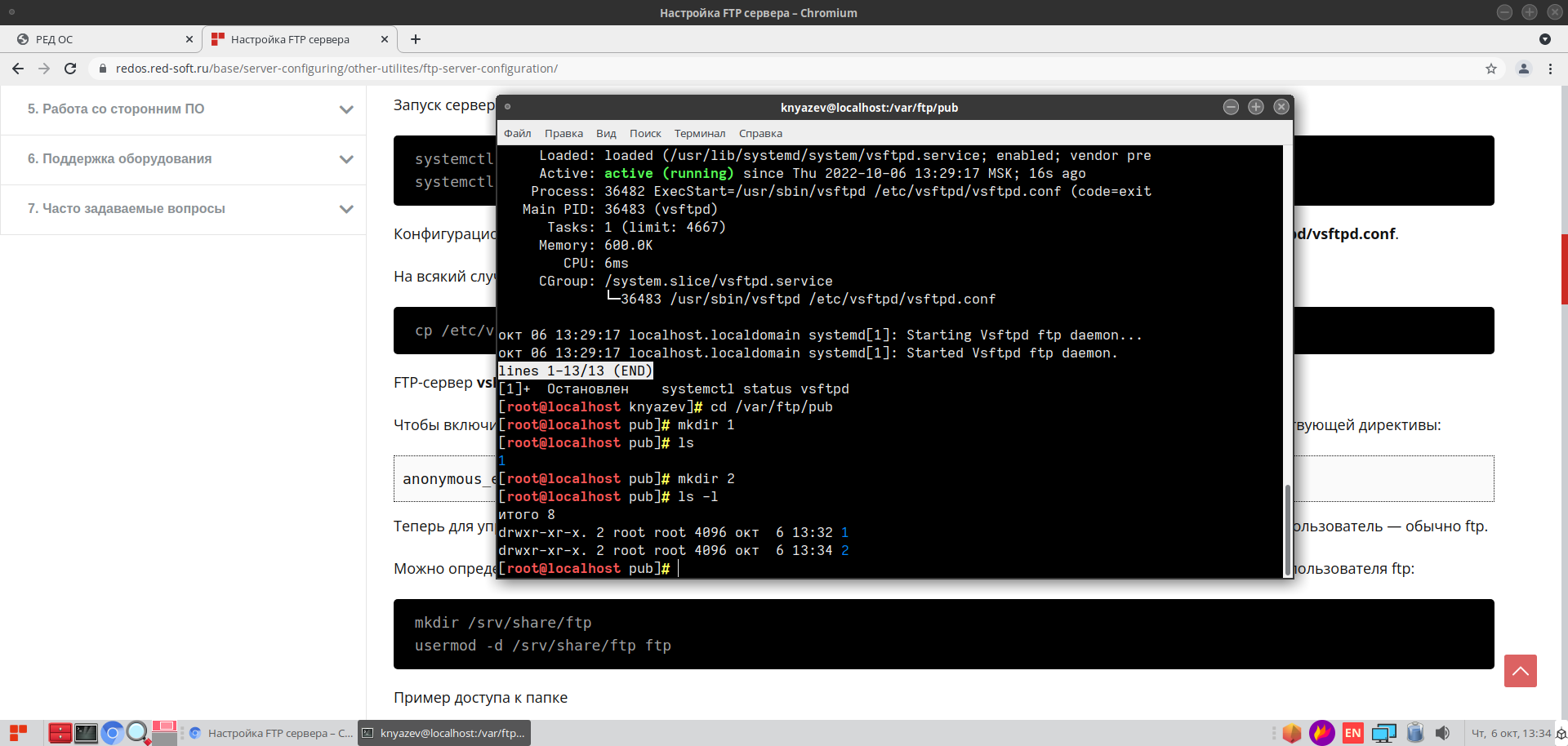
Через **cd /var/ftp/pub** зайдём в стандартную директорию управления vsftpd: change photo



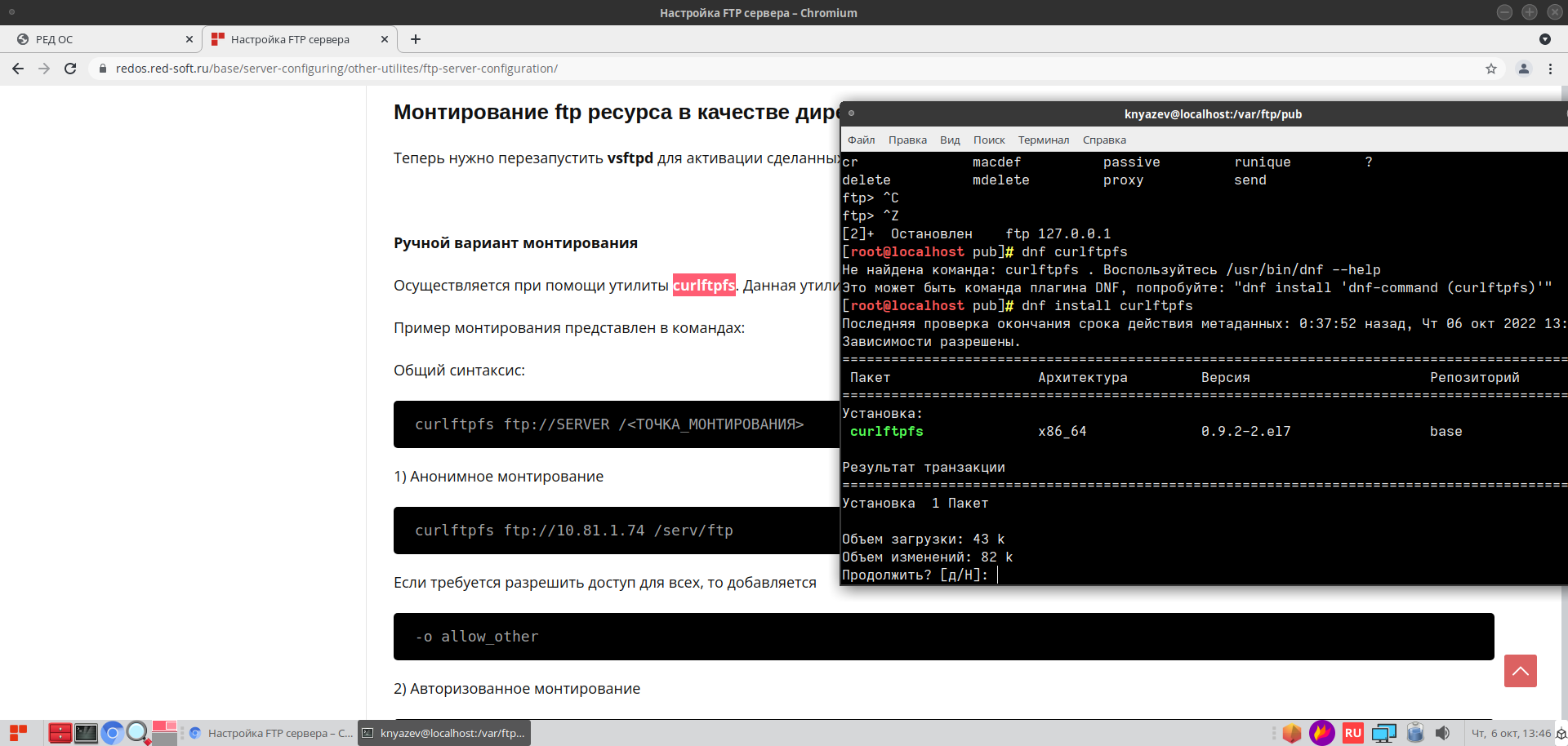
Специальной командой **mkdir 1** создадим папку 1, и после командой **ls** выведем содержимое директории: change photo



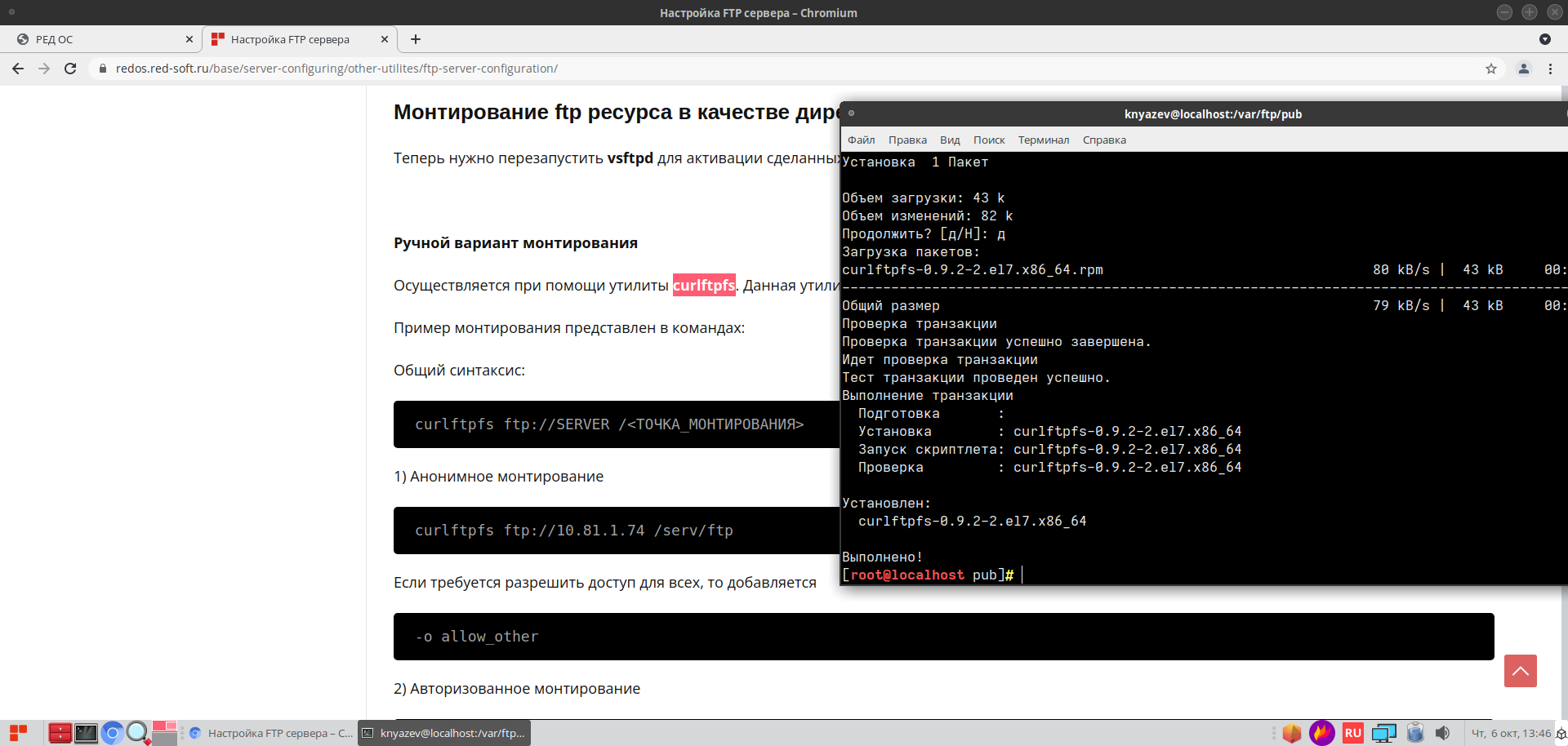
Той же командой **mkdir 2** создадим папку 2, и командой **ls –l** выведем содержимое директории в виде списка: change photo



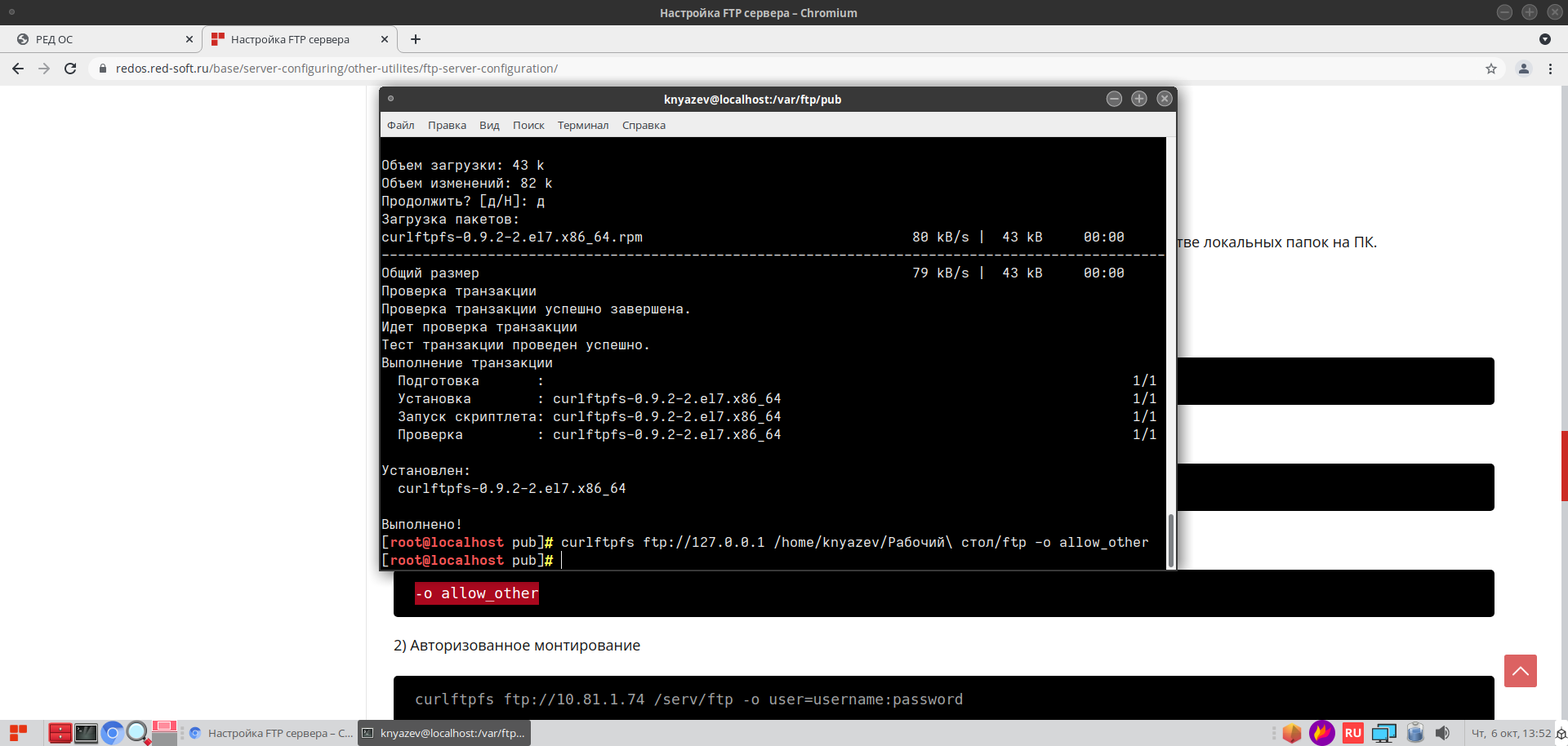
С помощью пакетного менеджера dnf (следующее поколение приложения yum, менеджер пакетов для дистрибутивов ОС Linux на основе RPM-пакетов) установим CurlFtpFS, утилиту для монтирования удалённых FTP-хранилищ в локальную файловую систему. Команда **dnf install CurlFtpFS**: change photo

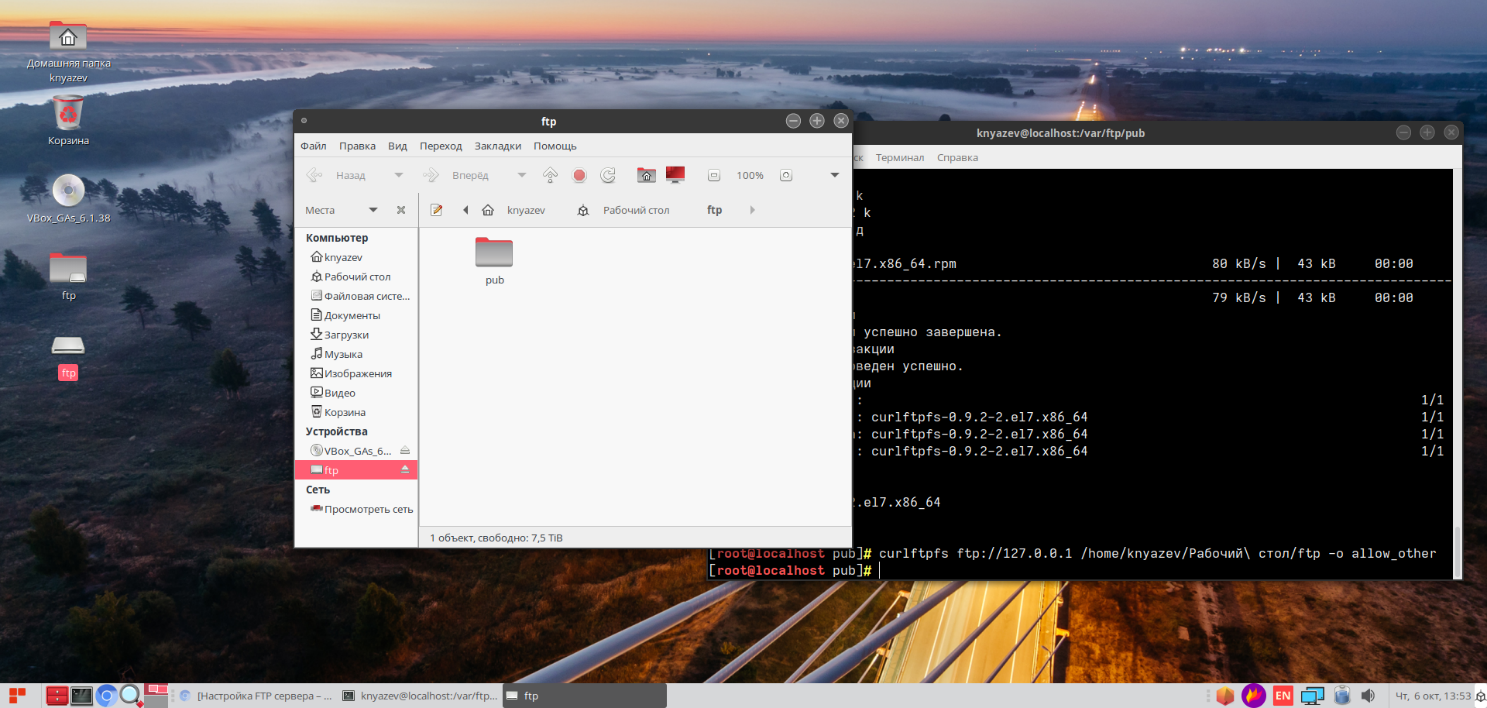


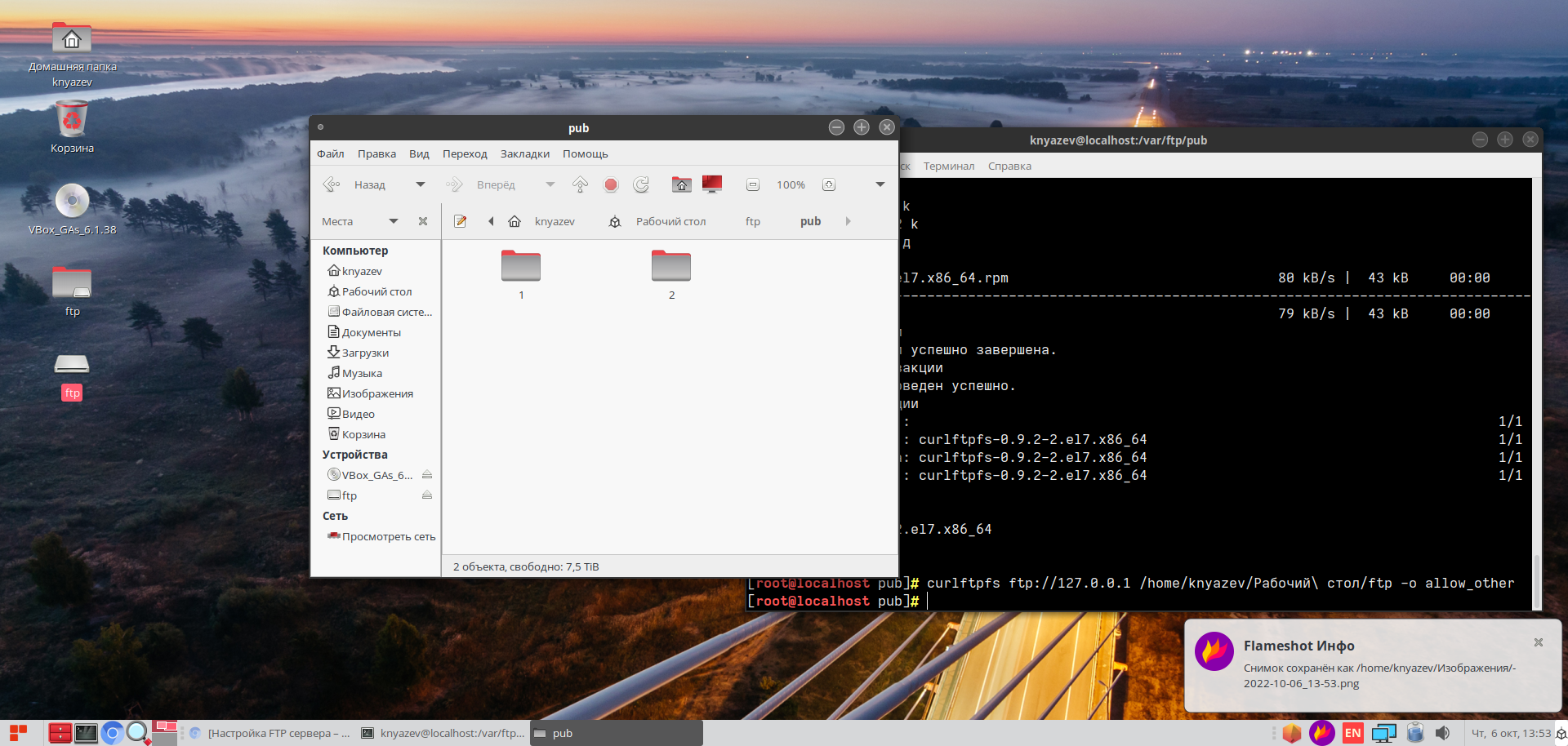
Соглашаемся с условиями и ждём завершения установки: change photo



Создадим папку на рабочем столе под названием ftp. Далее в терминале пропишем команду **curlftpfs** [**ftp://127.0.0.1**](ftp://127.0.0.1) *(адрес)* **/home/Knyazev/Рабочий\ стол/ftp** *(путь к созданной папке)* **-o allow\_other** *(дополнение: разрешить доступ для всех)* change photo



После всего этого на рабочем столе появится специальная папка с таким же названием ftp, но уже с доступом к созданным нами ранее директориям: change photo 



change photo

**Состав конфигурационного файла VSFTPD.CONF:**

Файл vsftpd.conf состоит либо из комментариев, либо из директива. Строчки комментариев начинаются с символа решётки "#" и игнорируются. У строки-директивы имеется свой формат: "параметр=значение". В конфигурационном файле **VSFTPD.CONF** всего имеется 3 вида параметров:

1. **Логические параметры**, которые могут содержать логические значения (YES/NO).

**Примеры:**

**anonymous\_enable=YES**, для включения/выключения анонимного доступ по FTP;

**local\_enable=YES**, для разрешения/запрета анонимного входа под локальными логинами.

1. **Числовые параметры**, которые могут содержать значения в числах (например, номер порта).

**Примеры:**

**local\_umask=022**, которое отвечает за то, какие права будут присвоены файлам, которые были закачаны через FTP. Маска рассчитывается по формуле 777-A=X, где: 777-права полного доступа А-права которые необходимо получить Х-маска, которую необходимо выставить;

**data\_connection\_timeout=120**, которое отвечает за максимальный временной промежуток в секундах, разрешенного замирания процесса передачи данных. Если перерыв превышен соединение с удаленным клиентом отбрасывается.

1. **Строковые параметры**, которые могут содержать строку, то есть путь к каталогу в файловой системе на диске.

**Примеры:**

**secure\_chroot\_dir=/var/run/vsftpd**. Эта опция указывает на имя пустого каталога. Также, каталог не должен быть записываемый для ftp пользователя. Этот каталог используется как безопасный chroot(), когда vsftpd не нужен доступ к файловой системе.

**email\_password\_file=/etc/vsftpd.email\_passwords**. Эта опция может быть использована для предоставления альтернативного пути файла, используемого secure\_email\_list\_enable опцией.

**Пакетный менеджер DNF:**

DNF - это менеджер пакетов проекта Fedora, который может запрашивать информацию о пакетах, извлекать пакеты из репозиториев, устанавливать и удалять пакеты с использованием автоматического разрешения зависимостей и обновлять всю систему до последних доступных пакетов. DNF выполняет автоматическое разрешение зависимостей для пакетов, которые вы обновляете, устанавливаете или удаляете, и, таким образом, может автоматически определять, извлекать и устанавливать все доступные зависимые пакеты. DNF можно настроить с помощью новых дополнительных репозиториев илиисходные коды пакета, а также предоставляет множество плагинов, которые улучшают и расширяют его возможности. DNF способен выполнять многие из тех же задач, что и RPM; кроме того, многие параметры командной строки аналогичны. DNF позволяет легко и просто управлять пакетами на одном компьютере или на их группах.

**Команды su и sudo:**

Как su, так и sudo используются для запуска команд с привилегиями root. Пользователь root в целом эквивалентен администратору в Windows - он имеет наибольшие права и может делать с системой практически все. Обычные пользователи в Linux имеют намного меньшие возможности - например, они не могут устанавливать программы или записывать в системные директории.

При выполнении команды su без дополнительных опций она переключает вас в сеанс суперпользователя (или root). Для этого вам необходимо знать пароль root. Это не все, что может делать команда su, на самом деле она может использоваться для переключения в любой пользовательский аккаунт, командой su [имя пользователя], после этого оболочка переключится в его аккаунт.

Sudo запускает с правами root только одну команду. При выполнении команды sudo система запросит у ваш текущий пользовательский пароль, а затем запустит команду на выполнение от имени root.

Это ключевое различие между su и sudo. Su переключает вас в аккаунт root и требует пароля root. Sudo запускает с привилегиями root одну команду - она не переключает вас в аккаунт суперпользователя и не требует отдельного пароля root.

**Файл конфигурации sudoers:**

Sudoers — это файл, который контролирует, какие учетные записи имеют доступ к команде sudo. Открыть конфиг можно специальной командой visudo, которая запускает текстовый редактор обычным образом, но проверяет синтаксис файла при его сохранении. Это не даст ошибкам конфигурации возможности блокировать операции sudo, что может быть единственным способом получить привилегии root.

Параметр Defaults env\_reset сбрасывает среду терминала для удаления переменных пользователя. Эта мера безопасности используется для сброса потенциально опасных переменных среды в сеансе sudo.

Второй параметр, начинающийся с “Defaults secure\_path=…”, задает переменную PATH (места в файловой системе, где операционная система будет искать приложения), которая будет использоваться для операций sudo. Это предотвращает использование пользовательских путей, которые могут быть вредоносными.

Далее идёт параметр, который определяет для пользователя root привилегии sudo: root ALL=(ALL:ALL) ALL. Первое поле показывает имя пользователя, которое правило будет применять к (root). Первое “ALL” означает, что данное правило применяется ко всем хостам. Второе “ALL” означает, что пользователь root может запускать команды от лица всех пользователей. Третье “ALL” означает, что пользователь root может запускать команды от лица всех групп. Последнее “ALL” означает, что данные правила применяются всем командам. Это означает, что наш пользователь root сможет выполнять любые команды с помощью sudo после ввода пароля.

Следующие параметры похожи на строки привилегий пользователя, но задают правила sudo для групп (%admin ALL=(ALL) ALL и % sudo ALL=(ALL:ALL) ALL). Имена, начинающиеся с %, означают названия групп. Здесь мы видим, что группа admin может выполнять любые команды от имени любого пользователя на любом хосте. Группа sudo имеет те же привилегии, но может выполнять команды от лица любой группы.

Последний параметр выглядит как комментарий: #includedir /etc/sudoers.d. Она действительно начинается с символа #, который обычно обозначает комментарии. Однако данная строка означает, что файлы в каталоге /etc/sudoers.d также рассматриваются как источники и применяются. Файлы в этом каталоге следуют тем же правилам, что и сам файл /etc/sudoers. Любой файл, который не заканчивается на ~ и не содержит символа ., также считывается и добавляется в конфигурацию sudo. В основном это нужно, чтобы приложения могли изменять привилегии sudo после установки. Размещение всех правил в одном файле в каталоге /etc/sudoers.d позволяет видеть, какие привилегии присвоены определенным учетным записям, а также легко сменять учетные данные без прямого изменения файла /etc/sudoers.

**Systemd, systemctl и Sysvinit:**

systemd — это система инициализации и системный диспетчер, который стал новым стандартом для дистрибутивов Linux. Основополагающая цель системы инициализации заключается в инициализации компонентов, которые должны запускаться после загрузки ядра Linux (традиционно называются компоненты пользовательского пространства). Система инициализации также используется для управления службами и демонами для сервера и в любой момент времени работы системы. В systemd целью большинства действий являются «модули», являющиеся ресурсами, которыми systemd знает, как управлять. Модули распределяются по категориям по типу ресурса, который они представляют, и определяются файлами, известными как файлы модулей. Тип каждого модуля можно вывести из суффикса в конце файла. Для задач по управлению службами целевым модулем будут модули службы, которые имеют файлы модулей с суффиксом .service.

В Systemd есть специальный инструмент для управления службами в Linux - systemctl. Эта утилита позволяет делать очень много вещей, начиная от перезапуска службы linux и проверки ее состояния, до анализа эффективности загрузки службы. Синтаксис у утилиты такой: systemctl [опции] [команда] [служба] [служба] ... . Опции настраивают поведение программы, подробность вывода, команда - указывает что нужно сделать со службой, а служба, это та самая служба, которой мы собираемся управлять. В некоторых случаях утилита может использоваться без указания команды и службы.

SystemV или SysV - это довольно старая, но до сих пор ещё популярная система инициализации Linux и Unix подобных операционных систем. Она была основой для создания многих других систем инициализации, а также первой коммерческой системой инициализации разработанной для Unix в AT&T. Она была разработана еще в 1983 году. Почти все дистрибутивы Linux изначально использовали SysV.

Основные возможности SysV:

1. Написание файлов запуска служб на bash;
2. Последовательный запуск служб;
3. Сортировка порядка запуска с помощью номеров в именах файлов;
4. Команды для запуска, остановки и проверки состояния служб.

Никакой параллельной загрузки, системы зависимостей, запуска по требованию и автоматического запуска здесь не было и в помине.

С момента ее разработки прошло много лет и из-за некоторых недостатков были разработаны другие системы для ее замены, они хоть и имели новые функции и были более эффективны, но они были по-прежнему совместимы с SysV.

**Монтирование файловой системы и удаленное подключение к серверу/ресурсу:**

Монтирование в Linux позволяет получить доступ к содержимому диска и организовать структуру файловой системы. С помощью монтирования также можно открыть для работы образ диска (например, созданного с помощью программы dd), а также открыть для доступа и редактирования самые разные файловые системы и образы дисков (например, образы дисков виртуальных машин); даже удалённые сетевые директории могут быть смонтированы, в результате чего они станут доступны как будто бы файлы на любом другом локальном хранилище. Кроме того, что с помощью монтирования можно работать с образами дисков, правильные настройки монтирования необходимы для компьютера к которому подключается диск для криминалистического анализа — например, этот диск не должен автоматически монтироваться с правами записи (чтобы не быть испорченным). В Linux есть такое понятие как «монтирование» диска. Чтобы получить доступ к файлам на этом диске, его нужно сначала смонтировать. Может возникнуть вопрос, зачем такие сложности? Монтирование это мощнейшая вещь, которая позволяет поразительно гибко настроить файловую систему! Суть монтирования в том, что в файловой системе создаётся новая директория (обычная папка), допустим, это папка /mnt/disk\_d. А затем командой mount указывается, что теперь, например, диск /dev/sda смонтирован в директорию /mnt/disk\_d. После этого можно получить доступ к файлам диска /dev/sda открыв папку /mnt/disk\_d в любом менеджере файлов.

Удаленное подключение к серверу/ресурсу — это технология, позволяющая на расстоянии управлять ресурсом или сервером через интернет. Принцип работы заключается в предоставлении административных прав одному устройству (клиенту) для управления другим (сервером).

Самое главное отличие между этими способами подключения – так это то, что при монтировании нам приходится ставить точку монтирования и взаимодействовать с файлами системы. Также при монтировании не нужно ограничивать пользователя в передвижении по директориям, что мы смонтировали, то и есть, а вот при удалённом подключении, нам нужно ограничивать. При монтировании мы можем использовать поиск файлов, в при удалённом, нам нужно будет использовать какой-нибудь сторонний ресурс для такого функционала.

**Список источников:**

**1)** Настройка FTP сервера // URL: https://redos.red-soft.ru/base/server-configuring/other-utilites/ftp-server-configuration/ (Дата обращения: data)

**2)** Опции vsftpd.conf // URL: <https://www.aitishnik.ru/linux/ftp-server-debian/optsii-vsftpd-conf.html> (Дата обращения: data)

**3)** DNF :: Fedora Docs // URL: <https://docs.fedoraproject.org/en-US/fedora/latest/system-administrators-guide/package-management/DNF/> (Дата обращения: data)

**4)** Редактирование файла Sudoers // URL: <https://www.digitalocean.com/community/tutorials/how-to-edit-the-sudoers-file-ru> (Дата обращения: data)

**5)** Системы инициализации Linux // URL: <https://losst.pro/sistemy-initsializatsii-linux> (Дата обращения: data)