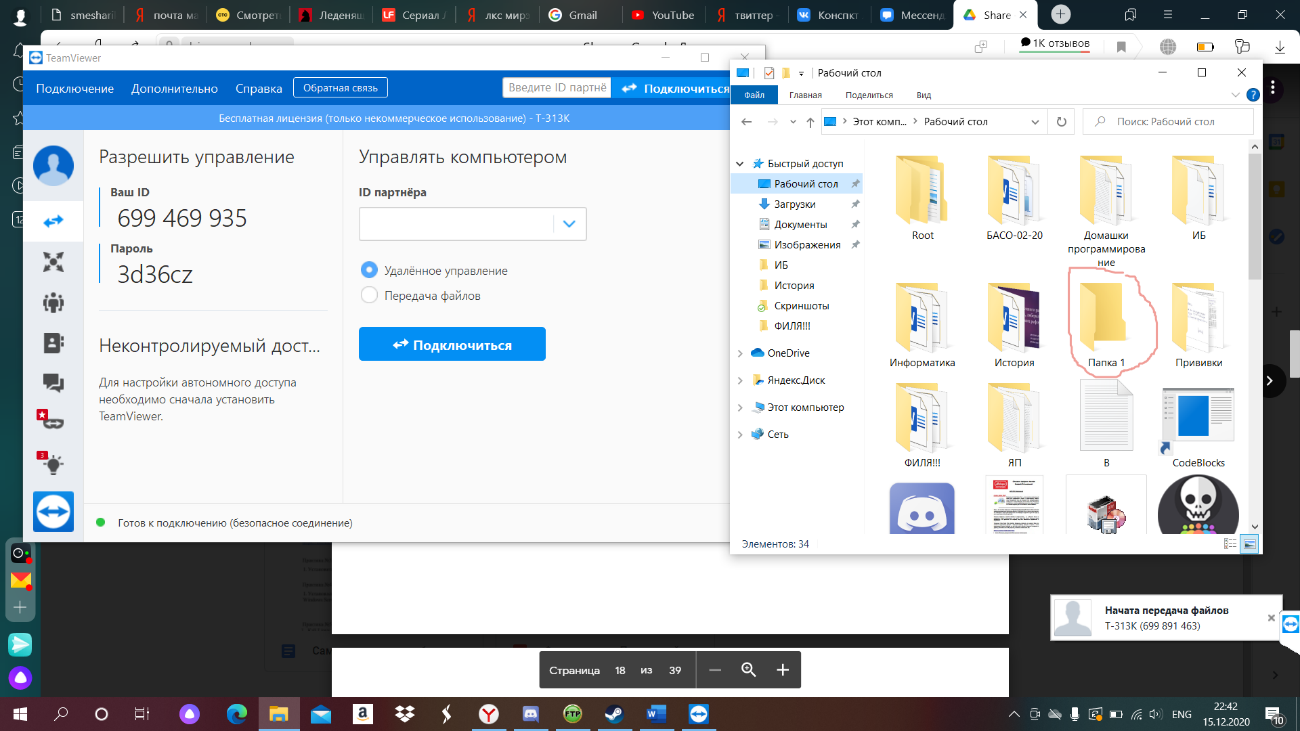
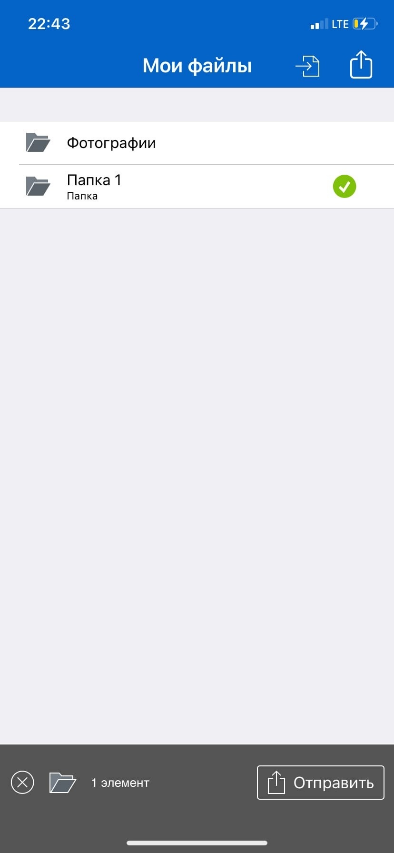
**Электронный отчет**

1. **TEAMVIEWER**Установим программу с официального сайта и добавим в автозагрузку.

После этого установим программу на телефон, создадим личный кабинет, выполним подключение и передадим какой-нибудь файл.



**2) RDP**  
  
Для настройки удаленного доступа требуется выполнить следующие

действия.

1. На устройстве c ОС Windows требуется открыть меню «Пуск» и нажать

значок «Параметры».

2. Выбрать группу «Система» возле элемента «Удаленный рабочий стол».

3. Включить удаленный рабочий стол с помощью ползунка.

4. Также рекомендуется оставить компьютер в режиме бодрствования

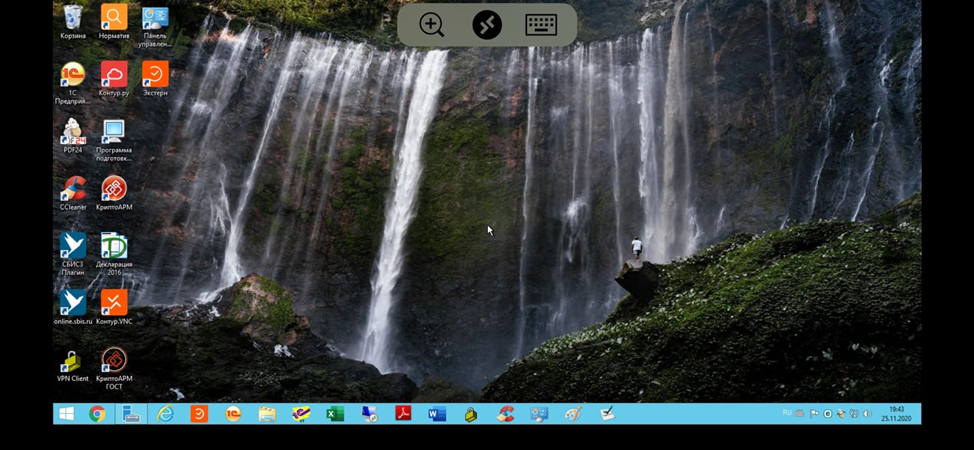
и доступным для обнаружения, чтобы упростить подключение. Щелкните

«Показать параметры» для включения.  
  
После этого необходимо добавить пользователя в группу «администраторы»  
для удаленного подключения к рабочему столу(необходимо, чтобы у учетной записи был пароль)

После этого с помощью консольной команды ipconfig узнаем айпи и далее проверим доступность узла с помощью команды ping. Произведем подключение к с телефона и с Windows.

1 с компьютера



2 с телефона

**3)SSH**В Kali пакет OpennSSH установлен по умолчанию, но автоматический запуск

отключён (как и всех других сетевых служб). В случае его отсутствия требуется

ввести следующие команды:

~$ sudo apt-get install openssh-server -y

Есть два варианта запуска службы SSH:

- ssh.service — служба SSH всегда будет запущена;

- ssh.socket — система откроет для прослушивания 22-й порт и будет

ожидать подключение к нему. В случае подключения, система запустит службу SSH.

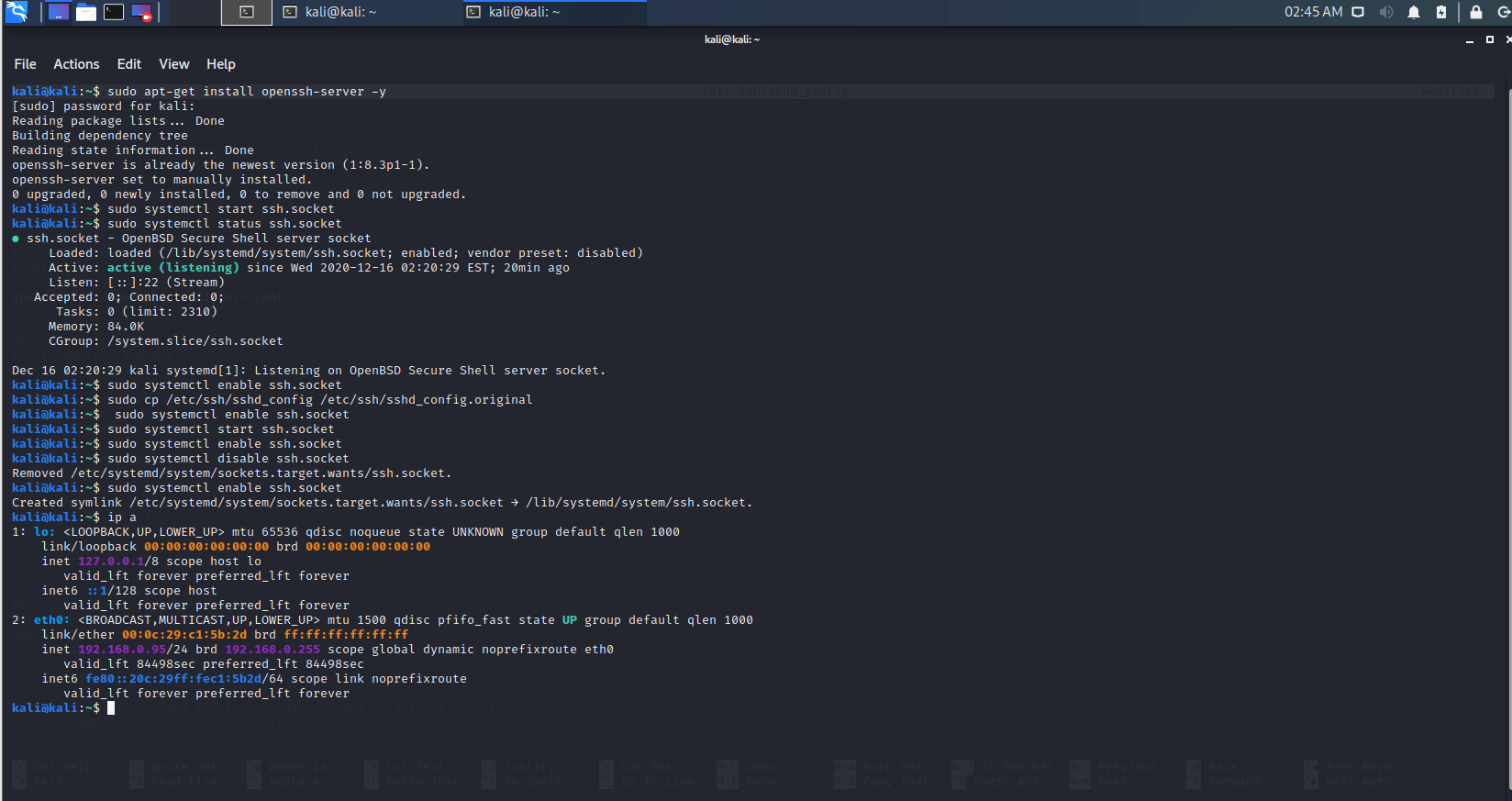
Во всё остальное время (когда нет подключений по SSH), служба SSH будет

отключена и не будет потреблять системные ресурсы.

Чтобы проверить успешное выполнение службы SSH, выполните следующую

команду:

Для того, чтобы служба запускалась автоматически требуется

добавить/удалить службу в автозагрузку.  
~$ sudo systemctl enable ssh.socket / sudo systemctl disable ssh.socket Изменение конфигурации сервера OpenSSH осуществляется с помощью

редактирования файла /etc/ssh/sshd\_config.

Изменим значение порта на 20021.  
По умолчанию аутентификация от имени суперпользователя разрешена,

рекомендуется отключить такую возможность. Для этого требуется найти строчку

PermitRootLogin и заменить ее значение на no.

Чтобы sshd разрешал учетные данные для входа на основе открытого ключа,

требуется добавить или измените строку: PubkeyAuthentication yes.

файле /etc/ssh/sshd\_config требуется указать путь до данного файла.

(К большому сожалению скриншоты не сохранились((( )

После этого внесем изменения в содержимое файла /etc/issue.net. Оно будет отображаться в качестве приветствия, укажем свои Ф.И.О.

(К большому сожалению скриншоты не сохранились((( )

Далее для отображения банера внесем изменения в содержимое файла /etc/motd и запишем там ст.272 п.1 УК РФ.

После внесения изменений в файл /etc/ssh/sshd\_config требуется сохранить

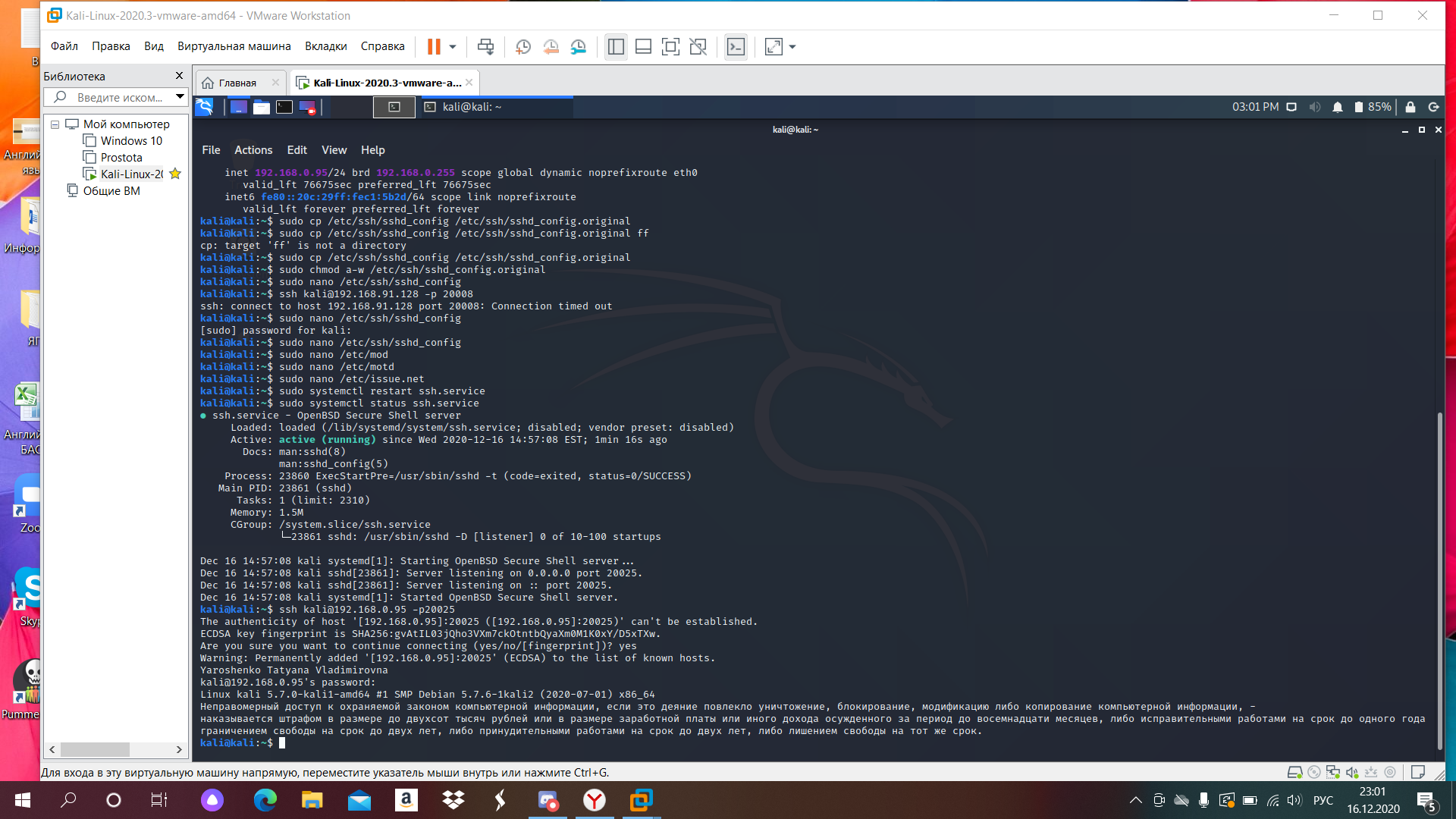
файл и перезапустить приложение-сервер sshd. Для изменения требуется

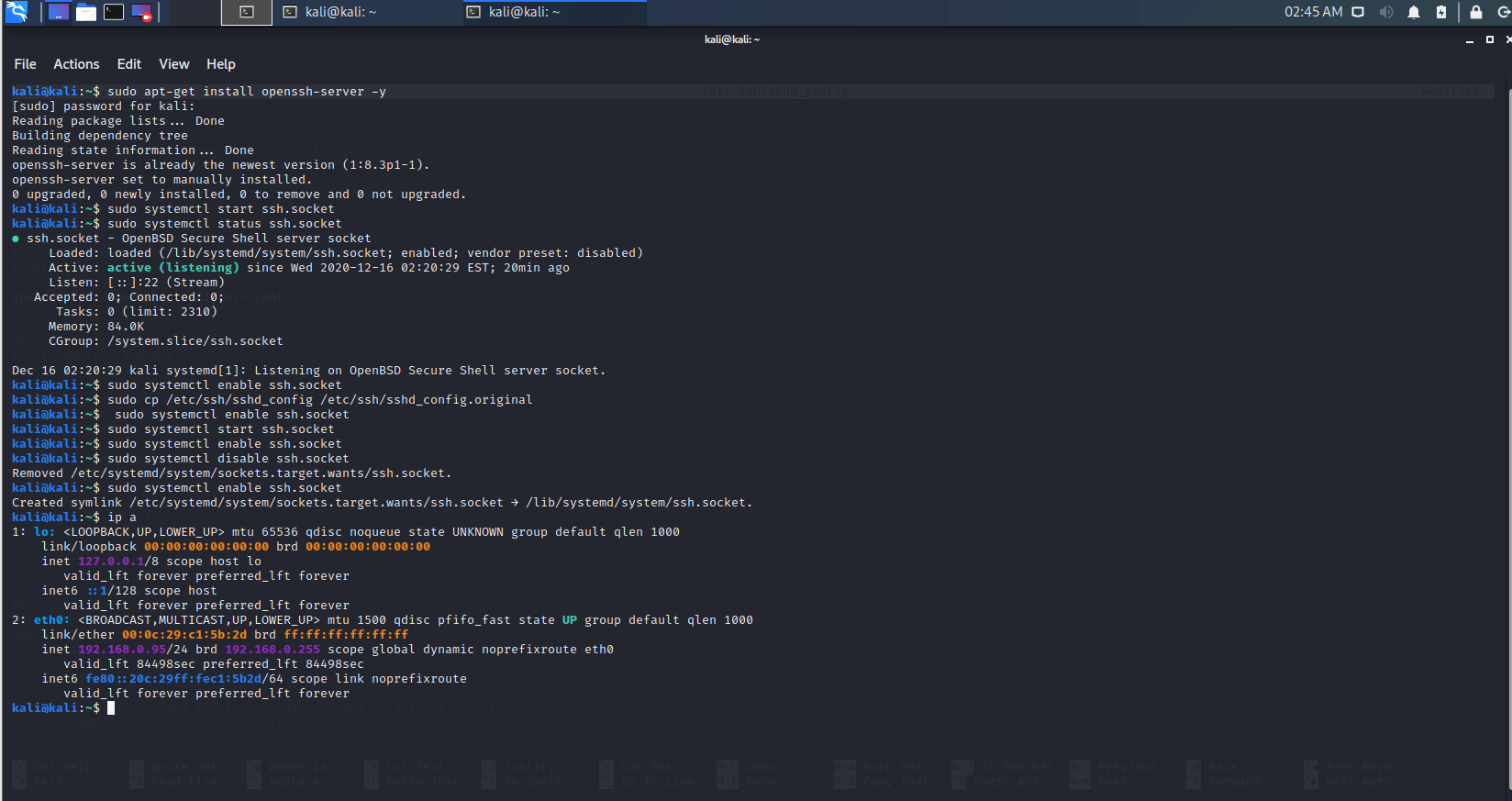
перезапустить сервис ssh.service:  
$ sudo systemctl restart ssh.service  
  
После запуска ssh сервера требуется проверка функционирования данного

сервиса:

С помощью данной команды выполняется попытка соединения ssh клиента

с сервером внутри одной ОС.

$ ssh kali@192.168.0.95 -p 20025

**Подключение к OC Linux:**  
  
Для подключения к удаленному серверу, используется следующая команда: $ ssh kali@192.168.0.95 -p 20025

**Подключение к OC Windows:**Для подключения к SSH используются программы типа PuTTY или Bitvise –

клиентские программы для работы с сетевыми протоколами Telnet, SSH, SCP, SFTP,

для подключения по COM-порту и ZModem. Включают в себя утилиты для

генерации RSA, DSA, ECDSA, Ed25519 цифровых SSH-ключей. PuTTY является

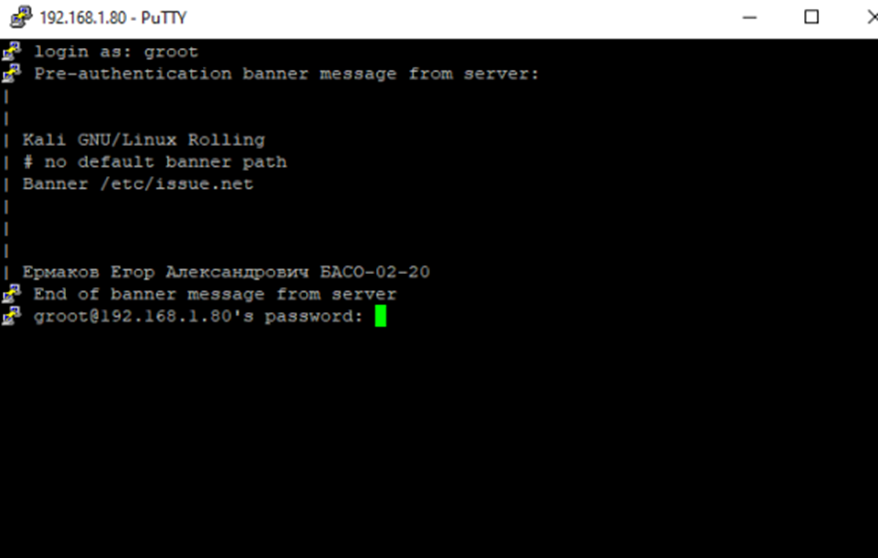
свободным приложением с открытым исходным кодом, содержит реализацию

сетевых протоколов SSH, Telnet, Rlogin, и распространяется под Open Source

лицензией MIT.

Установим программу PuTTY и произведем подключение к пользователю  
(Ермакову Егору).  
После ввода имени пользователя и пароля отобразится терминальное окно с

подключённой сессией.

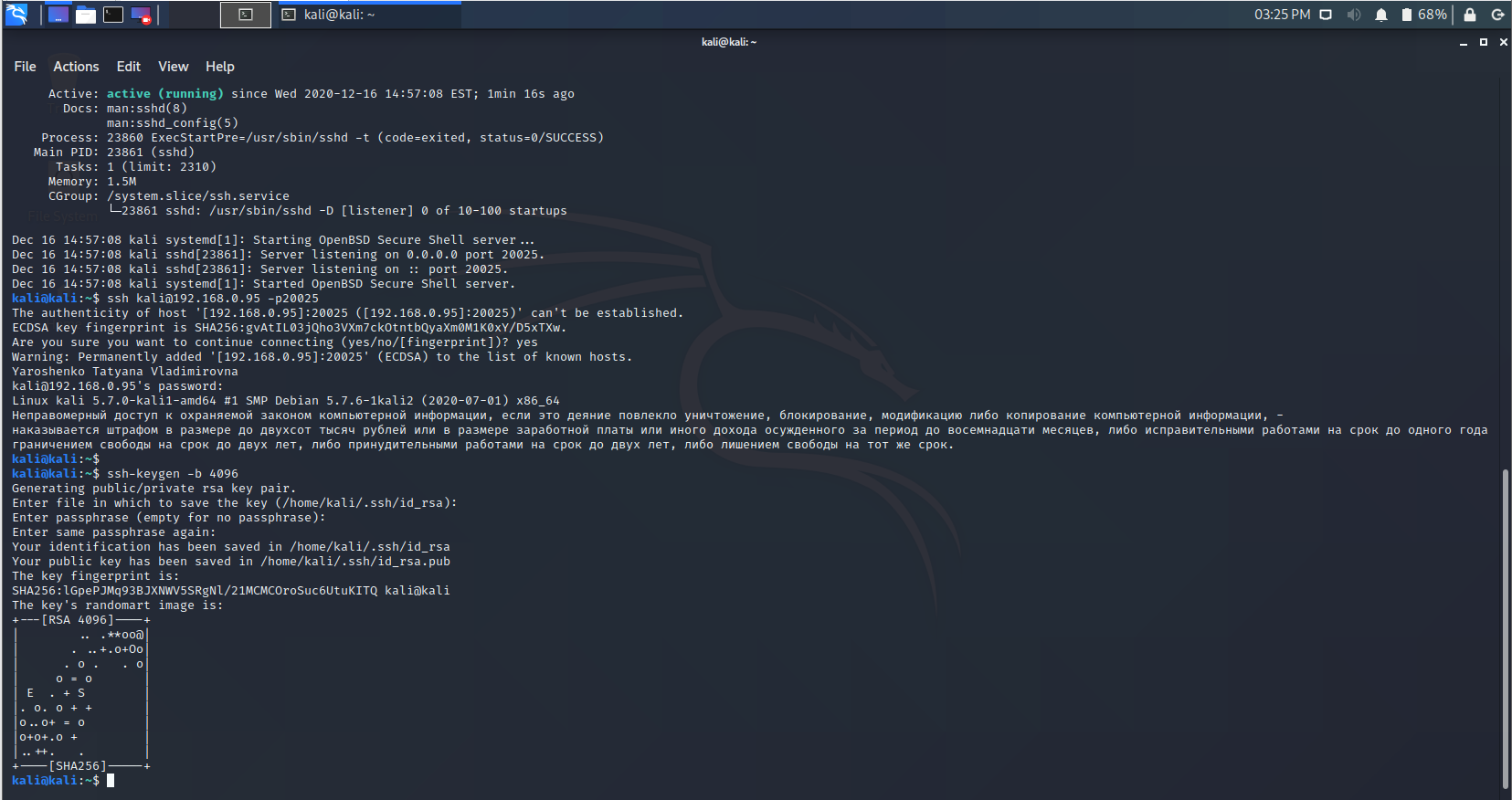




**Доступ с использованием ключа. Генерирование ключа на**

**удалённой системе.** **Подключение с ОС Linux:**

Создадим пару ключей с помощью команды

**$ ssh-keygen -b 4096**Далее имеется возможность задать ключевую фразу (passphrase), что обычно

рекомендуется сделать. Ключевая фраза добавляет дополнительный уровень

безопасности для предотвращения входа на сервер неавторизованных

пользователей.

В результате сгенерируется пара из публичного и приватного ключей, которые

возможно использовать для аутентификации. Далее требуется разместить

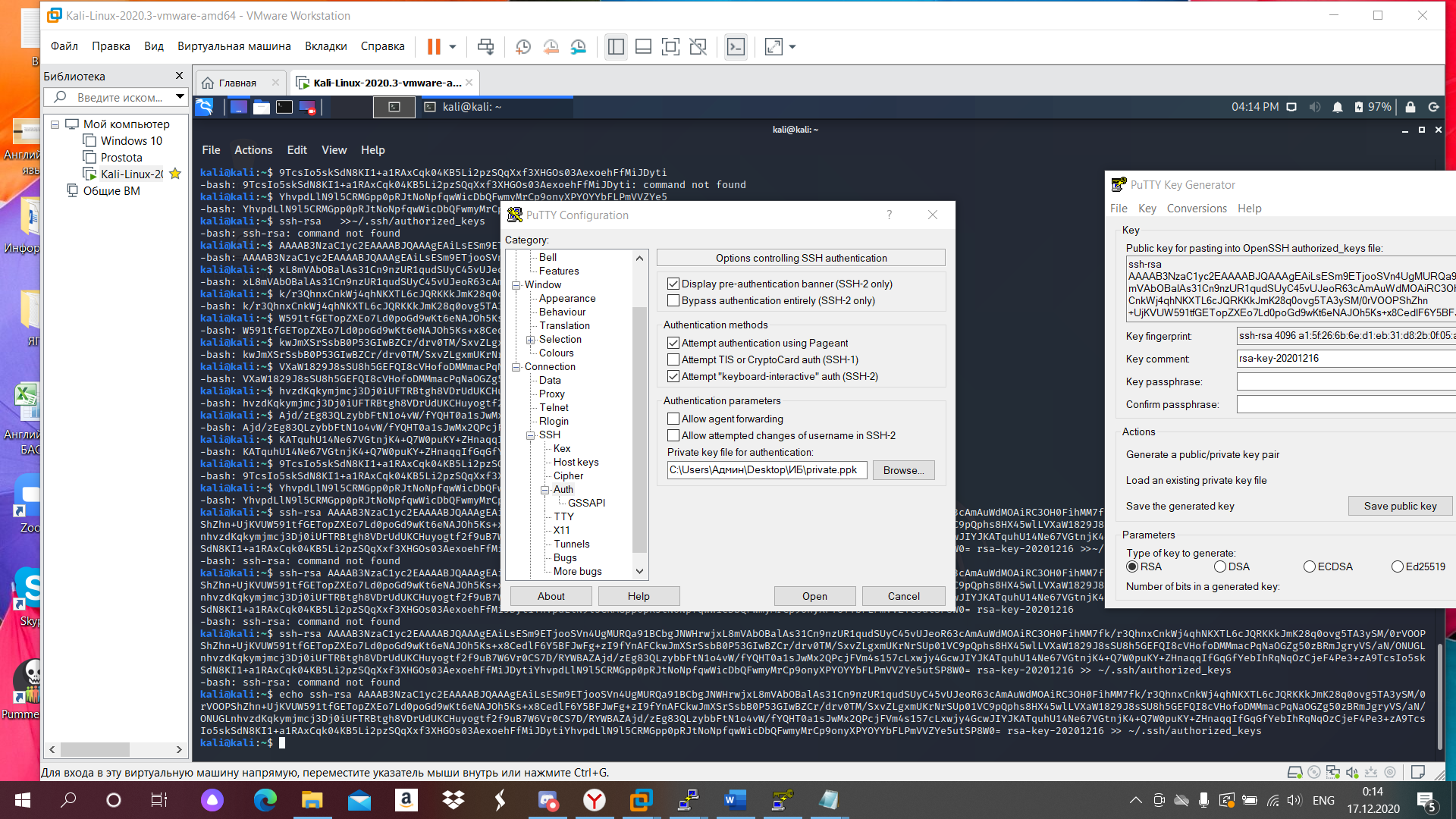
публичный ключ на компьютере пользователя, к которому будет производится

подключение (либо на своём), для того, чтобы была возможность использовать

аутентификацию по ключам SSH для входа.

**Подключение с ОС Windows**Для генерации ключей требуется запустить инструмент PuTTYgen («Пуск»>

PuTTY (64-bit) → PuTTYgen и потом сохранить их.



**Копирование публичного ключа на сервер ОС Linux.**Утилита ssh-copy-id доступна по умолчанию во многих операционных

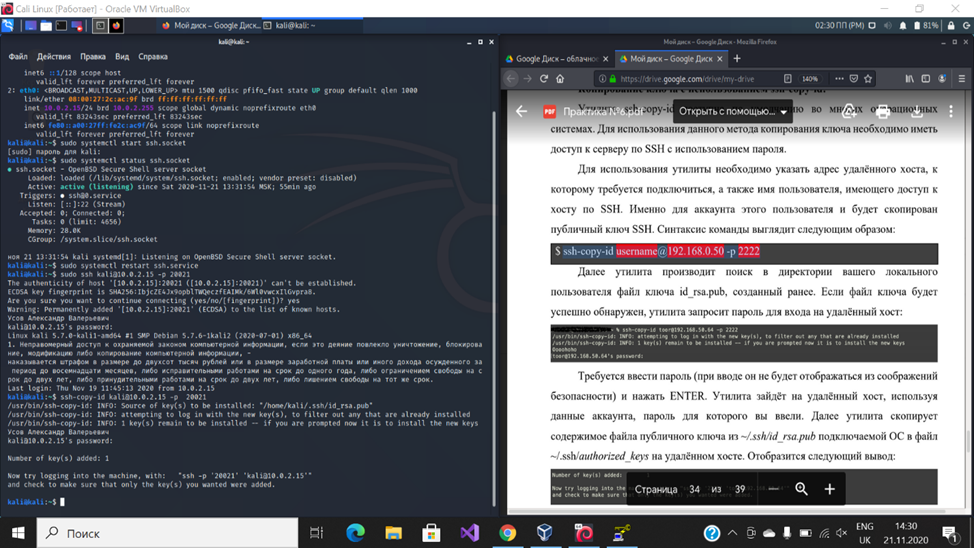
системах. Для использования данного метода копирования ключа необходимо иметь

доступ к серверу по SSH с использованием пароля.

Для использования утилиты необходимо указать адрес удалённого хоста, к

которому требуется подключиться, а также имя пользователя, имеющего доступ к

хосту по SSH. Именно для аккаунта этого пользователя и будет скопирован

публичный ключ SSH. Синтаксис команды выглядит следующим образом:  
$ ssh-copy-id username@ip -p 2222

**Аутентификация на сервере с использованием ключей SSH  
 OC Linux**Если при создании пары ключей не была задана ключевая фраза (passphrase),

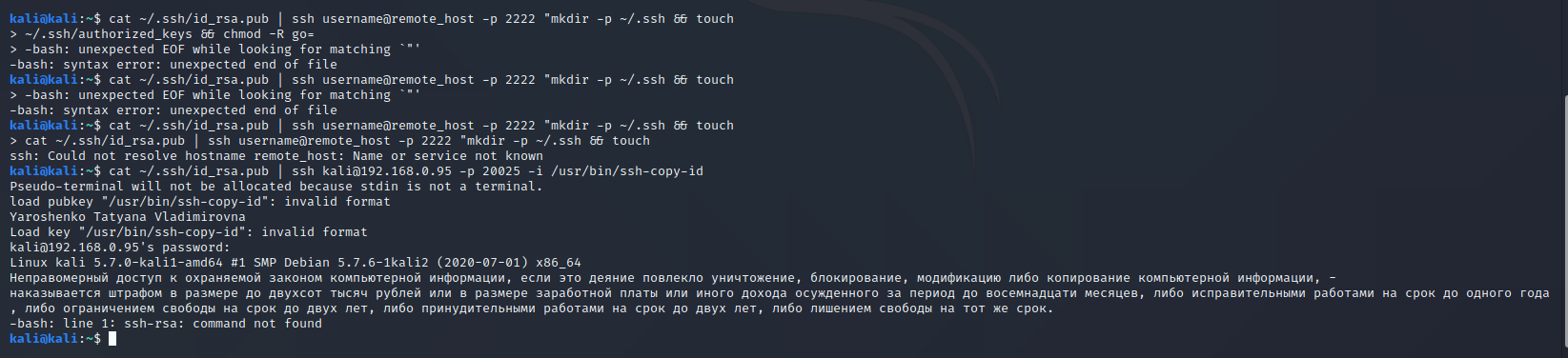
авторизация произойдёт автоматически.  
$ ssh username@ip -p 20021 -i путь до ключа

Если задана ключевая фраза, то будет предложено её ввести (обратите

внимание, что вводимые символы не будут отображаться на экране в целях

безопасности). После аутентификации откроется новая сессия оболочки (shell

session) на удалённом хосте от имени используемого вами удалённого аккаунта

пользователя.  
 **ОС Windows**Для передачи открытого ключа, используя putty, требуется скопировать

значение ключа из окна «Public key for pasting into OpenSSH authorized\_keys file:» в Putty Key Generator.

Далее требуется выполнить подключение с удалённому серверу по паролю.

После подключения требуется открыть/создать файл ~/.ssh/authorized\_keys.  
sudo nano .ssh/authorized\_keys

Далее с помощью команды требуется вставить скопированный ключ   
$ echo публичный\_ключ >> ~/.ssh/authorized\_keys

В команде требуется заменить «публичный\_ключ» на значение

скопированного публичного ключа из PuttyKeyGenerator. Строка должна начинаться

с ssh-rsa AAAA....

После настройки ssh сервера требуется указать в настройках подключения

путь к приватному ключу.  
После выполненных заданий, подключение должно производиться без ввода

пароля.(Видно на скрине выше)

**Отключение аутентификации по паролю на сервере**Если удалось войти в удалённый аккаунт на удалённом хосте по SSH без ввода

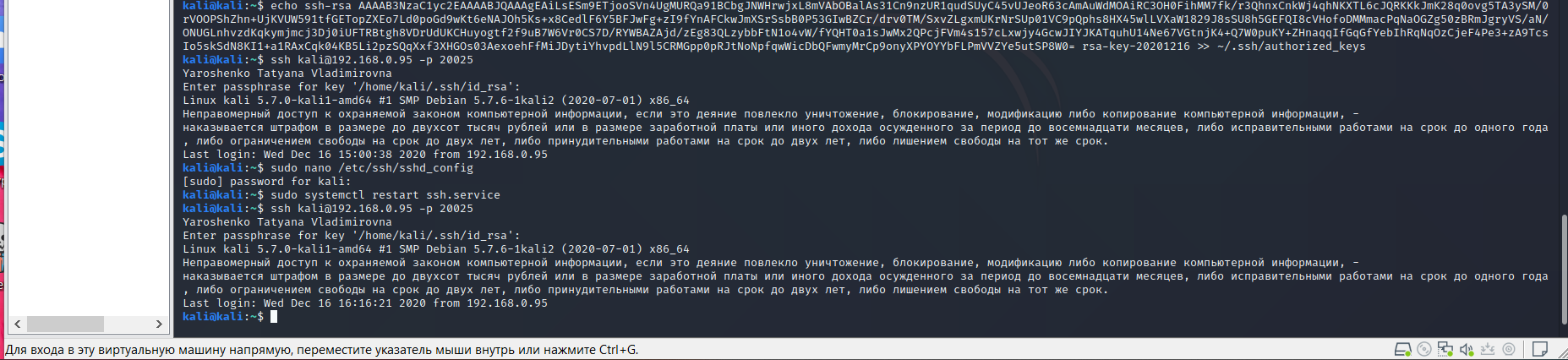
пароля, следовательно аутентификацию по ключу SSH для вашего аккаунта успешна

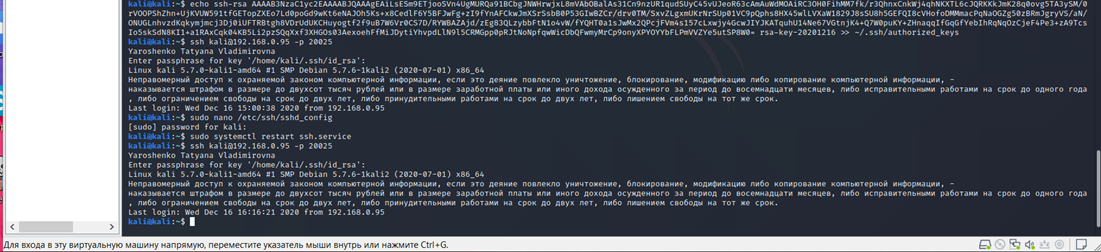
настроена. Однако возможность входить на сервер с использованием пароля всё есть

активна, что означает, что сервер уязвим для атак с перебором пароля (brute-force

attacks). Требуется открыть конфигурационный файл демона SSH:  
$ sudo nano /etc/ssh/sshd\_config  
Внутри файла требуется найти директиву PasswordAuthentication. Она может

быть закомментирована. Раскомментируйте её при необходимости и установите её

значение в «no». Данное действие отключит возможность входа на сервер по паролю.

После этого необходимо перезагрузить сервер:  
$ sudo systemctl restart ssh.service  
После этого отключиться возможность захода на сервер по паролю.  
**Создание тестового сервера Google**

Следуя инструкции из практической работы, я создала тестовый сервер Google.

