**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра теории вероятностей и кибербезопасности**

**ОТЧЁТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №11**

*дисциплина: Администрирование сетевых подсистем*

Студент: Махорин Иван Сергеевич

Студ. билет № 1032211221

Группа: НПИбд-02-21

**МОСКВА**

2023 г.

# Цель работы:

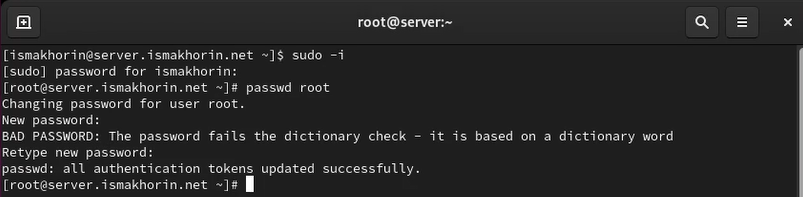
# Целью данной работы является приобретение практических навыков по настройке удалённого доступа к серверу с помощью SSH.

**Выполнение работы:**

На сервере зададим пароль для пользователя root (Рис. 1.1):

sudo -i

passwd root

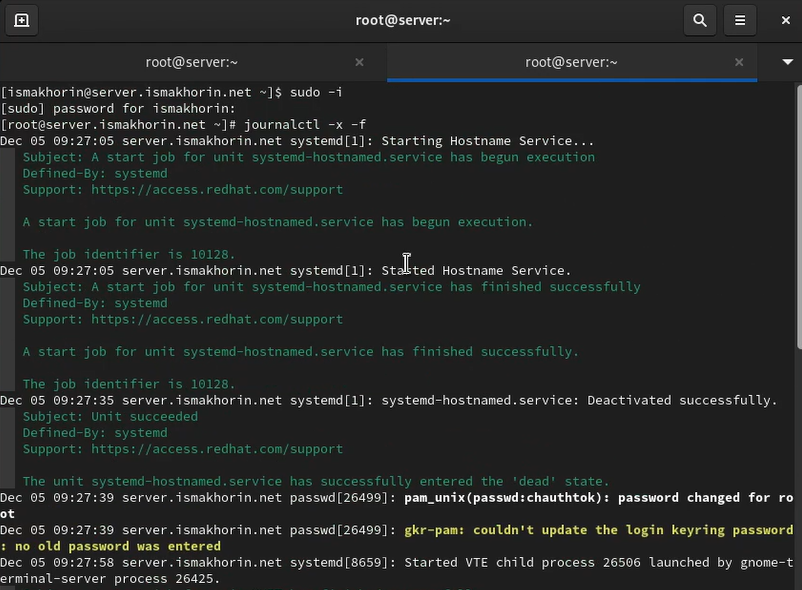
****

**Рис. 1.1.** Открытие режима суперпользователя на виртуальной машине server и создание пароля для пользователя root.

На сервере в дополнительном терминале запустим мониторинг системных событий (Рис. 1.2):

sudo -i

journalctl -x -f

****

**Рис. 1.2.** Запуск в дополнительном терминале мониторинга системных событий.

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя root (Рис. 1.3):

ssh root@server.ismakhorin.net

****

**Рис. 1.3.** Попытка получить с клиента доступ к серверу посредством SSH-соединения через пользователя root.

На сервере откроем файл /etc/ssh/sshd\_config конфигурации sshd для редактирования и запретим вход на сервер пользователю root, установив (Рис. 1.4):

PermitRootLogin no

****

**Рис. 1.4.** Открытие на сервере файла /etc/ssh/sshd\_config конфигурации sshd для редактирования и запрет входа на сервер пользователю root.

После сохранения изменений в файле конфигурации перезапустим sshd (Рис. 1.5):

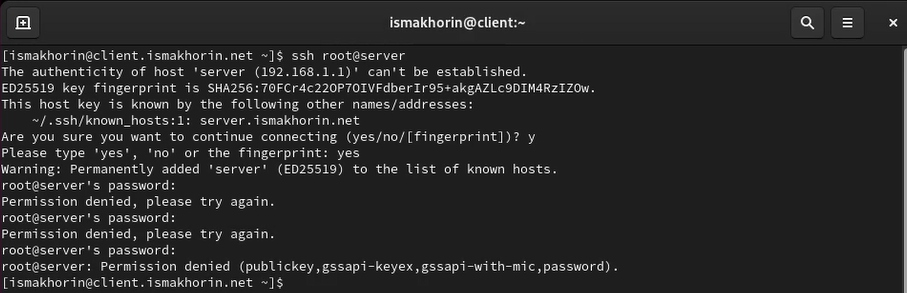
systemctl restart sshd

****

**Рис. 1.5.** Перезапуск sshd.

Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя root (Рис. 1.6):

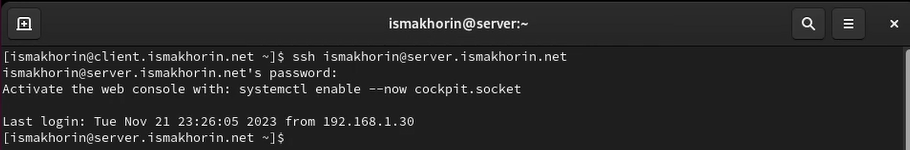
ssh root@server

****

**Рис. 1.6.** Повторная попытка получения доступа с клиента к серверу посредством SSH-соединения через пользователя root.

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя ismakhorin (Рис. 2.1):

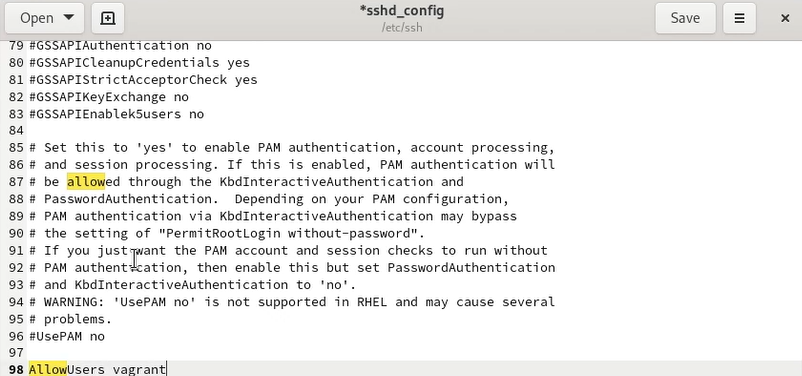
ssh ismakhorin@server.ismakhorin.net

****

**Рис. 2.1.** Попытка получить с клиента доступ к серверу посредством SSH-соединения через пользователя ismakhorin.

На сервере откроем файл /etc/ssh/sshd\_config конфигурации sshd на редактирование и добавим строку (Рис. 2.2):

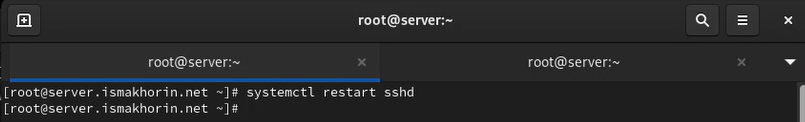
AllowUsers vagrant

****

**Рис. 2.2.** Открытие на сервере файла /etc/ssh/sshd\_config конфигурации sshd на редактирование и добавление нужной строки.

После сохранения изменений в файле конфигурации перезапустим sshd (Рис. 2.3):

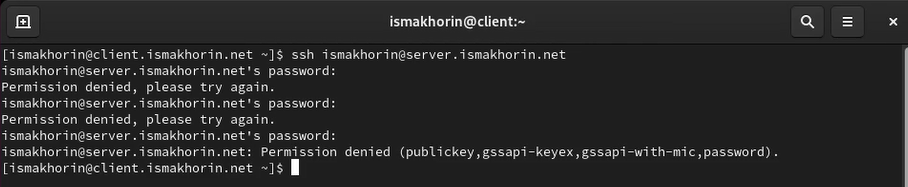
systemctl restart sshd

****

**Рис. 2.3.** Перезапуск sshd.

Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя ismakhorin (Рис. 2.4):

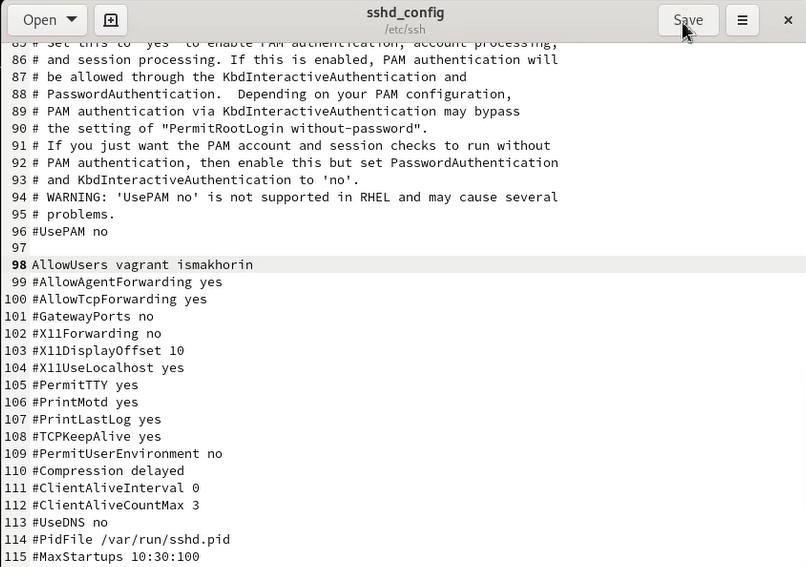
ssh ismakhorin@server.ismakhorin.net

****

**Рис. 2.4.** Повторная попытка получения доступа с клиента к серверу посредством SSH-соединения через пользователя ismakhorin.

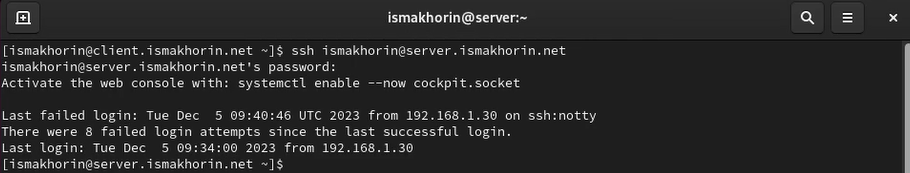
В файле /etc/ssh/sshd\_config конфигурации sshd внесём следующее изменение (Рис. 2.5):

AllowUsers vagrant ismakhorin

****

**Рис. 2.5.** Внесение изменения в файле /etc/ssh/sshd\_config конфигурации sshd.

После сохранения изменений в файле конфигурации перезапустим sshd и вновь попытаемся получить доступ с клиента к серверу посредством SSH-соединения через пользователя ismakhorin (Рис. 2.6):

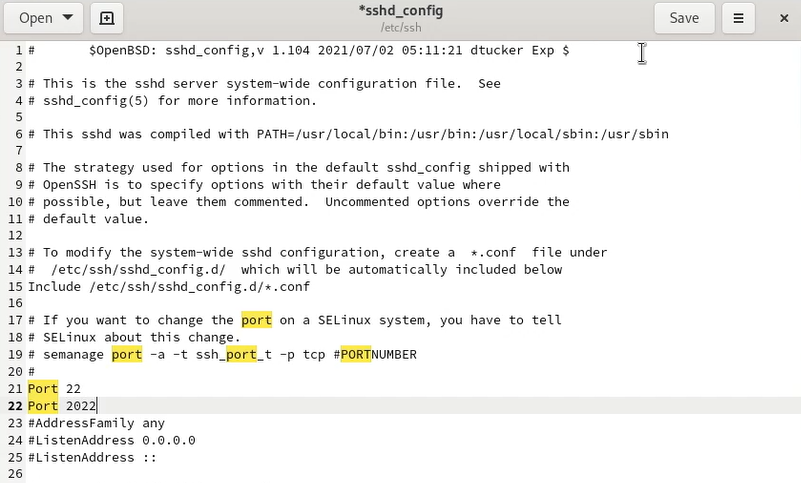
****

**Рис. 2.6.** Перезапуск sshd и повторная попытка получить доступ с клиента к серверу посредством SSH-соединения через пользователя ismakhorin.

На сервере в файле конфигурации sshd /etc/ssh/sshd\_config найдём строку Port и ниже этой строки добавим (Рис. 3.1):

Port 22

Port 2022

****

**Рис. 3.1.** Добавление ниже строки Port записей в файле конфигурации sshd /etc/ssh/sshd\_config на сервере.

После сохранения изменений в файле конфигурации перезапустим sshd:

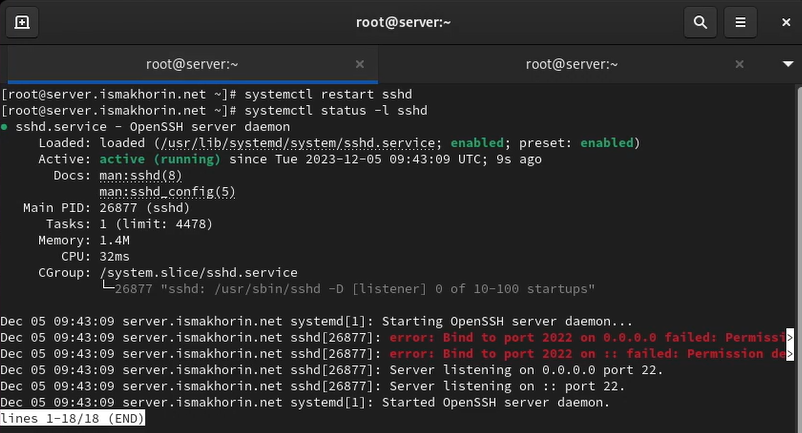
systemctl restart sshd

И посмотрим расширенный статус работы:

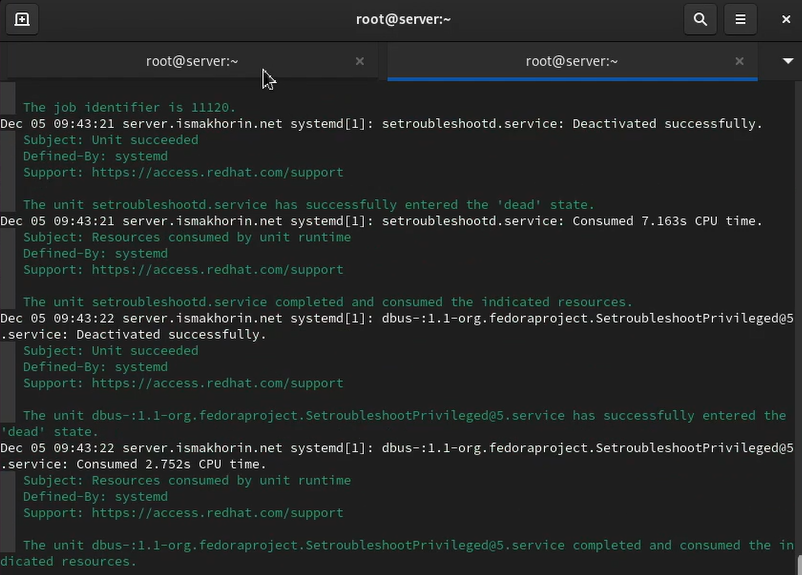
systemctl status -l sshd

Система сообщила нам об отказе в работе sshd через порт 2022 (Рис. 3.2):

Дополнительно посмотрим сообщения в терминале с мониторингом системных событий (Рис. 3.3):

****

**Рис. 3.2.** Перезапуск sshd и просмотр расширенного статуса работы.

****

**Рис. 3.3.** Просмотр сообщения в терминале с мониторингом системных событий.

Исправим на сервере метки SELinux к порту 2022:

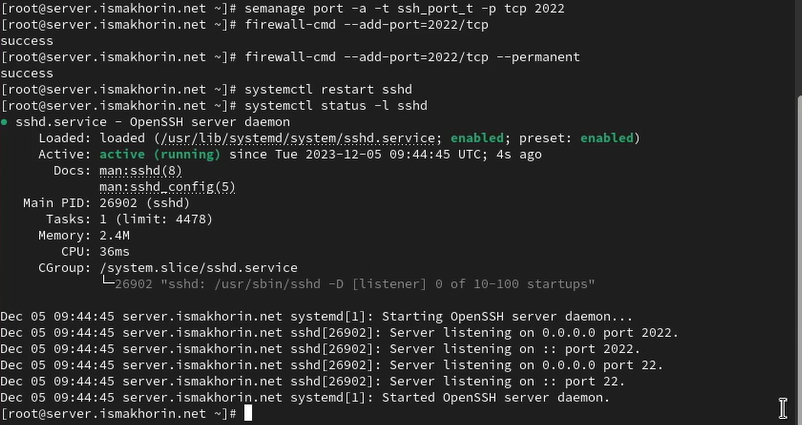
semanage port -a -t ssh\_port\_t -p tcp 2022

В настройках межсетевого экрана откроем порт 2022 протокола TCP:

firewall-cmd --add-port=2022/tcp

firewall-cmd --add-port=2022/tcp --permanent

Вновь перезапустим sshd и посмотрим расширенный статус его работы (статус показывает, что процесс sshd теперь прослушивает два порта) (Рис. 3.4):

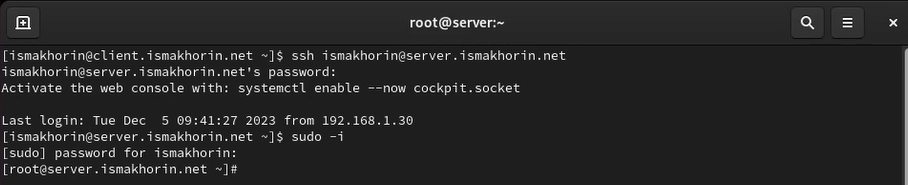
****

**Рис. 3.4.** Исправление на сервере метки SELinux к порту 2022, открытие в настройках межсетевого порта 2022 протокола TCP, повторный перезапуск sshd и просмотр расширенного статуса его работы.

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя ismakhorin:

ssh ismakhorin@server.ismakhorin.net

После открытия оболочки пользователя введём sudo -i для получения доступа root (Рис. 3.5):

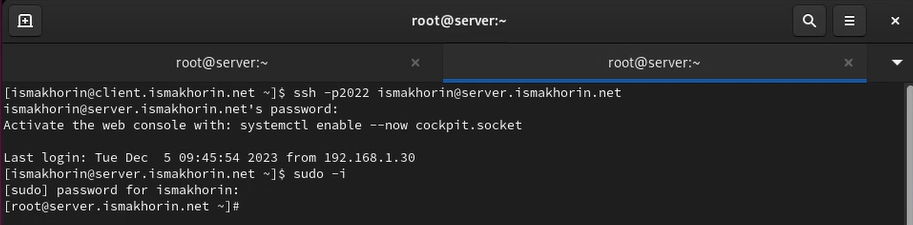
****

**Рис. 3.5.** Попытка получить с клиента доступа к серверу посредством SSH-соединения через пользователя ismakhorin и получение доступа root.

Теперь повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя ismakhorin, указав порт 2022:

ssh -p2022 ismakhorin@server.ismakhorin.net

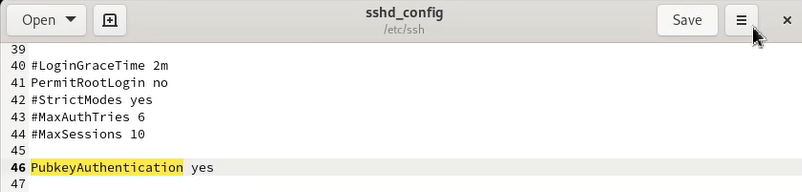
После открытия оболочки пользователя введём sudo -i для получения доступа root (рис. 3.6):

****

**Рис. 3.6.** Повторная попытка получения доступа с клиента к серверу посредством SSH-соединения через пользователя ismakhorin, указав порт 2022. Получение доступа root.

На сервере в конфигурационном файле /etc/ssh/sshd\_config зададим параметр, разрешающий аутентификацию по ключу (рис. 4.1):

PubkeyAuthentication yes

****

**Рис. 4.1.** Настройка параметра на сервере в конфигурационном файле /etc/ssh/sshd\_config, разрешающего аутентификацию по ключу.

После сохранения изменений в файле конфигурации перезапустим sshd (рис. 4.2):

****

**Рис. 4.2.** Перезапуск sshd.

На клиенте сформируем SSH-ключ, введя в терминале под пользователем ismakhorin

ssh-keygen

Далее скопируем открытый ключ на сервер, введя на клиенте (рис. 4.3):

ssh-copy-id ismakhorin@server.ismakhorin.net

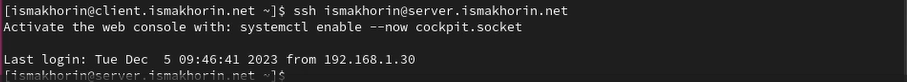
****

**Рис. 4.3.** Формирование на клиенте SSH-ключа и копирование открытого ключа на сервер.

Попробуем получить доступ с клиента к серверу посредством SSH-соединения:

ssh ismakhorin@server.ismakhorin.net

Теперь мы проходим аутентификацию без ввода пароля для учётной записи удалённого пользователя (рис. 4.4):

****

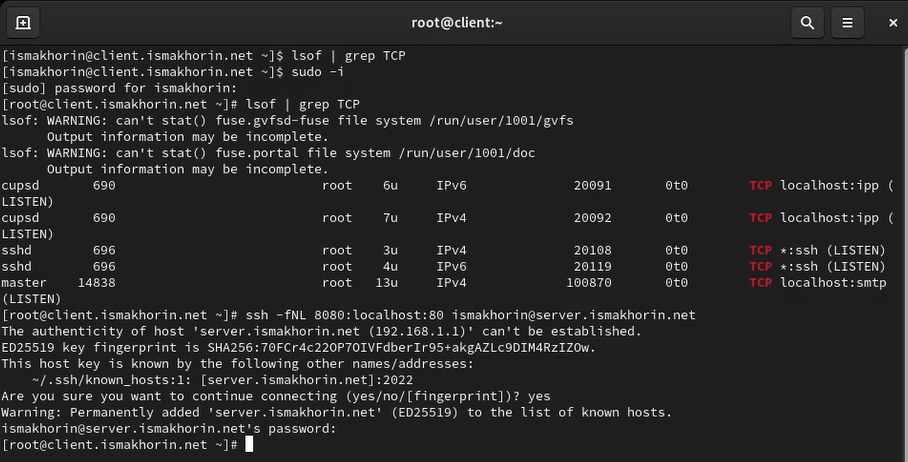
**Рис. 4.4.** Попытка получения доступа с клиента к серверу посредством SSH-соединения.

На клиенте посмотрим, запущены ли какие-то службы с протоколом TCP:

lsof | grep TCP

После чего перенаправим порт 80 на server.ismakhorin.net на порт 8080 на локальной машине (рис. 5.1):

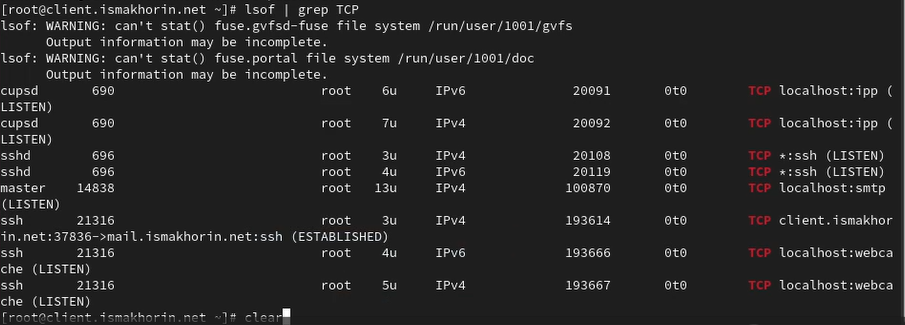
ssh -fNL 8080:localhost:80 ismakhorin@server.ismakhorin.net

****

**Рис. 5.1.** Просмотр на клиенте запущенных служб с протоколом TCP и перенаправление порта 80 на server.ismakhorin.net на порт 8080 на локальной машине.

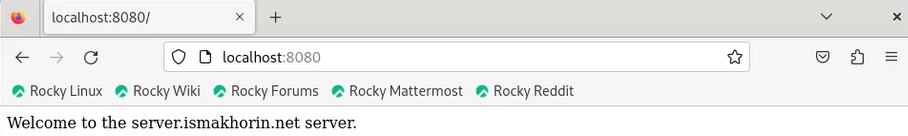
Вновь на клиенте посмотрим, запущены ли какие-то службы с протоколом TCP (рис. 5.2):

lsof | grep TCP

****

**Рис. 5.2.** Повторный просмотр на клиенте запущенных служб с протоколом TCP.

На клиенте запустим браузер и в адресной строке введём localhost:8080. Убедимся, что отобразилась страница с приветствием «Welcome to the server.ismakhorin.net server» (Рис. 5.3):

****

**Рис. 5.3.** Запуск на клиенте браузера и ввод в адресной строке localhost:8080.

На клиенте откроем терминал под пользователем ismakhorin и посмотрим с клиента имя узла сервера:

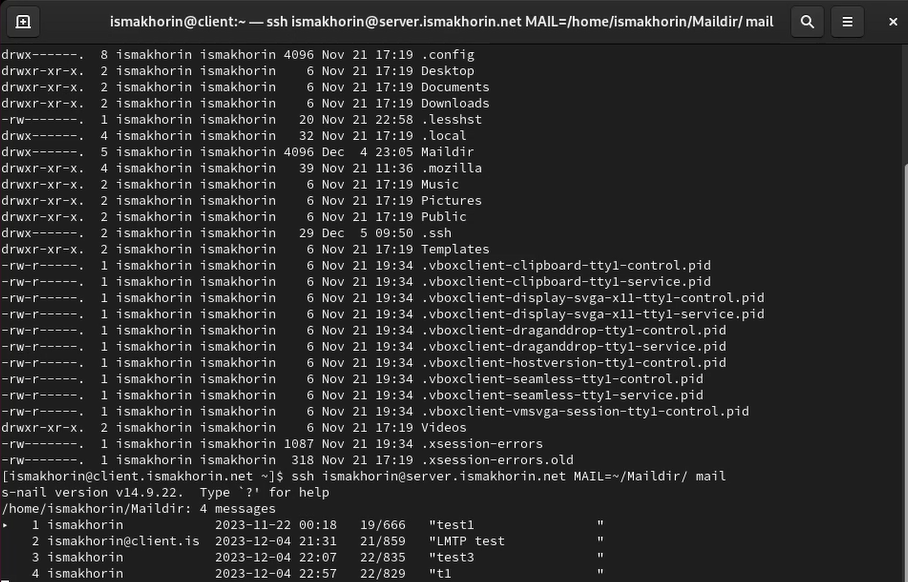
ssh ismakhorin@server.ismakhorin.net hostname

Посмотрим с клиента список файлов на сервере:

ssh ismakhorin@server.ismakhorin.net ls -Al

Посмотрим с клиента почту на сервере (рис. 6):

ssh ismakhorin@server.ismakhorin.net MAIL=~/Maildir/ mail

****

**Рис. 6.** Открытие на клиенте терминала под пользователем ismakhorin. Просмотр имени узла сервера, списка файлов на сервере и почты на сервере.

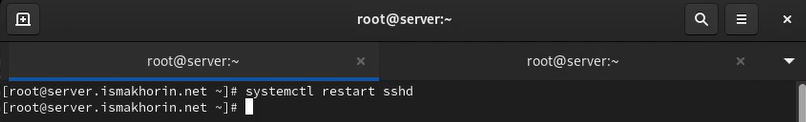
На сервере в конфигурационном файле /etc/ssh/sshd\_config разрешим отображать на локальном клиентском компьютере графические интерфейсы X11 (рис. 7.1):

X11Forwarding yes

****

**Рис. 7.1.** Разрешение отображать на сервере в конфигурационном файле /etc/ssh/sshd\_config на локальном клиентском компьютере графические интерфейсы X11.

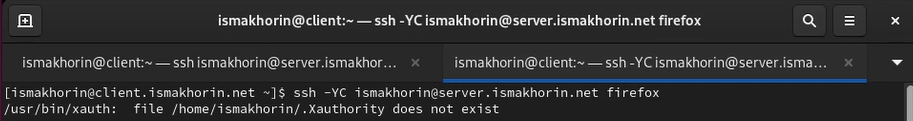
После сохранения изменения в конфигурационном файле перезапустим sshd (рис. 7.2):

****

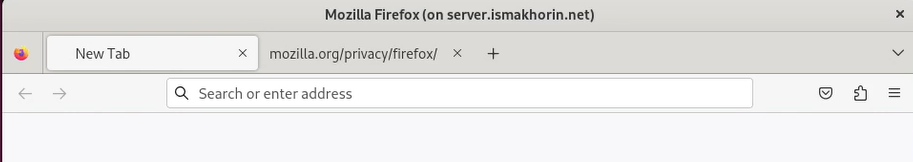
**Рис. 7.2.** Перезапуск sshd.

Попробуем с клиента удалённо подключиться к серверу и (рис. 7.3) запустить графическое приложение firefox (рис. 7.4):

ssh -YC ismakhorin@server.ismakhorin.net firefox

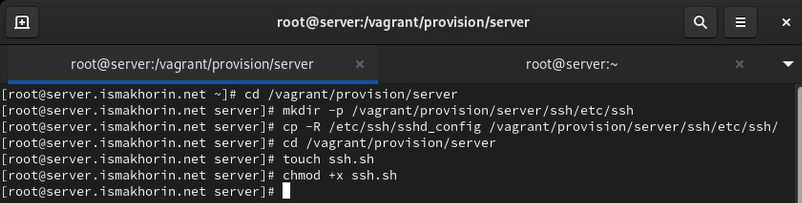
****

**Рис. 7.3.** Попытка с клиента удалённо подключиться к серверу и запустить графическое приложение firefox.

****

**Рис. 7.4.** Запуск графического приложения firefox.

На виртуальной машине server перейдём в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создадим в нём каталог ssh, в который поместим в соответствующие подкаталоги конфигурационный файл sshd\_config. В каталоге /vagrant/provision/server создадим исполняемый файл ssh.sh (рис. 8.1):

****

**Рис. 8.1.** Переход на виртуальной машине server в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создание в нём каталога ssh, в который поместили в соответствующие подкаталоги конфигурационный файл sshd\_config. Создание в каталоге /vagrant/provision/server исполняемого файла ssh.sh.

Открыв его на редактирование, пропишем в нём скрипт из лабораторной работы (Рис. 8.2):

****

**Рис. 8.2.** Открытие файла на редактирование и написание в нём скрипта.

Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile добавим в разделе конфигурации для сервера (рис. 8.3):

****

**Рис. 8.3.** Редактирование конфигурационного файла Vagrantfile.

**Вывод:**

# В ходе выполнения лабораторной работы были приобретены практические навыки по настройке удалённого доступа к серверу с помощью SSH.

**Ответы на контрольные вопросы:**

1. Вы хотите запретить удалённый доступ по SSH на сервер пользователю root и разрешить доступ пользователю alice. Как это сделать? **–**

**В конфигурационном файле SSH /etc/ssh/sshd\_config:**

**# Запрет удалённого доступа пользователю root**

**PermitRootLogin no**

**# Разрешение доступа пользователю alice**

**AllowUsers alice**

**После внесения изменений, необходимо перезапустить службу SSH:**

**sudo service ssh restart**

1. Как настроить удалённый доступ по SSH через несколько портов? Для чего это может потребоваться? **–**

**В конфигурационном файле /etc/ssh/sshd\_config добавьте строки:**

**# Первый порт (по умолчанию 22)**

**Port 22**

**# Второй порт**

**Port 2222**

**После изменений перезапустите службу SSH. Это может быть полезно для повышения безопасности, а также для избежания конфликтов с другими службами, использующими порт 22.**

1. Какие параметры используются для создания туннеля SSH, когда команда ssh устанавливает фоновое соединение и не ожидает какой-либо конкретной команды? **-**

**ssh -N -f -L local\_port:destination\_host:remote\_port user@ssh\_server**

**-N: Не выполнять команду на удаленном хосте.**

**-f: Перевести ssh в фоновый режим после установки туннеля.**

1. Как настроить локальную переадресацию с локального порта 5555 на порт 80 сервера server2.example.com? **–**

**ssh -L 5555:server2.example.com:80 user@ssh\_server**

**Теперь, при подключении к локальному порту 5555, трафик будет перенаправляться через SSH к порту 80 на сервере server2.example.com.**

1. Как настроить SELinux, чтобы позволить SSH связываться с портом 2022? **–**

**sudo semanage port -a -t ssh\_port\_t -p tcp 2022**

**Данная команда добавляет правило SELinux, разрешая использование порта 2022 для сервиса ssh.**

1. Как настроить межсетевой экран на сервере, чтобы разрешить входящие подключения по SSH через порт 2022? **–**

**sudo firewall-cmd --permanent --add-port=2022/tcp**

**sudo firewall-cmd --reload**

**Эти команды добавляют правило в межсетевой экран для разрешения входящих подключений по SSH через порт 2022 и перезагружают конфигурацию межсетевого экрана.**