Отчёт по лабораторной работе №11

Администрирование сетевых подсистем

Ищенко Ирина НПИбд-02-22

Содержание

# 1 Цель работы

Приобретение практических навыков по настройке удалённого доступа к серверу с помощью SSH.

# 2 Выполнение лабораторной работы

На сервере откроем терминал и перейдем в режим суперпользователя. Установим пароль для root пользователя.

В дополнительном терминале запустим мониторинг системных событий с помощью команды journalctl -x -f. С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя root(рис. fig. 1):

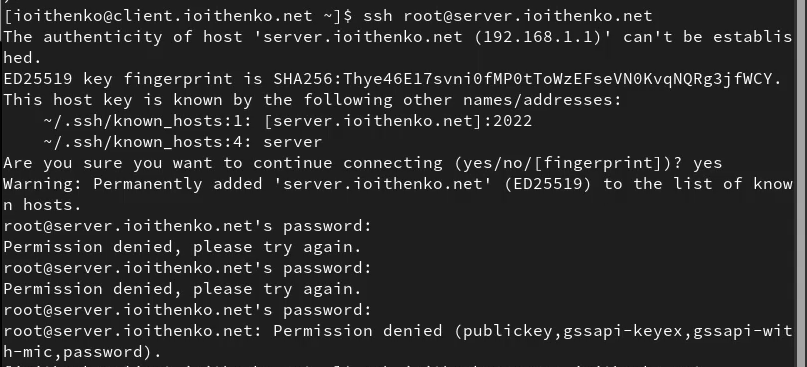


Рис. 1: Попытка установить SSH-соединение

При попытке соединения, так как мы делаем это первый раз, добавлям сервер в список известных хостов. Затем требуется ввести пароль от пользователя root, но соединение отклоняется.

На сервере откроем файл /etc/ssh/sshd\_config конфигурации sshd для редактирования и запретим вход на сервер пользователю root, установив(рис. fig. 2):

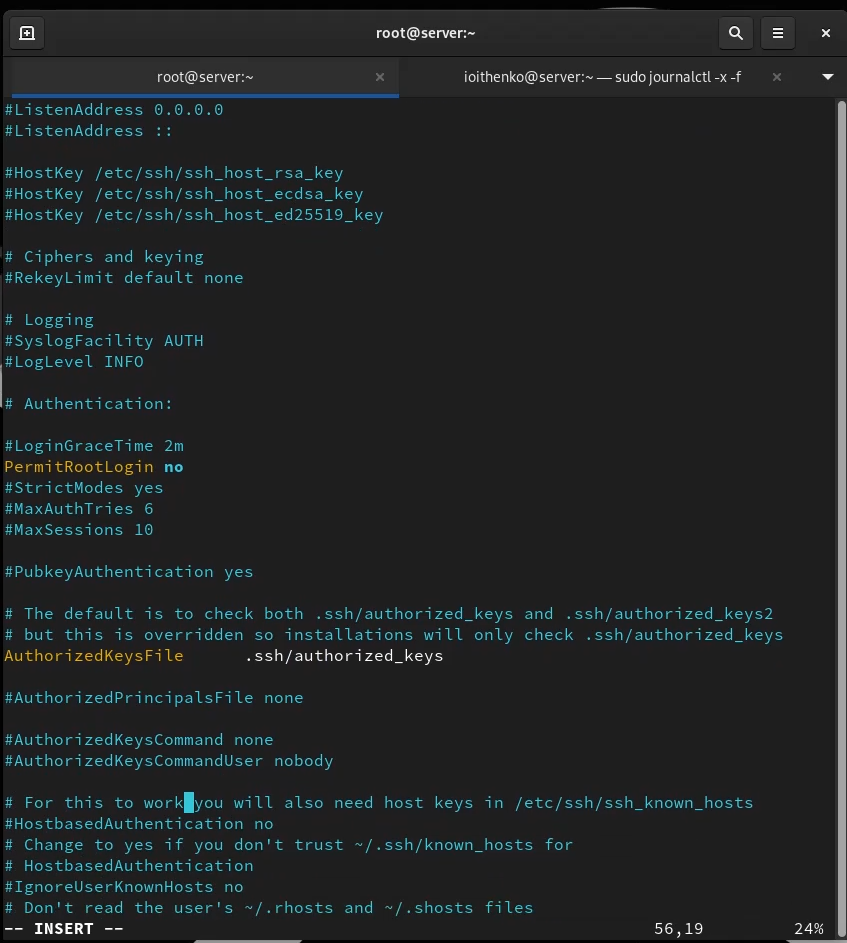


Рис. 2: Запрет входа на сервер пользователю root

После сохранения изменений в файле конфигурации перезапустим sshd с помощью команды systemctl restart sshd. Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя root(fig. 3):

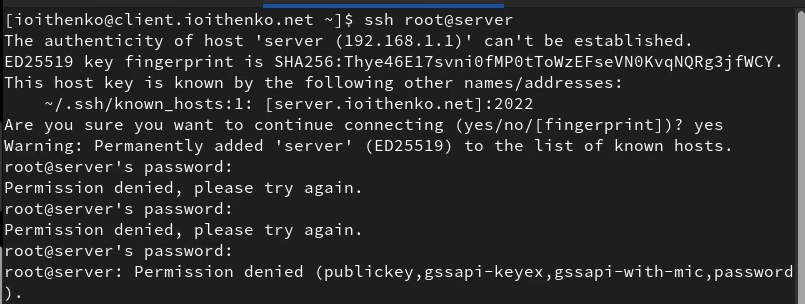


Рис. 3: Повторная попытка SSH-соединение

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя ioithenko(рис. fig. 4 и fig. 5):

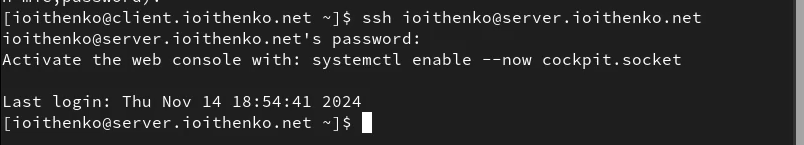


Рис. 4: Попытка установить SSH-соединение с клиента

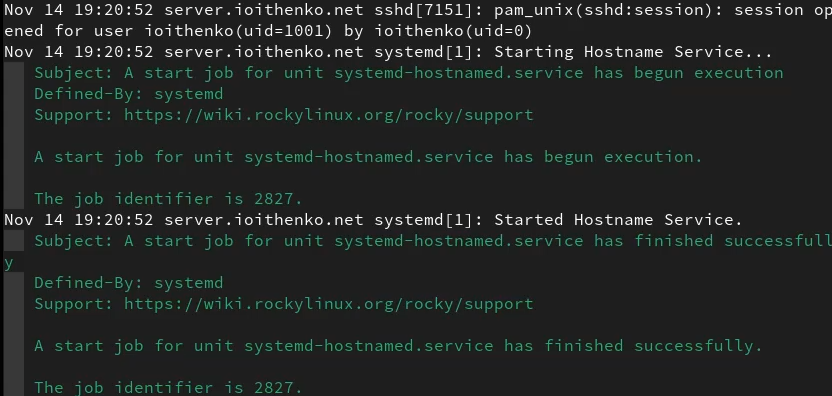


Рис. 5: Мониторинг

Соединениие проходит удачно.

На сервере откроем файл /etc/ssh/sshd\_config конфигурации sshd на редактирование и добавим строку(fig. 6):

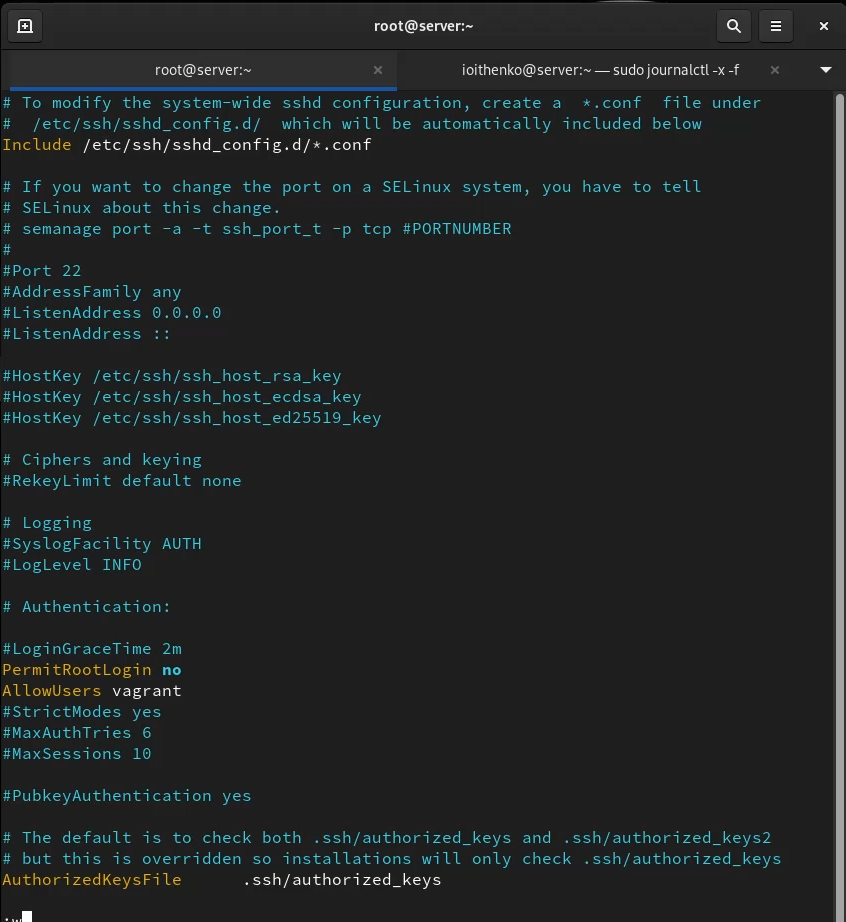


Рис. 6: Изменение разрешенных пользователей для sshd

После сохранения изменений в файле конфигурации перезапустим sshd.

Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя ioithenko(рис. fig. 7):

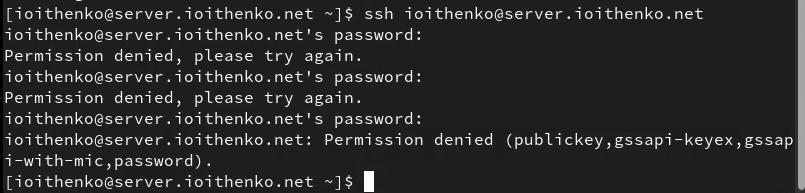


Рис. 7: Определение службы аутентификации пользователей

В этот раз соединение не устанавливается, так как в списке разрешенных пользователей нет нашего.

В файле /etc/ssh/sshd\_config конфигурации sshd внесем следующее изменение(fig. 8):

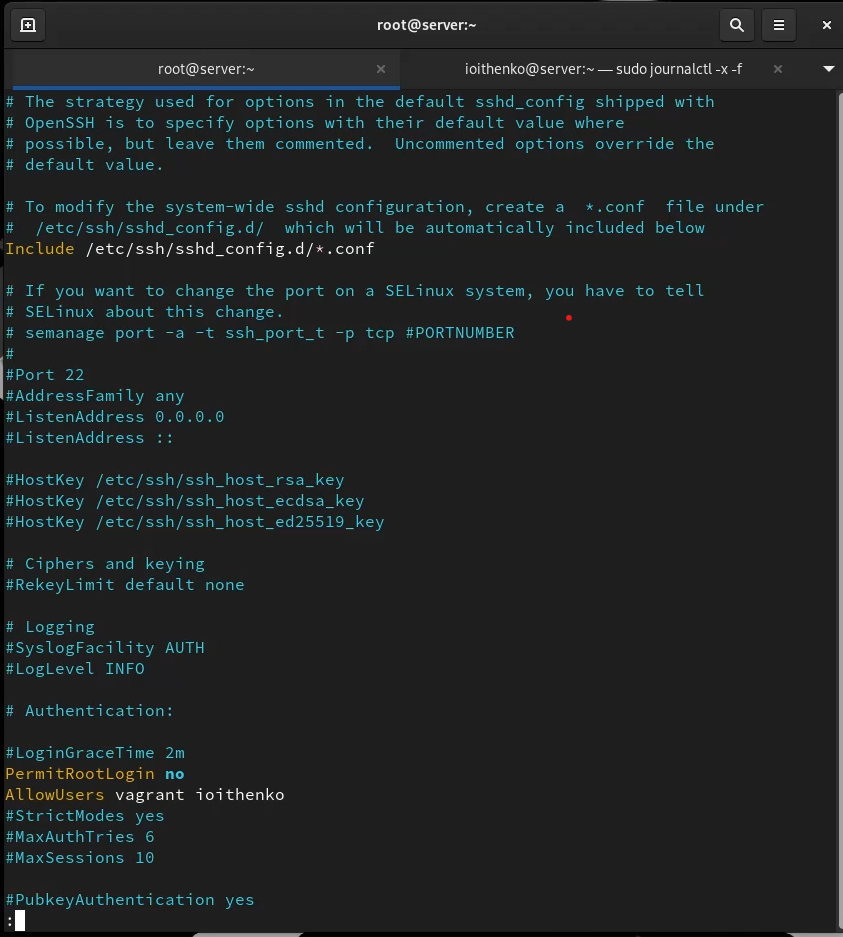


Рис. 8: Изменение разрешенных пользователей для sshd

Снова попытаемся установить соединение с клиента к серверу(fig. 9):

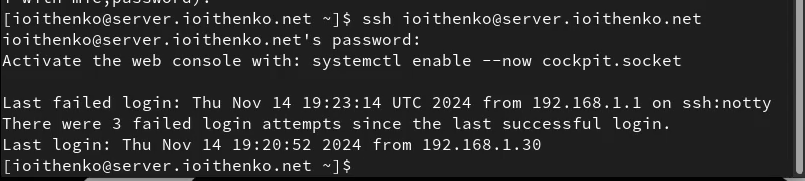


Рис. 9: Временный запуск SMTP-сервера

В этот раз доступ получен.

На сервере в файле конфигурации sshd /etc/ssh/sshd\_config найдем строку Port и ниже этой строки добавим(fig. 10):

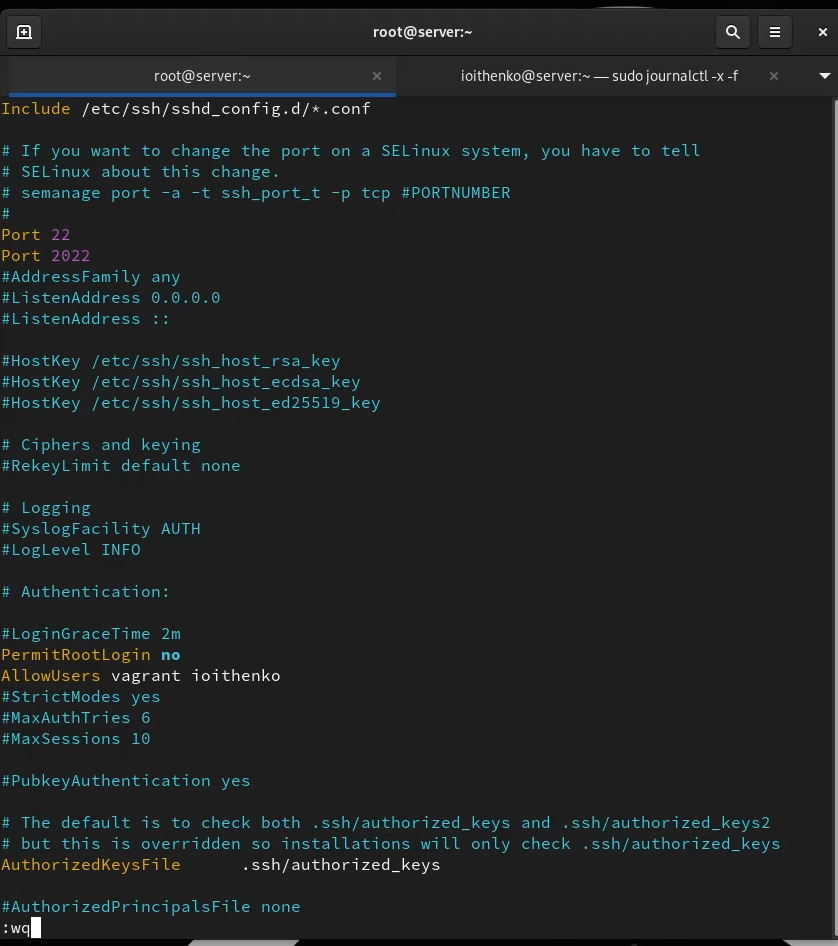


Рис. 10: Добавление портов в файл конфигураций

Эта запись сообщает процессу sshd о необходимости организации соединения через два разных порта, что даёт гарантию возможности открыть сеансы SSH, даже если была сделана ошибка в конфигурации.

После сохранения изменений в файле конфигурации перезапустим sshd.

Посмотрим расширенный статус работы sshd(fig. 11):

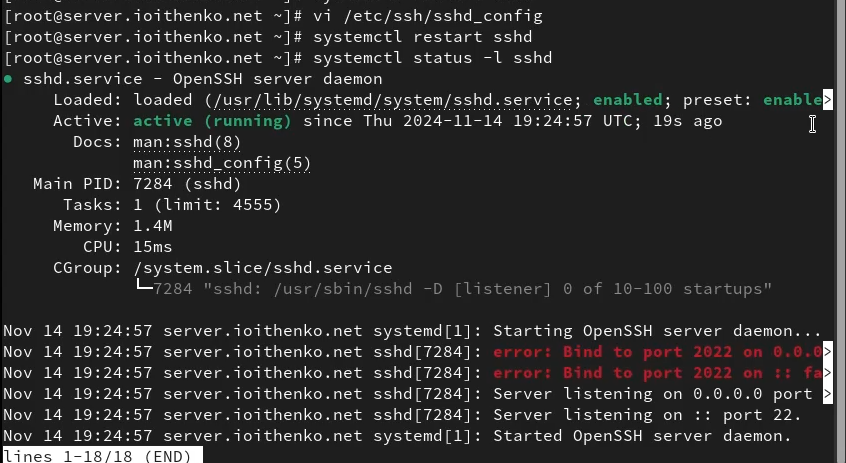


Рис. 11: Расширенный статус работы sshd

Система сообщает об отказе в работе sshd через порт 2022. Дополнительно посмотрим сообщения в терминале с мониторингом системных событий(рис. fig. 12):

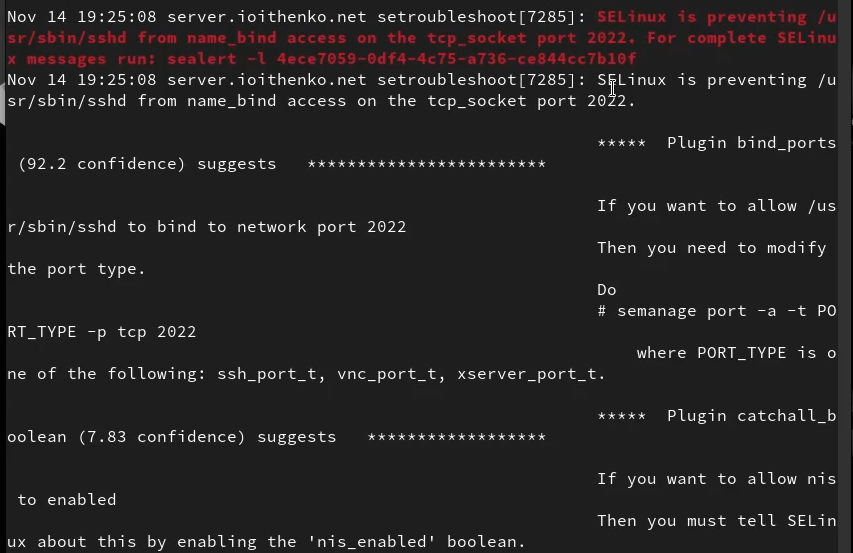


Рис. 12: Мониторинг системных сообщений

Можно увидеть, что отказ происходит из-за запрета SELinux на работу с этим портом.

Исправим на сервере метки SELinux к порту 2022 и в настройках межсетевого экрана откроем порт 2022 протокола. Вновь перезапустим sshd и посмотрите расширенный статус его работы. Статус показывает, что процесс sshd теперь прослушивает два порта(fig. 13)

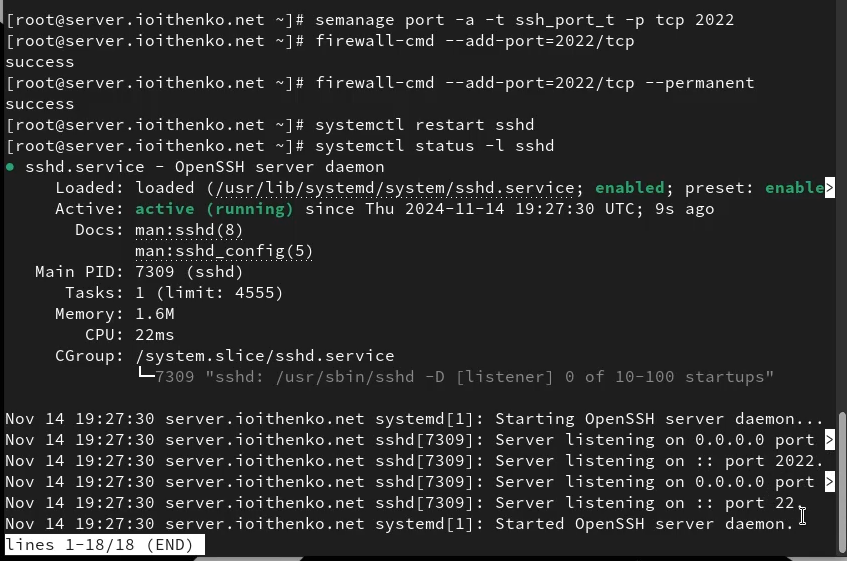


Рис. 13: Просмотр расширенного статуса работы sshd после настройки работы по порту 2022

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя ioithenko обычным способом и указав порт 2022(рис. fig. 14 и fig. 15):

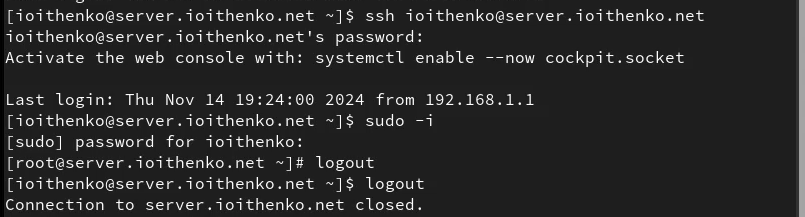


Рис. 14: Установка SSH-соединение с клиента



Рис. 15: Установка SSH-соединение с клиента с указанием порта 2022

Создадим пару из открытого и закрытого ключей для входа на сервер.

На сервере в конфигурационном файле /etc/ssh/sshd\_config зададим параметр, разрешающий аутентификацию по ключу (рис. fig. 16):

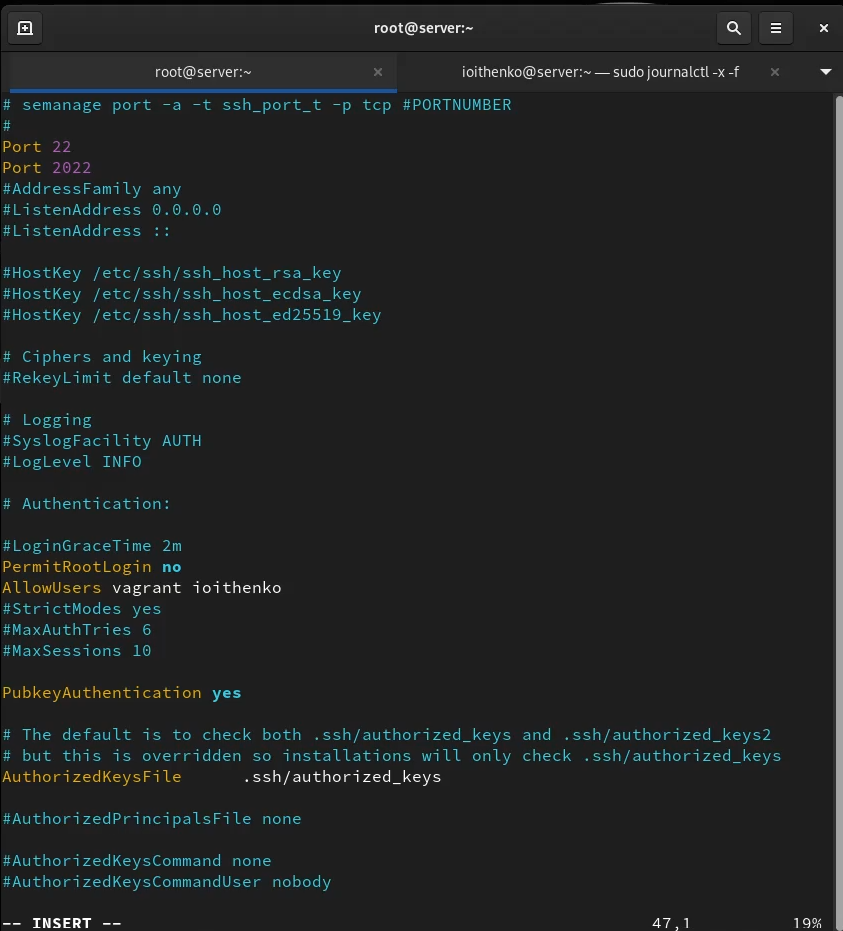


Рис. 16: Установка SSH-соединение с клиента

После сохранения изменений в файле конфигурации перезапустим sshd.

На клиенте сформируем SSH-ключ, введя в терминале.

Закрытый ключ теперь будет записан в файл ~/.ssh/id\_rsa, а открытый ключ записывается в файл ~/.ssh/id\_rsa.pub.

Скопируем открытый ключ на сервер.

Попробуем получить доступ с клиента к серверу посредством SSH-соединения(fig. 17):

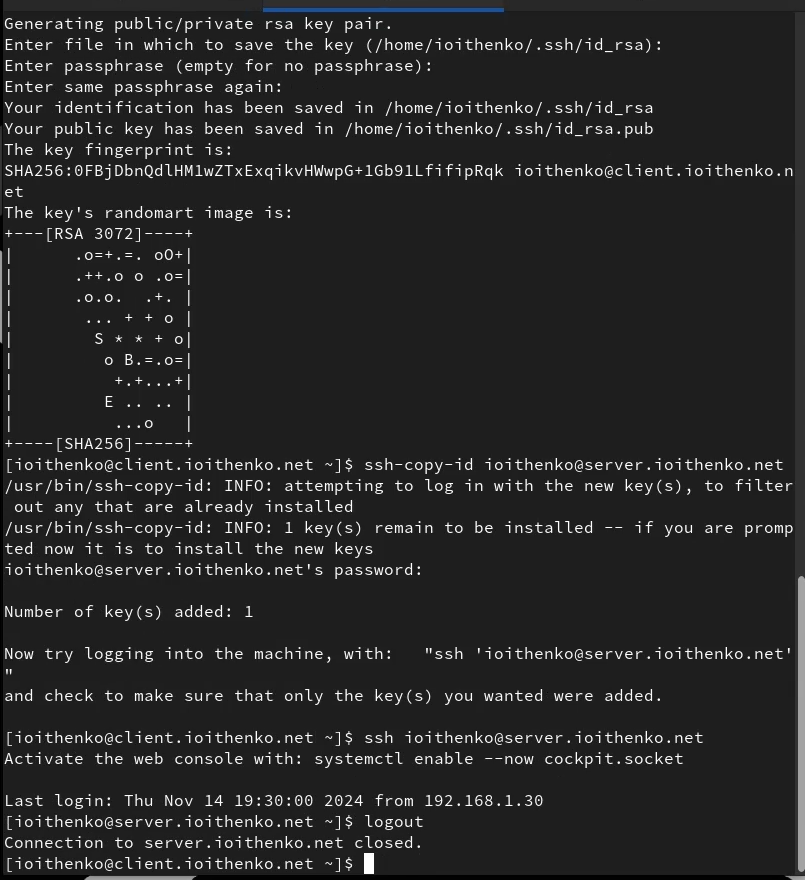


Рис. 17: Создание и копирование открытого ключа, установка SSH-соединения с сервером с клиента

На клиенте посмотрим, запущены ли какие-то службы с протоколом TCP, на данный момент их нет. Перенаправим порт 80 на server.ioithenko.net на порт 8080 на локальной машине и вновь на клиенте посмотрим, запущены ли какие-то службы с протоколом TCP(рис. fig. 18)

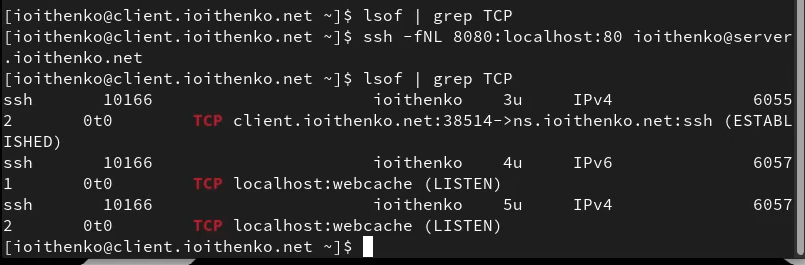


Рис. 18: Просмотр активных служб с протоколом TCP

Появилось три службы, использующие TCP протокол – появился доступ к mail.ioithenko.net по ssh, а также к локальному хосту по IPv4 и IPv6.

На клиенте запустим браузер и в адресной строке введем localhost:8080. Отображается страница с приветствием «Welcome to the server.ioithenko.net server»(fig. 19):



Рис. 19: Просмотр локального сервера в браузере на клиенте

На клиенте откроем терминал под пользователем ioithenko и посмотрим с клиента имя узла сервера, файлов на сервере и почту(рис. fig. 20 и fig. 21):

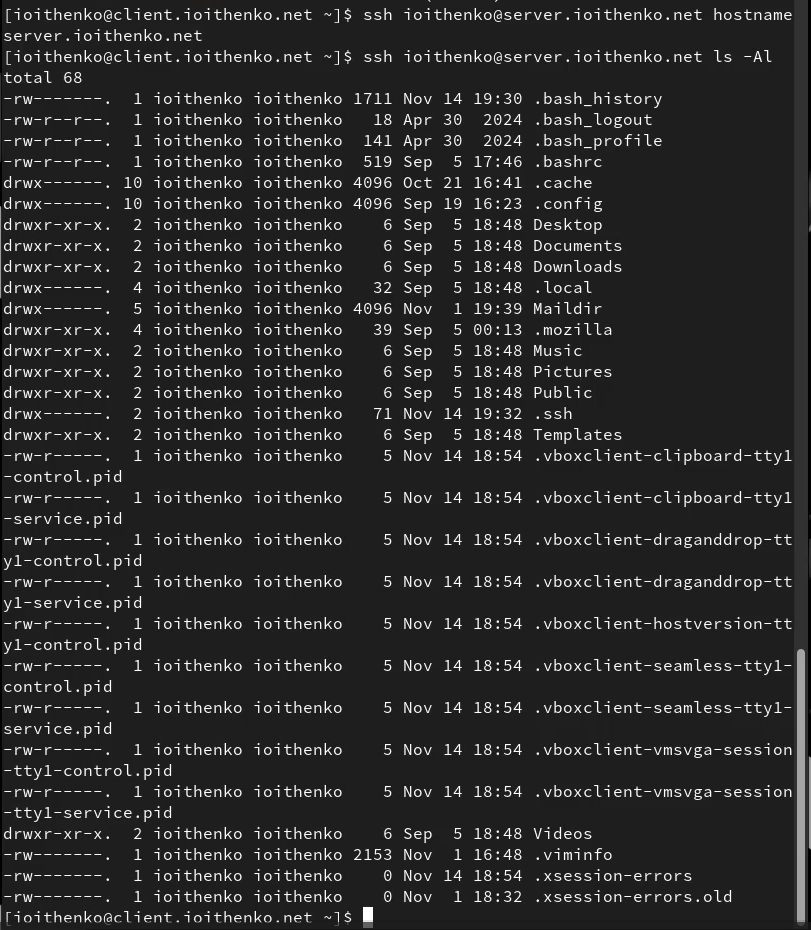


Рис. 20: Просмотр информации о сервере c клиента через ssh

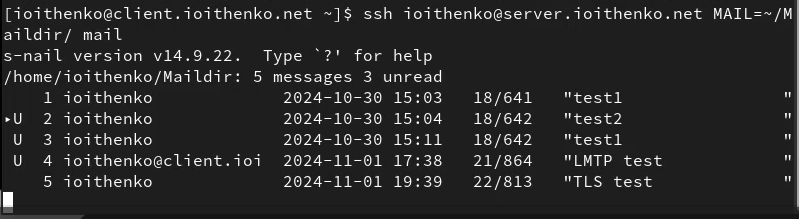


Рис. 21: Просмотр почты сервера c клиента через ssh

На сервере в конфигурационном файле /etc/ssh/sshd\_config разрешим отображать на локальном клиентском компьютере графические интерфейсы X11(рис. fig. 22):

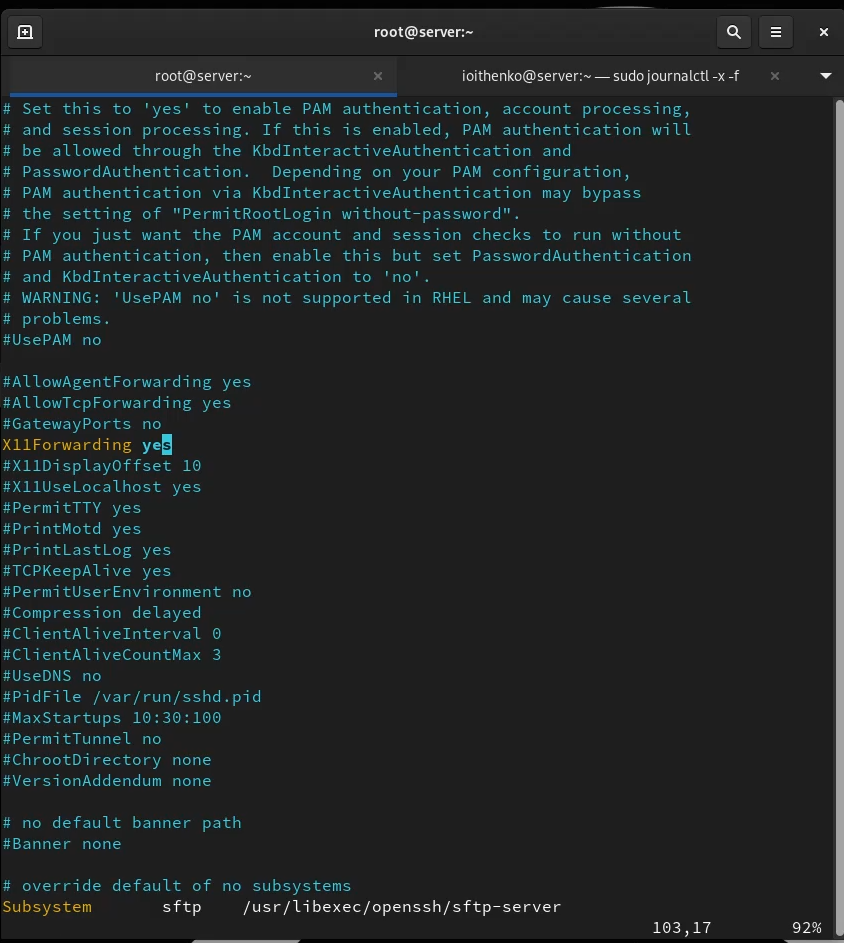


Рис. 22: Просмотр информации о сервере c клиента через ssh

После сохранения изменения в конфигурационном файле перезапустим sshd

Попробуем с клиента удалённо подключиться к серверу и запустить графическое приложение firefox(рис. fig. 23):

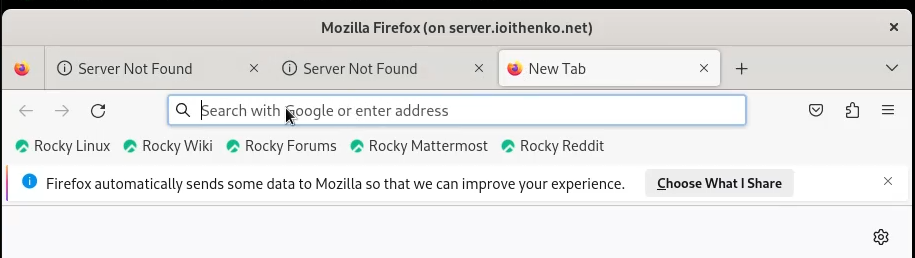


Рис. 23: Запуск графического приложения через ssh

На виртуальной машине server перейдем 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 и в каталоге /vagrant/provision/server создадим исполняемый файл ssh.sh.

Пропишем скрипт в /vagrant/provision/server/ssh.sh. Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile добавим следующую запись в разделе конфигурации для сервера.

# 3 Выводы

В ходе выполнения данной работы я приобрела практические навыки по настройке удалённого доступа к серверу с помощью SSH.

# 4 Контрольные вопросы

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

В файле /etc/ssh/sshd\_config конфигурации прописать PermitRootLogin no и AllowUsers alice.

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

Для настройки удалённого доступа по SSH через несколько портов нужно отредактировать файл конфигурации SSH и добавить строку Port <порт>.

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

Для установки фонового соединения без команды используется параметр -N при использовании команды ssh: ssh -N <hostname>

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

ssh -fNL 80:localhost:55555 server2.example.com

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

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

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

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