Лабораторная работа №11

Настройка безопасного удалённого доступа по протоколу SSH

Демидова Екатерина Алексеевна

Содержание

# 1 Цель работы

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

# 2 Задание

1. Настройте запрет удалённого доступа на сервер по SSH для пользователя root.
2. Настройте разрешение удалённого доступа к серверу по SSH только для пользователей группы vagrant и вашего пользователя.
3. Настройте удалённый доступ к серверу по SSH через порт 2022.
4. Настройте удалённый доступ к серверу по SSH по ключу.
5. Организуйте SSH-туннель с клиента на сервер, перенаправив локальное соединение с TCP-порта 80 на порт 8080.
6. Используя удалённое SSH-соединение, выполните с клиента несколько команд на сервере.
7. Используя удалённое SSH-соединение, запустите с клиента графическое приложение на сервере.
8. Напишите скрипт для Vagrant, фиксирующий действия по настройке SSH-сервера во внутреннем окружении виртуальной машины server. Соответствующим образом внесите изменения в Vagrantfile.

# 3 Выполнение лабораторной работы

## 3.1 Запрет удалённого доступа по SSH для пользователя root

Загрузим нашу операционную систему и перейдем в рабочий каталог с проектом:

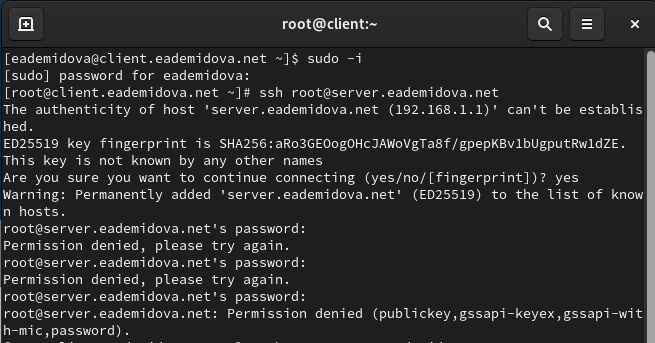
cd /var/tmp/eademidova/vagrant

Затем запустим виртуальную машину server:

make server-up

Откроем терминал и перейдем в режим суперпользователя

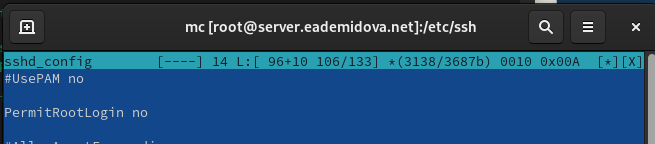
В дополнительном терминале запустим мониторинг системных событий с помощью команды journalctl -x -f. С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя root(рис. ??):



Попытка установить SSH-соединение

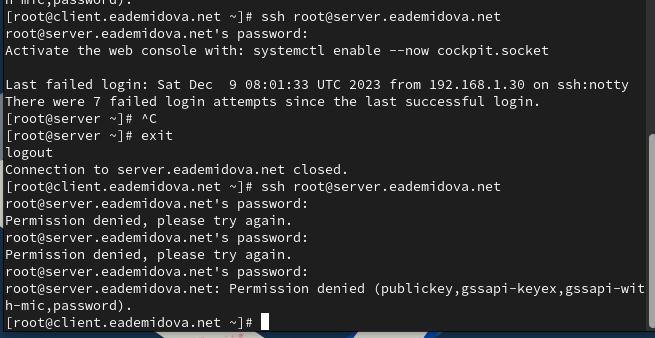
При попытке соединения, так как мы делаем это первый раз, добавлям сервер в список известных хостов. Затем требуется ввести пароль от пользователя root, но соединение отклоняется.

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



Запрет входа на сервер пользователю root

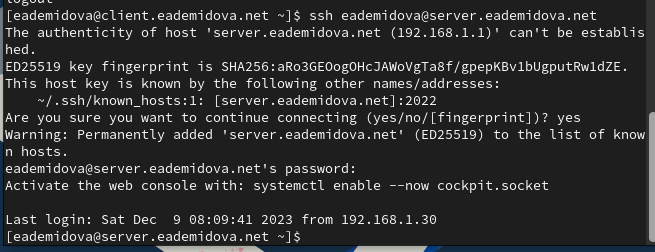
После сохранения изменений в файле конфигурации перезапустим sshd с помощью команды systemctl restart sshd. Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя root(здесь первое удачное соединение с разрешенным входом на сервер пользователю рут, а второ подключение уже с запрещенным)(??):



Повторная попытка SSH-соединение

## 3.2 Ограничение списка пользователей для удалённого доступа по SSH

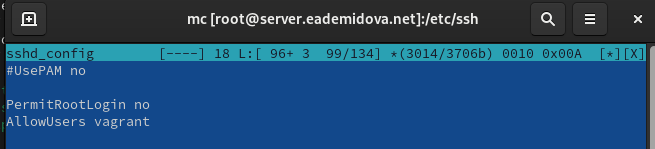
С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя eademidova(рис. ??):



Попытка установить SSH-соединение с клиента

Соединениие проходит удачно.

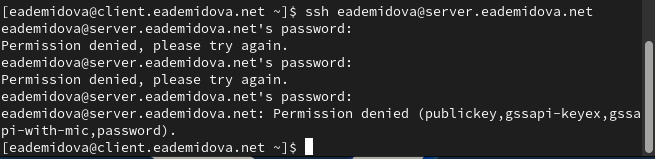
На сервере откроем файл /etc/ssh/sshd\_config конфигурации sshd на редактирование и добавим строку(??):



Изменение разрешенных пользователей для sshd

После сохранения изменений в файле конфигурации перезапустим sshd.

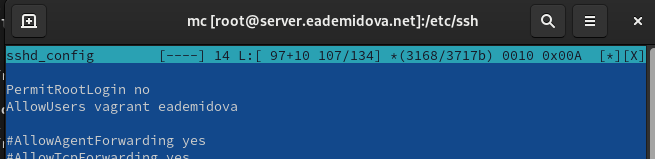
Повторим попытку получения доступа с клиента к серверу посредством SSH-соединения через пользователя eademidova(рис. ??):



Определение службы аутентификации пользователей

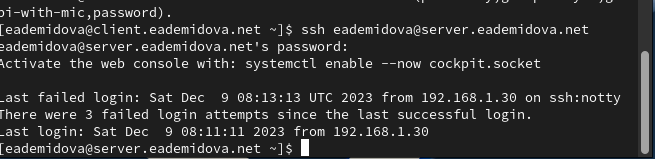
В этот раз соединение не устанавливается, так как в списке разрешенных пользователей нет нашего.

В файле /etc/ssh/sshd\_config конфигурации sshd внесем следующее изменение(??):



Изменение разрешенных пользователей для sshd

Снова попытаемся установить соединение с клиента к серверу(??):

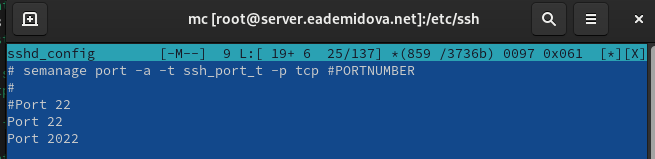


Временный запуск SMTP-сервера

В этот раз доступ получен.

## 3.3 Настройка дополнительных портов для удалённого доступа по SSH

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

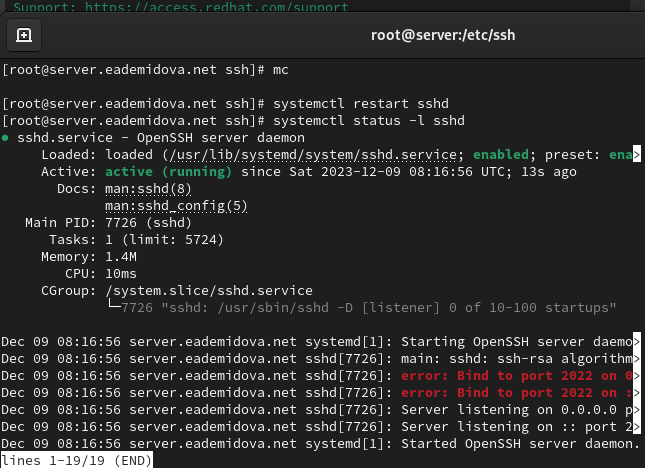


Добавление портов в файл конфигураций

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.

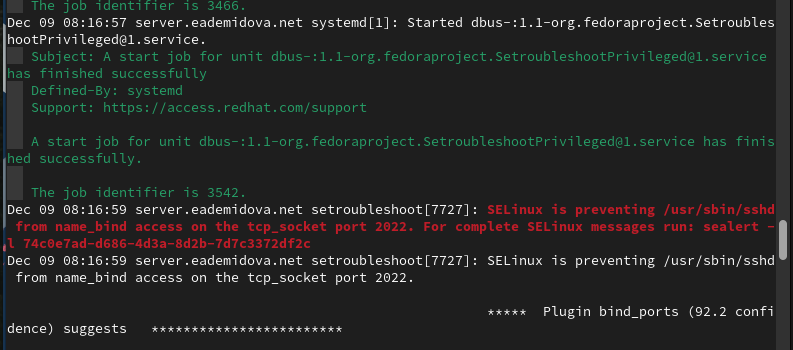
После сохранения изменений в файле конфигурации перезапустим sshd.

Посмотрим расширенный статус работы sshd(??):



Расширенный статус работы sshd

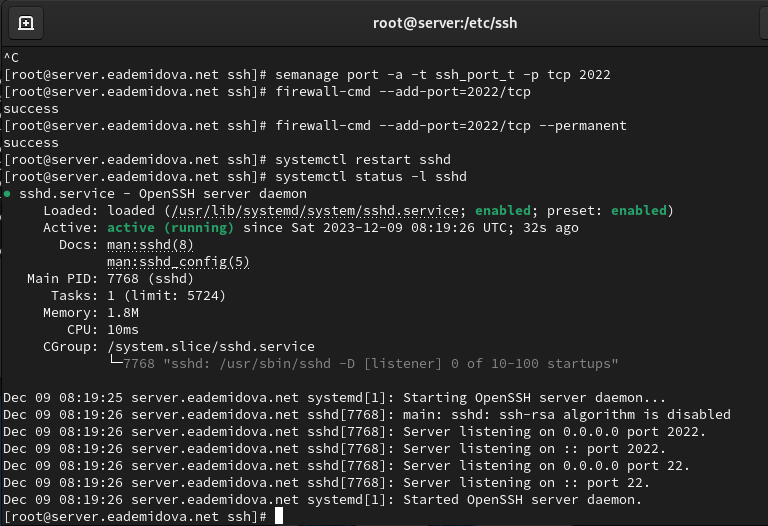
Система сообщает об отказе в работе sshd через порт 2022. Дополнительно посмотрим сообщения в терминале с мониторингом системных событий(рис. ??):



Мониторинг системных сообщений

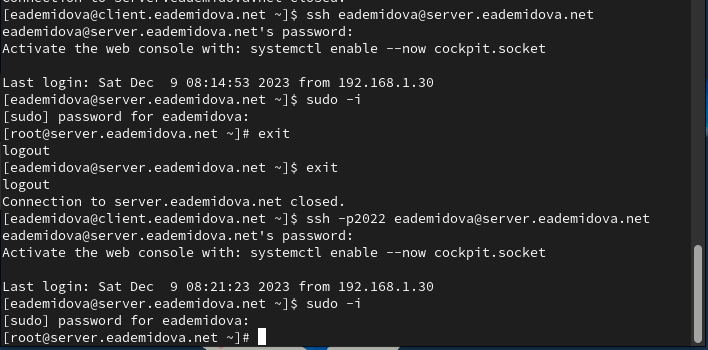
Можно увидеть, что отказ происходит из-за запрета SELinux на работу с этим портом.

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(??)



Просмотр расширенного статуса работы sshd после настройки работы по порту 2022

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



Установка SSH-соединение с клиента

## 3.4 Настройка удалённого доступа по SSH по ключу

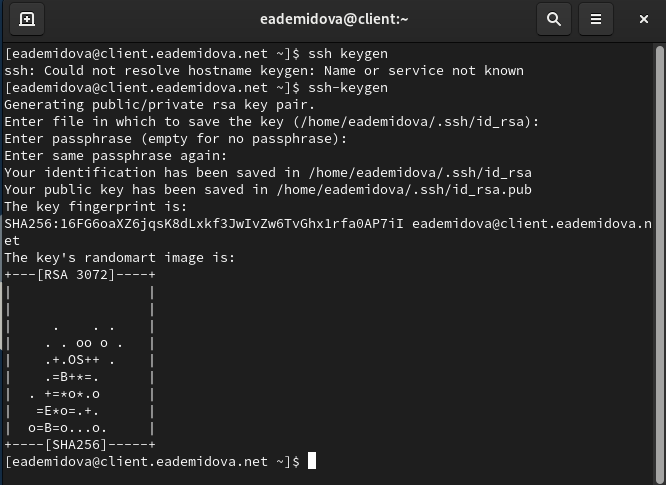
Создадим пару из открытого и закрытого ключей для входа на сервер.

На сервере в конфигурационном файле /etc/ssh/sshd\_config зададим параметр, разрешающий аутентификацию по ключу, написав:

PubkeyAuthentication yes

После сохранения изменений в файле конфигурации перезапустим sshd.

На клиенте сформируем SSH-ключ, введя в терминале(??):



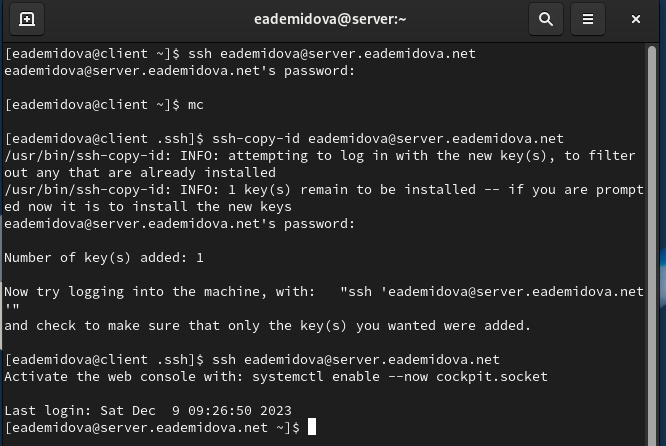
Формирования SSH-ключа на клиенте

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, введя на клиенте:

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

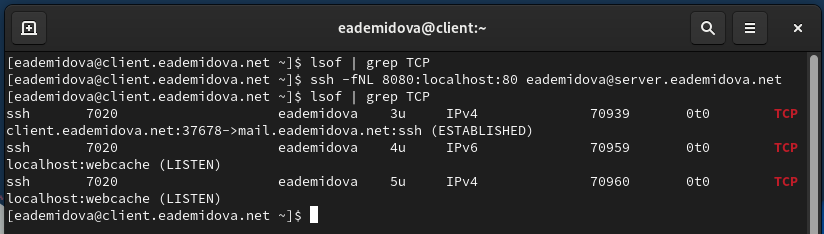
Попробуем получить доступ с клиента к серверу посредством SSH-соединения(??):



Установка SSH-соединения с сервером с клиента

## 3.5 Организация туннелей SSH, перенаправление TCP-портов

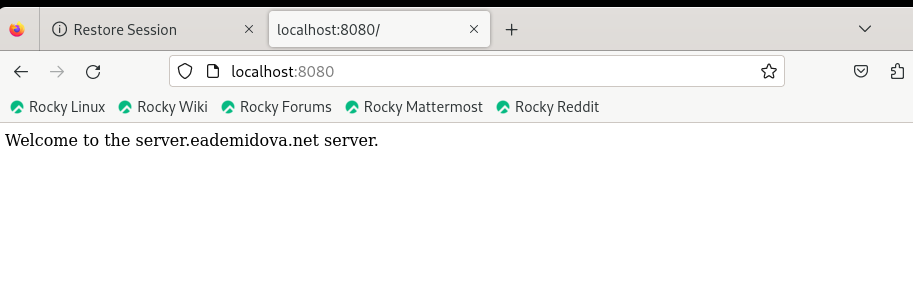
На клиенте посмотрим, запущены ли какие-то службы с протоколом TCP, на данный момент их нет. Перенаправим порт 80 на server.eademidova.net на порт 8080 на локальной машине и вновь на клиенте посмотрим, запущены ли какие-то службы с протоколом TCP(рис. ??)



Просмотр активных служб с протоколом TCP

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

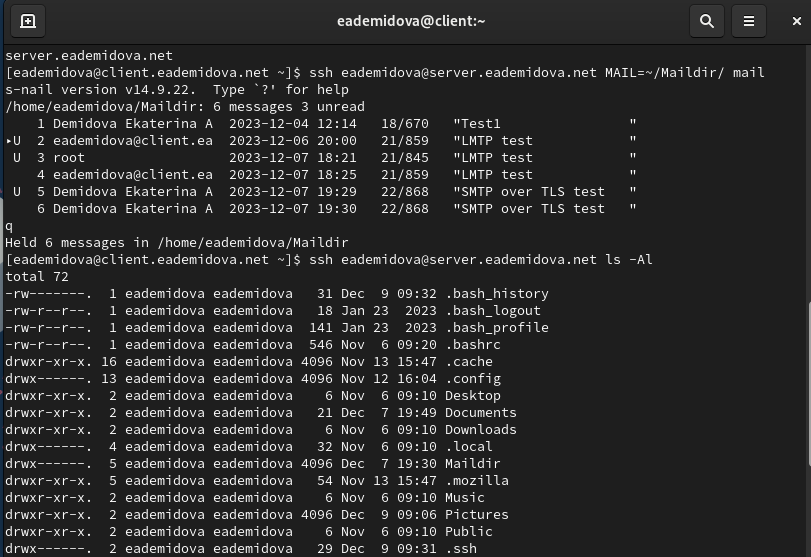
На клиенте запустим браузер и в адресной строке введем localhost:8080. Отображается страница с приветствием «Welcome to the server.eademidova.net server»(??):



Просмотр локального сервера в браузере на клиенте

## 3.6 Запуск консольных приложений через SSH

На клиенте откроем терминал под пользователем eademidova и посмотрим с клиента имя узла сервера, файлов на сервере и почту(рис. ??):



Просмотр информации о сервере c клиента через ssh

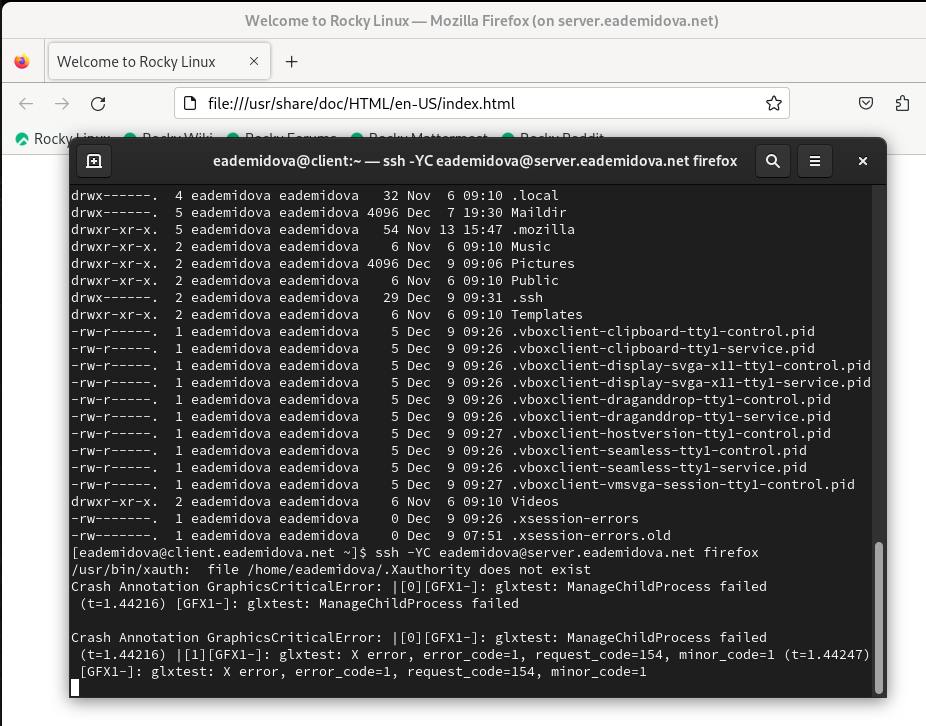
## 3.7 Запуск графических приложений через SSH (X11Forwarding)

На сервере в конфигурационном файле /etc/ssh/sshd\_config разрешим отображать на локальном клиентском компьютере графические интерфейсы X11, прописав:

X11Forwarding yes

После сохранения изменения в конфигурационном файле перезапустим sshd

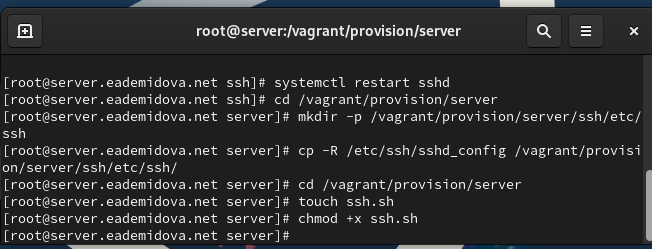
Попробуем с клиента удалённо подключиться к серверу и запустить графическое приложение firefox(рис. ??):



Запуск графического приложения через ssh

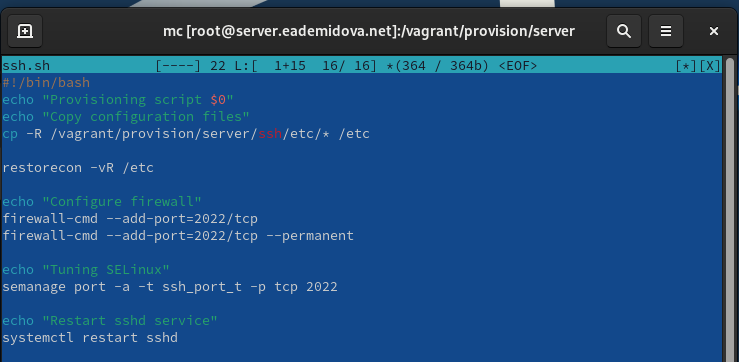
## 3.8 Внесение изменений в настройки внутреннего окружения виртуальной машины

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(рис. ??)



Создание окружения для внесения изменений в настройки окружающей среды

Пропишем скрипт в /vagrant/provision/server/ssh.sht(??):



Скрипта файла /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(??):



Изменние конфигурационного файла Vogrant

# 4 Контрольные вопросы

1. Вы хотите запретить удалённый доступ по SSH на сервер пользователю root и разрешить доступ пользователю alice. Как это сделать?
2. Как настроить удалённый доступ по SSH через несколько портов? Для чего это может потребоваться?
3. Какие параметры используются для создания туннеля SSH, когда команда ssh устанавливает фоновое соединение и не ожидает какой-либо конкретной команды?
4. Как настроить локальную переадресацию с локального порта 5555 на порт 80 сервера server2.example.com?
5. Как настроить SELinux, чтобы позволить SSH связываться с портом 2022?
6. Как настроить межсетевой экран на сервере, чтобы разрешить входящие подключения по SSH через порт 2022?
7. В файле /etc/ssh/sshd\_config конфигурации прописать PermitRootLogin no и AllowUsers alice.
8. Для настройки удалённого доступа по SSH через несколько портов нужно отредактировать файл конфигурации SSH (/etc/ssh/sshd\_config) и добавить строку Port <порт>.
9. Для установки фонового соединения без команды используется параметр -N при использовании команды ssh: ssh -N <hostname>.
10. Для настройки локальной переадресации с локального порта 5555 на порт 80 сервера server2.example.com следует использовать команду: ssh -fNL 80:localhost:55555 server2.example.com.
11. Для настройки SELinux и разрешения SSH связываться с портом 2022 можно использовать команду: semanage port -a -t ssh\_port\_t -p tcp 2022.
12. Для разрешения входящих подключений по SSH через порт 2022 с помощью межсетевого экрана следует использовать команду: firewall-cmd --add-port=2022/tcp --permanent.

# 5 Выводы

В результате выполнения данной работы были приобретены практические навыки по настройке удалённого доступа к серверу с помощью SSH.