

Отчёт по лабораторной работе №11

Администрирование сетевых подсистем

Ищенко Ирина НПИбд-02-22

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
3 Выводы	20
4 Контрольные вопросы	21

Список иллюстраций

2.1 Попытка установить SSH-соединение	6
2.2 Запрет входа на сервер пользователю root	7
2.3 Повторная попытка SSH-соединение	7
2.4 Попытка установить SSH-соединение с клиента	8
2.5 Мониторинг	8
2.6 Изменение разрешенных пользователей для sshd	9
2.7 Определение службы аутентификации пользователей	9
2.8 Изменение разрешенных пользователей для sshd	10
2.9 Временный запуск SMTP-сервера	10
2.10 Добавление портов в файл конфигураций	11
2.11 Расширенный статус работы sshd	12
2.12 Мониторинг системных сообщений	12
2.13 Просмотр расширенного статуса работы sshd после настройки работы по порту 2022	13
2.14 Установка SSH-соединение с клиента	13
2.15 Установка SSH-соединение с клиента с указанием порта 2022	13
2.16 Установка SSH-соединение с клиента	14
2.17 Создание и копирование открытого ключа, установка SSH-соединения с сервером с клиента	15
2.18 Просмотр активных служб с протоколом TCP	15
2.19 Просмотр локального сервера в браузере на клиенте	16
2.20 Просмотр информации о сервере с клиента через ssh	17
2.21 Просмотр почты сервера с клиента через ssh	17
2.22 Просмотр информации о сервере с клиента через ssh	18
2.23 Запуск графического приложения через ssh	18

Список таблиц

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. 2.1):

```
[ioithenko@client.ioithenko.net ~]$ ssh root@server.ioithenko.net
The authenticity of host 'server.ioithenko.net (192.168.1.1)' can't be established.
ED25519 key fingerprint is SHA256:Thye46E17svni0fMP0tToWzEFseVN0KvqNQRg3jfWCY.
This host key is known by the following other names/addresses:
  ~/ssh/known_hosts:1: [server.ioithenko.net]:22
  ~/ssh/known_hosts:4: server
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server.ioithenko.net' (ED25519) to the list of known hosts.
root@server.ioithenko.net's password:
Permission denied, please try again.
root@server.ioithenko.net's password:
Permission denied, please try again.
root@server.ioithenko.net's password:
root@server.ioithenko.net: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
```

Рис. 2.1: Попытка установить SSH-соединение

При попытке соединения, так как мы делаем это первый раз, добавляем 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. 2.2):

```

root@server:~          ioithenko@server:~ -- sudo journalctl -x -f
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO
|
# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work, you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
-- INSERT --

```

Рис. 2.2: 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. 2.3):

```

[ioithenko@client.ioithenko.net ~]$ ssh root@server
The authenticity of host 'server (192.168.1.1)' can't be established.
ED25519 key fingerprint is SHA256:Thye46E17svni0fMP0tToWzEFseVN0KvqNQRg3jfWCY.
This host key is known by the following other names/addresses:
  ~/ssh/known_hosts:1: [server.ioithenko.net]:22
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server' (ED25519) to the list of known hosts.
root@server's password:
Permission denied, please try again.
root@server's password:
Permission denied, please try again.
root@server's password:
root@server: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).

```

Рис. 2.3: Повторная попытка SSH-соединение

С клиента попытаемся получить доступ к серверу посредством SSH-соединения через пользователя ioithenko(рис. fig. 2.4 и fig. 2.5):

```
[ioithenko@client.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net  
ioithenko@server.ioithenko.net's password:  
Activate the web console with: systemctl enable --now cockpit.socket  
  
Last login: Thu Nov 14 18:54:41 2024  
[ioithenko@server.ioithenko.net ~]$ █
```

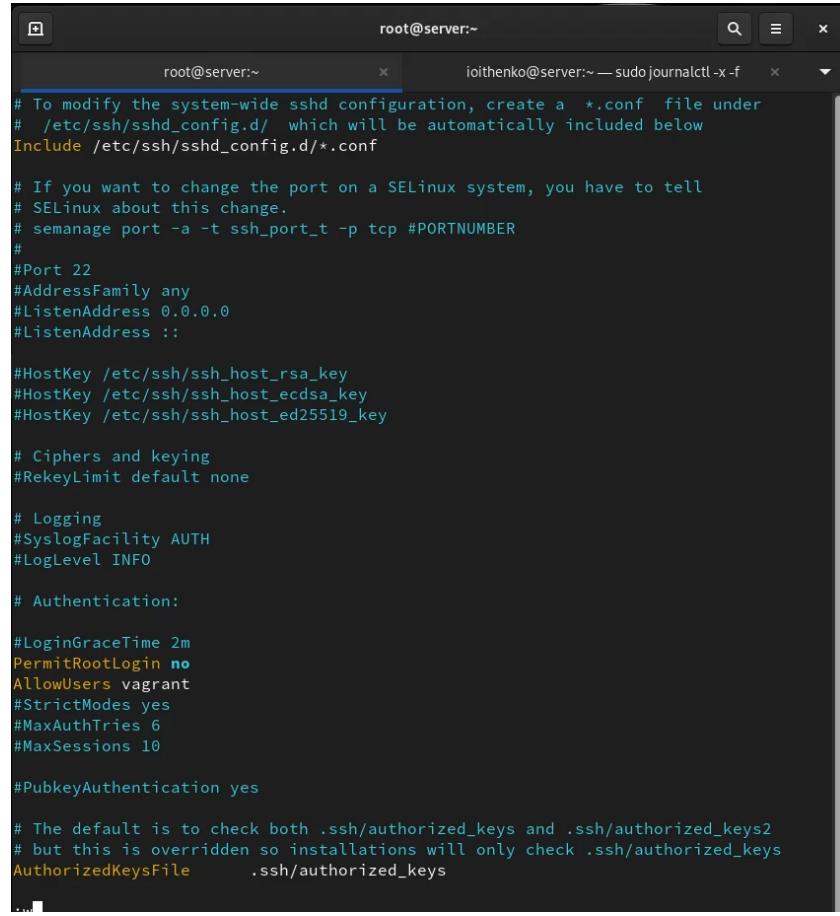
Рис. 2.4: Попытка установить SSH-соединение с клиента

```
Nov 14 19:20:52 server.ioithenko.net sshd[7151]: pam_unix(sshd:session): session opened for user ioithenko(uid=1001) by ioithenko(uid=0)  
Nov 14 19:20:52 server.ioithenko.net systemd[1]: Starting Hostname Service...  
  Subject: A start job for unit systemd-hostnamed.service has begun execution  
  Defined-By: systemd  
  Support: https://wiki.rockylinux.org/rocky/support  
  
    A start job for unit systemd-hostnamed.service has begun execution.  
  
    The job identifier is 2827.  
Nov 14 19:20:52 server.ioithenko.net systemd[1]: Started Hostname Service.  
  Subject: A start job for unit systemd-hostnamed.service has finished successfully  
  Defined-By: systemd  
  Support: https://wiki.rockylinux.org/rocky/support  
  
    A start job for unit systemd-hostnamed.service has finished successfully.  
  
    The job identifier is 2827.
```

Рис. 2.5: Мониторинг

Соединение проходит удачно.

На сервере откроем файл /etc/ssh/sshd_config конфигурации sshd на редактирование и добавим строку(fig. 2.6):



The screenshot shows a terminal window with two tabs. The left tab is titled 'root@server:~' and contains the configuration file for the sshd service. The right tab is titled 'ioithenko@server:~ — sudo journalctl -x -f' and shows the system logs.

```
# To modify the system-wide sshd configuration, create a *.conf file under
# /etc/ssh/sshd_config.d/ which will be automatically included below
Include /etc/ssh/sshd_config.d/*.conf

# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
AllowUsers vagrant
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

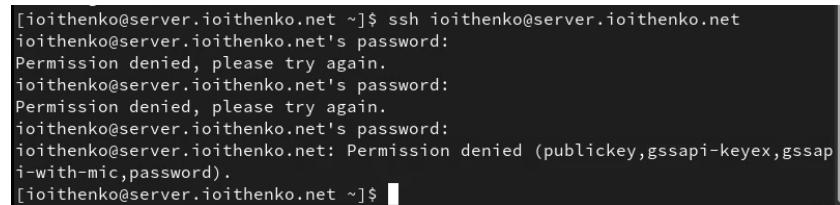
#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys
```

Рис. 2.6: 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. 2.7):



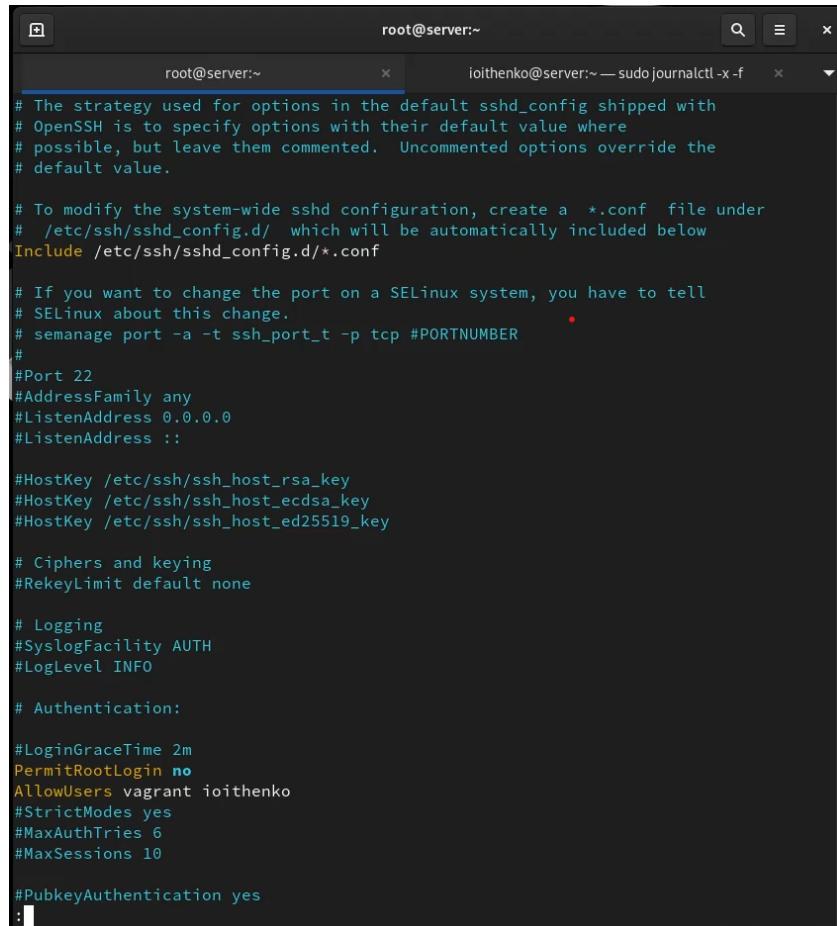
The screenshot shows a terminal window with a single command being run:

```
[ioithenko@server.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net
ioithenko@server.ioithenko.net's password:
Permission denied, please try again.
ioithenko@server.ioithenko.net's password:
Permission denied, please try again.
ioithenko@server.ioithenko.net's password:
ioithenko@server.ioithenko.net: Permission denied (publickey,gssapi-keyex,gssapi
i-with-mic,password).
[ioithenko@server.ioithenko.net ~]$
```

Рис. 2.7: 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. 2.8):



```
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

# To modify the system-wide sshd configuration, create a *.conf file under
# /etc/ssh/sshd_config.d/ which will be automatically included below
Include /etc/ssh/sshd_config.d/*.conf

# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

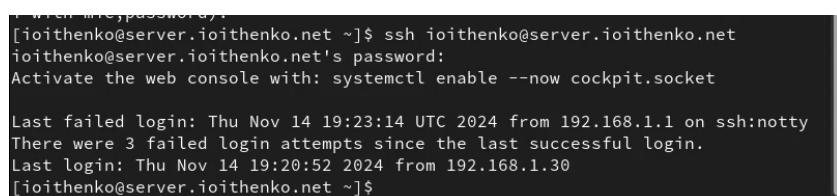
# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
AllowUsers vagrant ioithenko
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
:|
```

Рис. 2.8: Изменение разрешенных пользователей для sshd

Снова попытаемся установить соединение с клиентом к серверу(fig. 2.9):



```
[root@vagrant ~]$ ssh ioithenko@server.ioithenko.net
[ioithenko@server.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net
ioithenko@server.ioithenko.net's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last failed login: Thu Nov 14 19:23:14 UTC 2024 from 192.168.1.1 on ssh:notty
There were 3 failed login attempts since the last successful login.
Last login: Thu Nov 14 19:20:52 2024 from 192.168.1.30
[ioithenko@server.ioithenko.net ~]$
```

Рис. 2.9: 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. 2.10):

```
# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
Port 22
Port 2022
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
AllowUsers vagrant ioithenko
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none
:wq
```

Рис. 2.10: 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. 2.11):

```
[root@server.ioithenko.net ~]# vi /etc/ssh/sshd_config
[root@server.ioithenko.net ~]# systemctl restart sshd
[root@server.ioithenko.net ~]# systemctl status -l sshd
● sshd.service - OpenSSH server daemon
    Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
      Active: active (running) since Thu 2024-11-14 19:24:57 UTC; 19s ago
        Docs: man:sshd(8)
               man:sshd_config(5)
       Main PID: 7284 (sshd)
          Tasks: 1 (limit: 4555)
         Memory: 1.4M
            CPU: 15ms
           CGroup: /system.slice/sshd.service
                   └─7284 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Nov 14 19:24:57 server.ioithenko.net systemd[1]: Starting OpenSSH server daemon...
Nov 14 19:24:57 server.ioithenko.net sshd[7284]: error: Bind to port 2022 on 0.0.0.0 failed.
Nov 14 19:24:57 server.ioithenko.net sshd[7284]: error: Bind to port 2022 on :: failed.
Nov 14 19:24:57 server.ioithenko.net sshd[7284]: Server listening on 0.0.0.0 port 22.
Nov 14 19:24:57 server.ioithenko.net sshd[7284]: Server listening on :: port 22.
Nov 14 19:24:57 server.ioithenko.net systemd[1]: Started OpenSSH server daemon.

[lines 1-18/18 (END)]
```

Рис. 2.11: 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. 2.12):

```
Nov 14 19:25:08 server.ioithenko.net setroubleshoot[7285]: SELinux is preventing /usr/sbin/sshd from name_bind access on the tcp_socket port 2022. For complete SELinux messages run: sealert -l 4ece7059-0df4-4c75-a736-ce844cc7b10f
Nov 14 19:25:08 server.ioithenko.net setroubleshoot[7285]: SELinux is preventing /usr/sbin/sshd from name_bind access on the tcp_socket port 2022.

***** Plugin bind_ports (92.2 confidence) suggests *****

If you want to allow /usr/sbin/sshd to bind to network port 2022
Then you need to modify
the port type.
Do
# semanage port -a -t PORT_TYPE
where PORT_TYPE is one of the following: ssh_port_t, vnc_port_t, xserver_port_t.

***** Plugin catchall_boolean (7.83 confidence) suggests *****

If you want to allow nis to enable
Then you must tell SELinux about this by enabling the 'nis_enabled' boolean.
```

Рис. 2.12: 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. 2.13)

```
[root@server.ioithenko.net ~]# semanage port -a -t ssh_port_t -p tcp 2022
[root@server.ioithenko.net ~]# firewall-cmd --add-port=2022/tcp
success
[root@server.ioithenko.net ~]# firewall-cmd --add-port=2022/tcp --permanent
success
[root@server.ioithenko.net ~]# systemctl restart sshd
[root@server.ioithenko.net ~]# systemctl status -l sshd
● sshd.service - OpenSSH server daemon
    Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
      Active: active (running) since Thu 2024-11-14 19:27:30 UTC; 9s ago
        Docs: man:sshd(8)
               man:sshd_config(5)
       Main PID: 7309 (sshd)
          Tasks: 1 (limit: 4555)
         Memory: 1.6M
            CPU: 22ms
           CGroup: /system.slice/sshd.service
                   └─7309 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Nov 14 19:27:30 server.ioithenko.net systemd[1]: Starting OpenSSH server daemon...
Nov 14 19:27:30 server.ioithenko.net sshd[7309]: Server listening on 0.0.0.0 port >
Nov 14 19:27:30 server.ioithenko.net sshd[7309]: Server listening on :: port 2022.
Nov 14 19:27:30 server.ioithenko.net sshd[7309]: Server listening on 0.0.0.0 port >
Nov 14 19:27:30 server.ioithenko.net sshd[7309]: Server listening on :: port 22.
Nov 14 19:27:30 server.ioithenko.net systemd[1]: Started OpenSSH server daemon.
lines 1-18/18 (END)
```

Рис. 2.13: 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. 2.14 и fig. 2.15):

```
[ioithenko@server.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net
ioithenko@server.ioithenko.net's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Nov 14 19:24:00 2024 from 192.168.1.1
[ioithenko@server.ioithenko.net ~]$ sudo -i
[sudo] password for ioithenko:
[root@server.ioithenko.net ~]# logout
[ioithenko@server.ioithenko.net ~]$ logout
Connection to server.ioithenko.net closed.
```

Рис. 2.14: Установка SSH-соединение с клиента

```
[ioithenko@client.ioithenko.net ~]$ ssh -p2022 ioithenko@server.ioithenko.net
ioithenko@server.ioithenko.net's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Nov 14 19:28:28 2024 from 192.168.1.1
[ioithenko@server.ioithenko.net ~]$ sudo -i
[sudo] password for ioithenko:
[root@server.ioithenko.net ~]# logout
[ioithenko@server.ioithenko.net ~]$ logout
Connection to server.ioithenko.net closed.
[ioithenko@client.ioithenko.net ~]$
```

Рис. 2.15: 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. 2.16):

The screenshot shows a terminal window titled 'root@server:~' with two tabs open. The left tab shows the command: '# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER'. The right tab shows the output of the command: 'ioithenka@server:~ — sudo journalctl -x -f'. The configuration file content is as follows:

```
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
#Port 2022
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
AllowUsers vagrant ioithenko
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

-- INSERT --
```

Рис. 2.16: Установка SSH-соединение с клиентом

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. 2.17):

```

Generating public/private rsa key pair.
Enter file in which to save the key (/home/ioithenko/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ioithenko/.ssh/id_rsa
Your public key has been saved in /home/ioithenko/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:0FBjDbnQdLHM1wZTxExqikvHWwpG+1Gb91LfifipRqk ioithenko@client.ioithenko.net
The key's randomart image is:
+---[RSA 3072]---+
| .o+=.=. o0+ |
| .++.o o .o= |
| .o.o. .+. |
| ... + + o |
| S * * + o |
| o B.=o= |
| +.+....+ |
| E ... . |
| ...o |
+---[SHA256]---+
[ioithenko@client.ioithenko.net ~]$ ssh-copy-id ioithenko@server.ioithenko.net
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
ioithenko@server.ioithenko.net's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'ioithenko@server.ioithenko.net'"
"
and check to make sure that only the key(s) you wanted were added.

[ioithenko@client.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Nov 14 19:30:00 2024 from 192.168.1.30
[ioithenko@server.ioithenko.net ~]$ logout
Connection to server.ioithenko.net closed.
[ioithenko@client.ioithenko.net ~]$ 
```

Рис. 2.17: 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. 2.18)

```

[ioithenko@client.ioithenko.net ~]$ lsof | grep TCP
[ioithenko@client.ioithenko.net ~]$ ssh -fNL 8080:localhost:80 ioithenko@server.ioithenko.net
[ioithenko@client.ioithenko.net ~]$ lsof | grep TCP
ssh      10166          ioithenko    3u      IPv4          6055
2      0t0      TCP  client.ioithenko.net:38514->ns.ioithenko.net:ssh (ESTABLISHED)
ssh      10166          ioithenko    4u      IPv6          6057
1      0t0      TCP  localhost:webcache (LISTEN)
ssh      10166          ioithenko    5u      IPv4          6057
2      0t0      TCP  localhost:webcache (LISTEN)
[ioithenko@client.ioithenko.net ~]$ 
```

Рис. 2.18: Просмотр активных служб с протоколом TCP

Появились 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. 2.19):

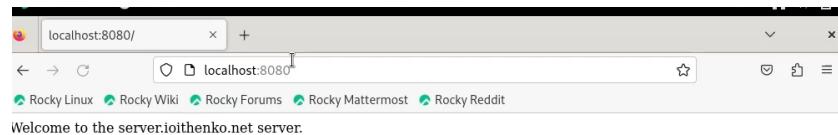


Рис. 2.19: Просмотр локального сервера в браузере на клиенте

На клиенте откроем терминал под пользователем ioithenko и посмотрим с клиента имя узла сервера, файлов на сервере и почту(рис. fig. 2.20 и fig. 2.21):

```
[ioithenko@client.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net hostname
server.ioithenko.net
[ioithenko@client.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net ls -Al
total 68
-rw-----. 1 ioithenko ioithenko 1711 Nov 14 19:30 .bash_history
-rw-r--r--. 1 ioithenko ioithenko 18 Apr 30 2024 .bash_logout
-rw-r--r--. 1 ioithenko ioithenko 141 Apr 30 2024 .bash_profile
-rw-r--r--. 1 ioithenko ioithenko 519 Sep 5 17:46 .bashrc
drwx-----. 10 ioithenko ioithenko 4096 Oct 21 16:41 .cache
drwx-----. 10 ioithenko ioithenko 4096 Sep 19 16:23 .config
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Desktop
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Documents
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Downloads
drwx-----. 4 ioithenko ioithenko 32 Sep 5 18:48 .local
drwx-----. 5 ioithenko ioithenko 4096 Nov 1 19:39 Maildir
drwxr-xr-x. 4 ioithenko ioithenko 39 Sep 5 00:13 .mozilla
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Music
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Pictures
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Public
drwx-----. 2 ioithenko ioithenko 71 Nov 14 19:32 .ssh
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Templates
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-clipboard-tty1
-control.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-clipboard-tty1
-service.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-draganddrop-tt
y1-control.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-draganddrop-tt
y1-service.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-hostversion-tt
y1-control.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-seamless-tty1-
control.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-seamless-tty1-
service.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-vmsvga-session
-tty1-control.pid
-rw-----. 1 ioithenko ioithenko 5 Nov 14 18:54 .vboxclient-vmsvga-session
-tty1-service.pid
drwxr-xr-x. 2 ioithenko ioithenko 6 Sep 5 18:48 Videos
-rw-----. 1 ioithenko ioithenko 2153 Nov 1 16:48 .viminfo
-rw-----. 1 ioithenko ioithenko 0 Nov 14 18:54 .xsession-errors
-rw-----. 1 ioithenko ioithenko 0 Nov 1 18:32 .xsession-errors.old
[ioithenko@client.ioithenko.net ~]$
```

Рис. 2.20: Просмотр информации о сервере с клиента через ssh

```
[ioithenko@client.ioithenko.net ~]$ ssh ioithenko@server.ioithenko.net MAIL=~/M
aildir/mail
s-nail version v14.9.22. Type `?' for help
/home/ioithenko/Maildir: 5 messages 3 unread
  1 ioithenko      2024-10-30 15:03  18/641 "test1"
>U 2 ioithenko    2024-10-30 15:04  18/642 "test2"
U 3 ioithenko    2024-10-30 15:11  18/642 "test1"
U 4 ioithenko@cl 2024-11-01 17:38  21/864 "LMTP test"
  5 ioithenko      2024-11-01 19:39  22/813 "TLS test"
```

Рис. 2.21: Просмотр почты сервера с клиента через ssh

На сервере в конфигурационном файле /etc/ssh/sshd_config разрешим отображать на локальном клиентском компьютере графические интерфейсы X11(рис. fig. 2.22):

```

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the KbdInteractiveAuthentication and
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication via KbdInteractiveAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and KbdInteractiveAuthentication to 'no'.
# WARNING: 'UsePAM no' is not supported in RHEL and may cause several
# problems.
#UsePAM no

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
#PrintMotd yes
#PrintLastLog yes
#TCPKeepAlive yes
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# override default of no subsystems
Subsystem      sftp      /usr/libexec/openssh/sftp-server

```

103,17 92%

Рис. 2.22: Просмотр информации о сервере с 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. 2.23):

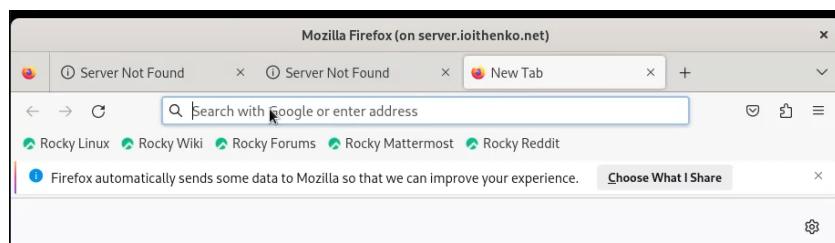


Рис. 2.23: Запуск графического приложения через ssh

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.

2. 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>.

3. 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>

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

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

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

```
semanage port -a -t ssh_port_t -p tcp 2022
```

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

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