Отчёт о лабораторной работе

Лабораторная работа 11

Мошаров Денис Максимович

Содержание

Цель работы

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

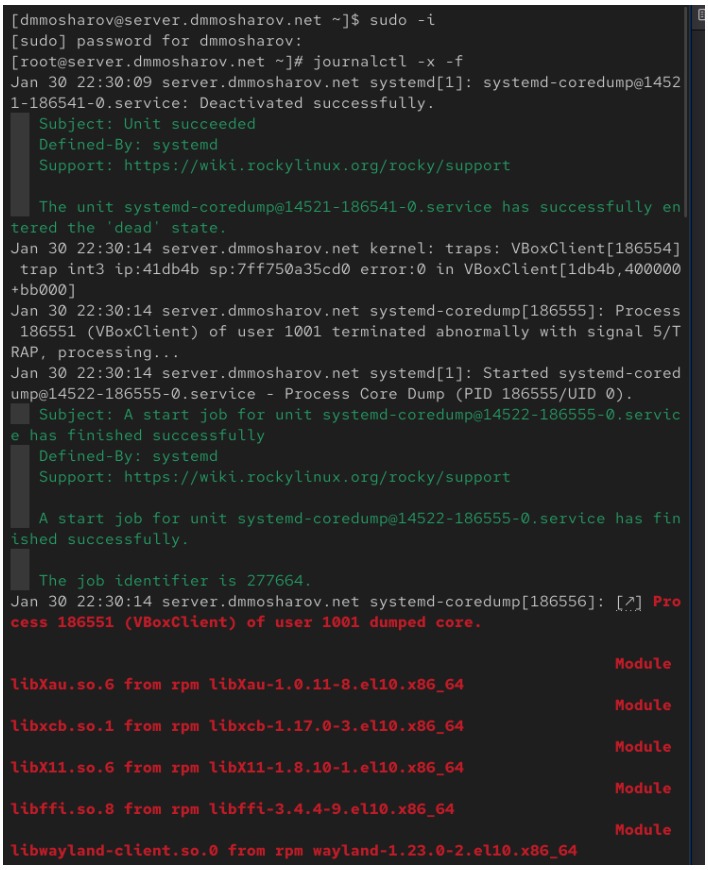
Выполнение лабораторной работы

Для начала зайдём на сервер и авторизуемся от рута, после чего сменим для него пароль (рис. [-@fig:001]).

```
[dmmosharov@server.dmmosharov.net ~]$ sudo -i
[sudo] password for dmmosharov:
[root@server.dmmosharov.net ~]# passwd root
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
[root@server.dmmosharov.net ~]#
```

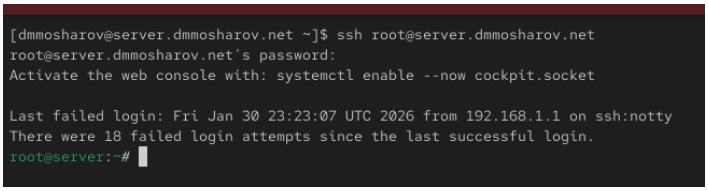
Смена пароля для root

Во втором терминале запустим вывод логов journalctl в реальном времени (рис. [-@fig:002]).



journalctl

Далее, зайдём на клиент. С клиента попытаемся по ssh подключиться к серверу от имени учётной записи root. Операция успешна (рис. [-@fig:003]).



Авторизация под root по ssh

Посмотрим логи и увидим, что авторизация была разрешена, и парол был принят (рис. [-@fig:004]).

```
  should be reported to its vendor as a bug.
Jan 30 23:26:53 server.dmmosharov.net systemd[1]: systemd-coredump@1516
8-195032-0.service: Deactivated successfully.
  Subject: Unit succeeded
  Defined-By: systemd
  Support: https://wiki.rockylinux.org/rocky/support

  The unit systemd-coredump@15168-195032-0.service has successfully en
tered the 'dead' state.
Jan 30 23:26:58 server.dmmosharov.net kernel: traps: VBoxClient[195041]
 trap int3 ip:41db4b sp:7ff750a35cd0 error:0 in VBoxClient[1db4b,400000
+bb000]
Jan 30 23:26:58 server.dmmosharov.net systemd-coredump[195042]: Process
 195038 (VBoxClient) of user 1001 terminated abnormally with signal 5/T
RAP, processing...
Jan 30 23:26:58 server.dmmosharov.net systemd[1]: Started systemd-cored
ump@15169-195042-0.service - Process Core Dump (PID 195042/UID 0).
  Subject: A start job for unit systemd-coredump@15169-195042-0.servic
e has finished successfully
  Defined-By: systemd
  Support: https://wiki.rockylinux.org/rocky/support

  A start job for unit systemd-coredump@15169-195042-0.service has fin
```

Лог авторизации

Теперь откроем на сервере файл /etc/ssh/sshd_config и пропишем там PermitRootLogin как no, для запрета авторизации от имени рута (рис. [-@fig:005]).

```
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
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

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

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody
                        [ Wrote 132 lines ]
^G Help       ^O Write Out  ^F Where Is   ^K Cut        ^T Execute
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

Редактирование /etc/ssh/sshd_config

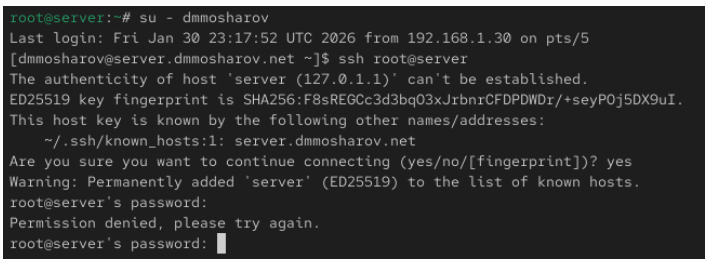
Перезапустим теперь службу sshd на сервере (рис. [-@fig:006]).

```
[root@server.dmmosharov.net ~]# systemctl restart sshd
[root@server.dmmosharov.net ~]#
```

Перезапуск sshd

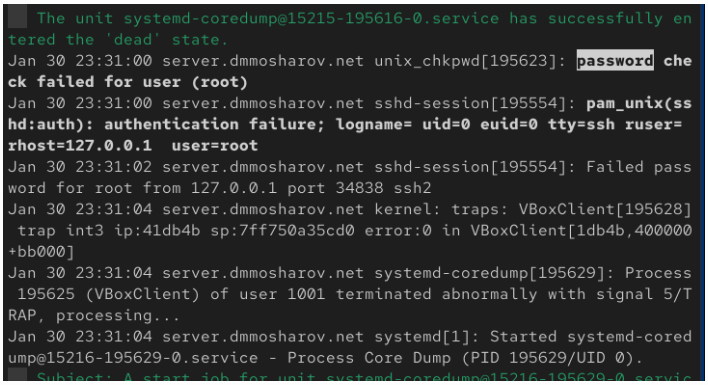
Теперь на клиенте попробуем подключиться по ssh под пользователем root и увидим, что пароль неверный. Это не так, но это значит, что авторизация была запрещена (рис.

[-@fig:007]).



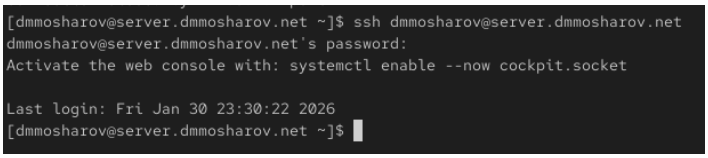
Повторная авторизация под root по ssh

В логах увидим, что авторизация провалена, и не удалось проверить пароль (рис. [-@fig:008]).



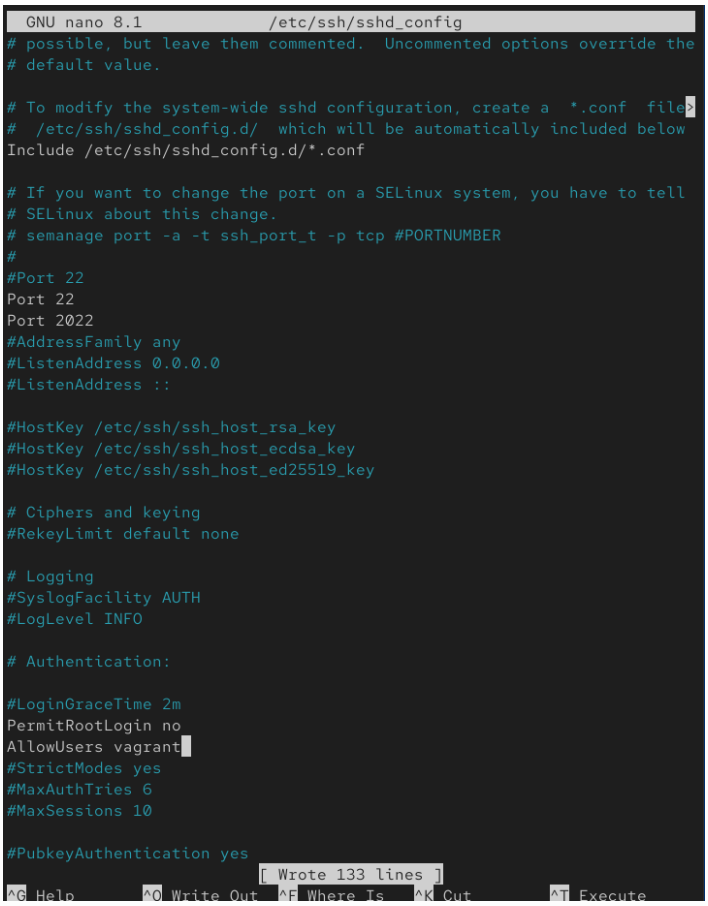
Лог авторизации

Попробуем подключиться по ssh к серверу от имени своей учётной записи. Как видим, авторизация успешна (рис. [-@fig:009]).



Авторизация под своей учётной записью по ssh

Теперь в файле конфигурации sshd пропишем список разрешённых пользователей для авторизации. Пока это будет 1 пользователь - vagrant (рис. [-@fig:010]).



Добавление белого списка пользователей

Перезапустим sshd (рис. [-@fig:011]).

```
[root@server.dmmosharov.net ~]# systemctl restart sshd
[root@server.dmmosharov.net ~]#
```

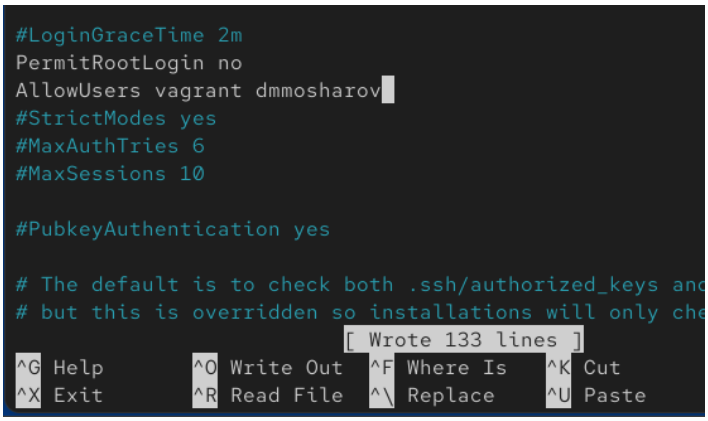
Перезапуск sshd

Вновь на клиенте попробуем авторизаваться под пользователем. Но теперь операция авторизации была провалена, поскольку пользователя нет в белом списке, обозначенном на прошлом этапе (рис. [-@fig:012]).

```
[dmmosharov@server.dmmosharov.net ~]$ ssh dmmosharov@server.dmmosharov.net
dmmosharov@server.dmmosharov.net's password:
Permission denied, please try again.
dmmosharov@server.dmmosharov.net's password:
```

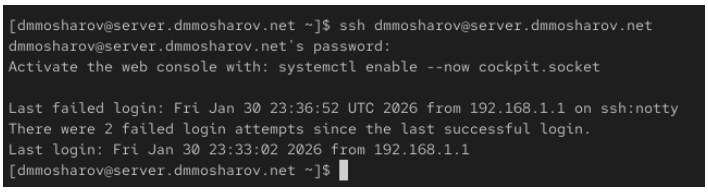
Провальная авторизация под своей учётной записью

Снова откроем файл конфигурации на сервере и добавим пользователя в белый список (рис. [-@fig:013]).



Добавление нового пользователя в белый

После перезапуска службы мы увидим, что подключение с клиента на сервер под учётной записью пользователя проходит успешно (рис. [-@fig:014]).



Успешная авторизация под своей учётной записью

Теперь в файле конфигурации на сервере добавим рабочие порты 22 и 2022 (рис. [-@fig:015]).

Добавление портов

Теперь с клиента попробуем подключиться к серверу по ssh, используя добавленные нами порты. Как видим, оба подключения прошли успешно (рис. [-@fig:019]).

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

Last login: Fri Jan 30 23:36:59 2026 from 192.168.1.1
[dmmosharov@server.dmmosharov.net ~]$ sudo -i
[sudo] password for dmmosharov:
[root@server.dmmosharov.net ~]# logout
[dmmosharov@server.dmmosharov.net ~]$ sudo -i
[root@server.dmmosharov.net ~]# logout
[dmmosharov@server.dmmosharov.net ~]$
logout
Connection to server.dmmosharov.net closed.
[dmmosharov@server.dmmosharov.net ~]$ ssh -p2022 dmmosharov@server.dmmosharov.net
dmmosharov@server.dmmosharov.net's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Fri Jan 30 23:47:44 2026 from 192.168.1.1
[dmmosharov@server.dmmosharov.net ~]$ sudo -i
[sudo] password for dmmosharov:
[root@server.dmmosharov.net ~]#
logout
[dmmosharov@server.dmmosharov.net ~]$
logout
Connection to server.dmmosharov.net closed.
[dmmosharov@server.dmmosharov.net ~]$
```

Проверка портов

Вновь откроем файл конфигурации и разрешим авторизацию по ключу (рис. [-@fig:020]).

```
  GNU nano 8.1                /etc/ssh/sshd_config
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
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 dmmosharov
PubkeyAuthentication yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

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

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
                        [ Wrote 134 lines ]
^G Help       ^O Write Out  ^F Where Is   ^K Cut        ^T Execute
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

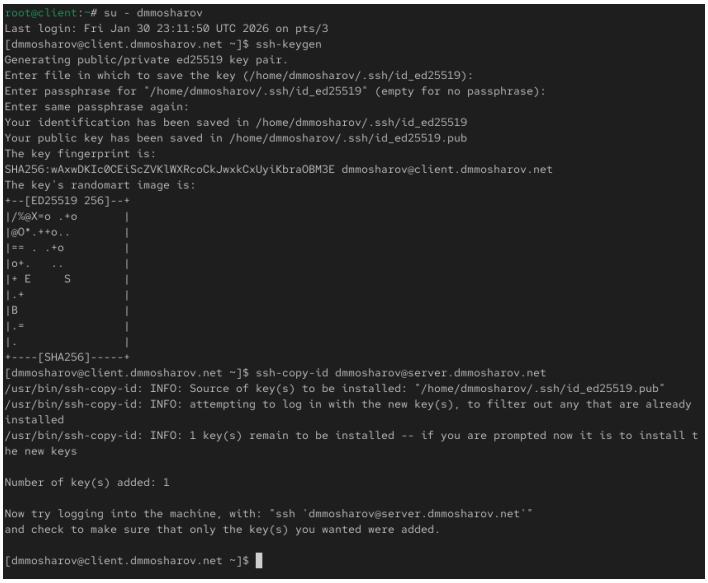
Включение авторизации по ключу

Перезапустим sshd (рис. [-@fig:021]).

```
[root@server.dmmosharov.net ~]# systemctl restart sshd
[root@server.dmmosharov.net ~]#
```

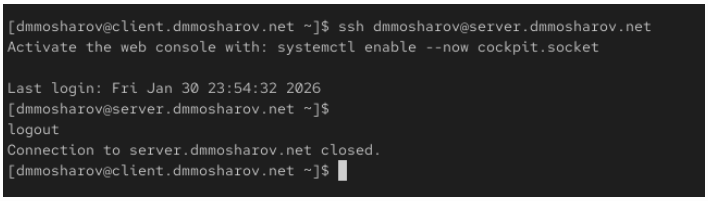
Перезапуск sshd

Теперь на клиенте сгенерируем rsa ключ и отправим его на сервер (рис. [-@fig:022]).



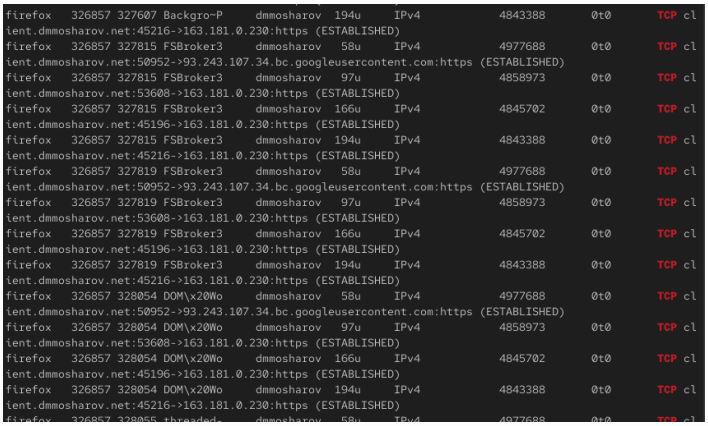
Формирование ключа для авторизации

Теперь попробуем подключиться к серверу по ssh, и на этот раз у нас не спросят пароль (рис. [-@fig:023]).



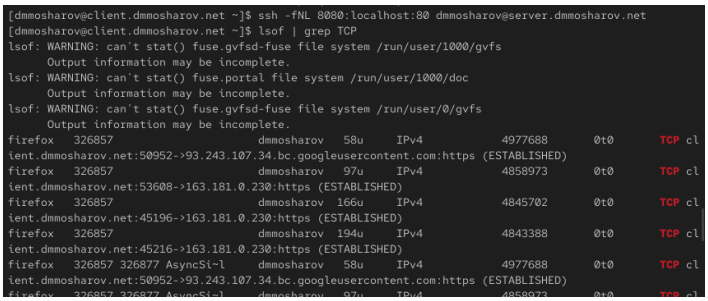
Подключение без пароля

Теперь выведем список служб, использующих TCP (рис. [-@fig:024]).



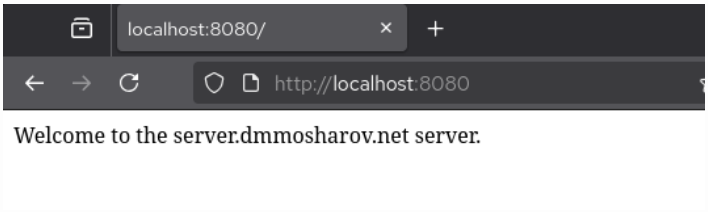
Список служб, использующих TCP

Теперь перенаправим порт 8080 на server.nsandryushin.net на 80 порт локалхоста (рис. [-@fig:025]).



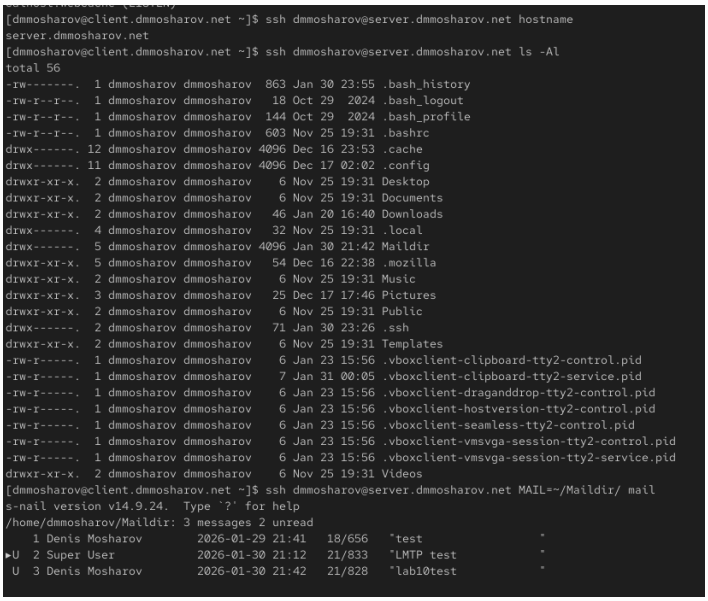
Перенаправление порта

Теперь, обращаясь к localhost:8080 мы видим наш сайт по адресу server.nsandryushin.net:80 (рис. [-@fig:027]).



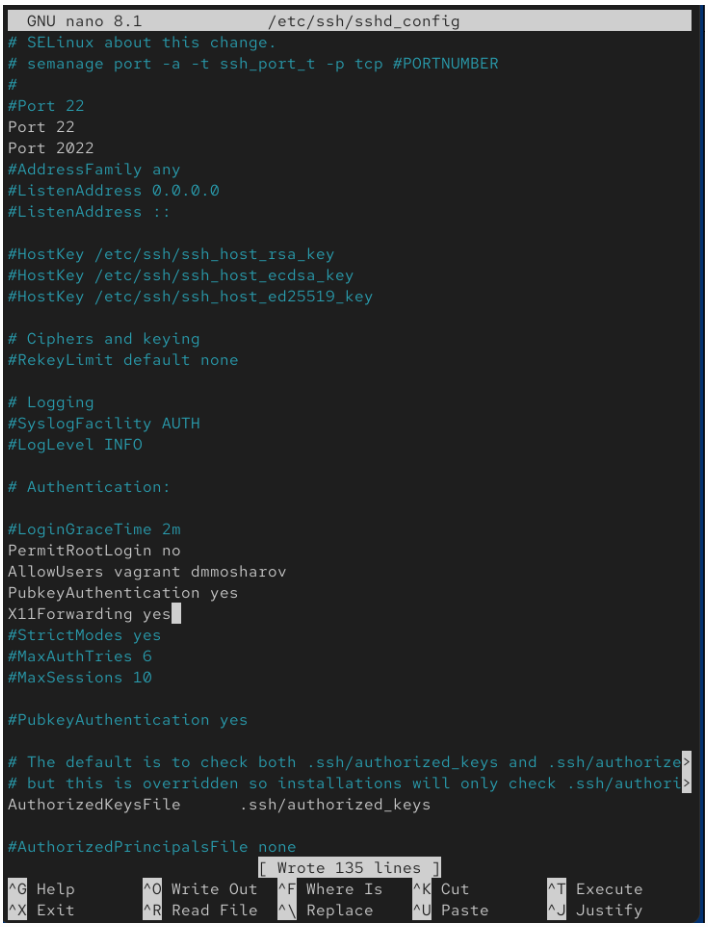
Результат переадресации

На клиенте теперь попробуем вызвать несколько консольных утилит через ssh на сервере, а именно вывод имени хоста, вывод содержимого домашнего каталога и вывод почты (рис. [-@fig:028]).



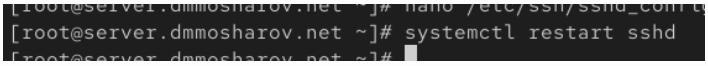
Запуск консольных утилит по ssh

В конфигурационном файле sshd на сервере включим форвардинг X11 (рис. [-@fig:029]).



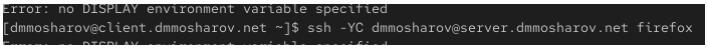
Включение форвардинга иксов

Перезапустим sshd (рис. [-@fig:030]).



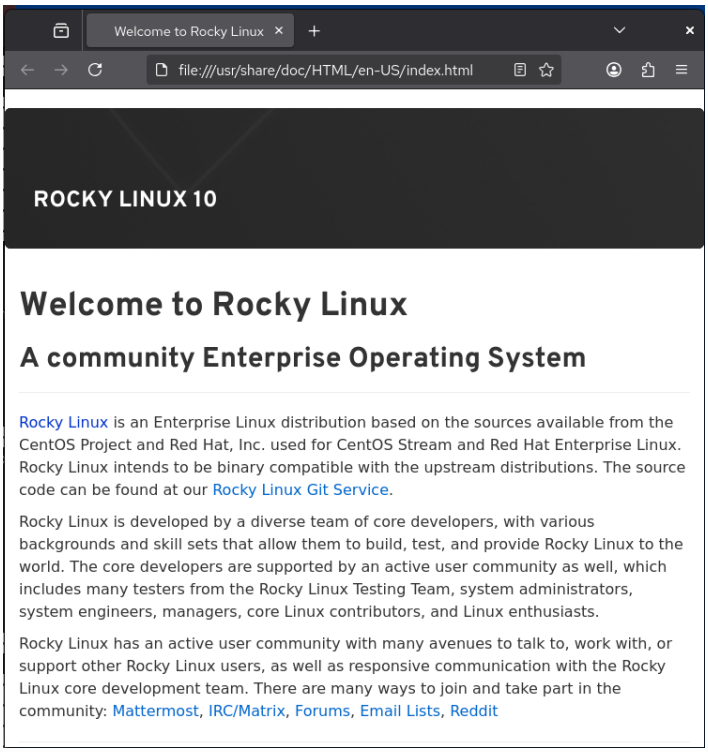
Перезапуск sshd

И запустим графическое приложение, например firefox, по ssh (рис. [-@fig:031]).



Запуск графического приложения по ssh

Как видим, на сервере запустился firefox (рис. [-@fig:032]).



Результат запуска графического приложения по ssh

Теперь сохраним файлы, с которыми велась работа, в vagrant (рис. [-@fig:033]).

```
[root@server.dmmosharov.net ~]# cd /vagrant/provision/server
[root@server.dmmosharov.net server]# mkdir -p /vagrant/provision/server
/ssh/etc/ssh
[root@server.dmmosharov.net server]# cp -R /etc/ssh/sshd_config /vagran
t/provision/server/ssh/etc/ssh/
[root@server.dmmosharov.net server]# cd /vagrant/provision/server
[root@server.dmmosharov.net server]# touch ssh.sh
[root@server.dmmosharov.net server]# chmod +x ssh.sh
[root@server.dmmosharov.net server]# nano ssh.sh
```

Сохранение в vagrant

Создадим скрипт ssh.sh и наполним его следующим содержимым (рис. [-@fig:034]).

```
 GNU nano 8.1                    ssh.sh
#!/bin/bash
echo "Provisioning script $0"
echo "Copy configuration files"
cp -R /vagrant/provision/server/ssh/etc/* /etc
restorecon -vR /etc
echo "Configure firewall"
firewall-cmd --add-port=2022/tcp
firewall-cmd --add-port=2022/tcp --permanent
echo "Tuning SELinux"
semanage port -a -t ssh_port_t -p tcp 2022
echo "Restart sshd service"
systemctl restart sshd
```

ssh.sh

И добавим этот скрипт в vagrantfile (рис. [-@fig:035]).

Vagrantfile

Выводы

В результате выполнения лабораторной работы были получены навыки настройки ssh, форвардинга через него и настройки авторизации