

Департамент образования и науки города Москвы  
Государственное автономное образовательное учреждение  
высшего образования города Москвы  
«Московский городской педагогический университет»  
Институт цифрового образования  
Департамент информатики управления и технологий

Ли Александр Андреевич БД-241м

**Практическая работа 2-1. Основы Linux**

Направление подготовки/специальность

38.04.05 - Бизнес-информатика

Бизнес-аналитика и большие данные

(очная форма обучения)

Вариант 12

Москва

2024

## Содержание

<b>Задачи .....</b>	<b>3</b>
<b>Основная часть .....</b>	<b>3</b>
<b>1.Установка Ubuntu 24 в VirtualBox 7.0.x 1.1. ....</b>	<b>3</b>
<b>2.Предварительная настройка системы.....</b>	<b>6</b>
Создание пользователя ansible .....	9
<b>Предварительная настройка системы .....</b>	<b>15</b>
<b>3.Установка SSH-сервер .....</b>	<b>17</b>
Подключение к Mobaextern по ssh.....	25
<b>Вариант 12.....</b>	<b>27</b>
<b>Заключение .....</b>	<b>30</b>

## Введение

Практическая работа нацелена на знакомство студентов с основами работы в Linux, установку системы, проведение предварительной настройки системы и настройка SSH на Ubuntu 24.

## Задачи

1. Установить Ubuntu 24 в VirtualBox 7.0.x.
2. Выполнить предварительную настройку системы.
3. Настроить SSH-сервер.

## Основная часть

### 1. Установка Ubuntu 24 в VirtualBox 7.0.x 1.1.

Установлен VirtualBox 7.0.x с официального сайта.

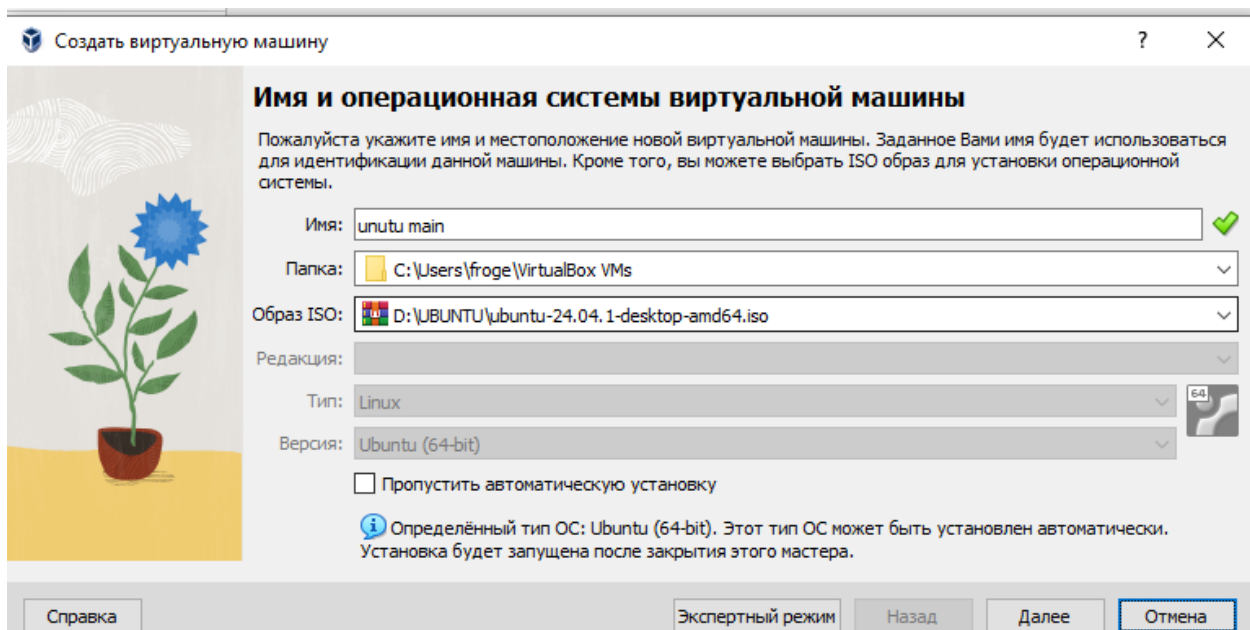


Рис.1

Выбран iso образ с нужной версией ubuntu.

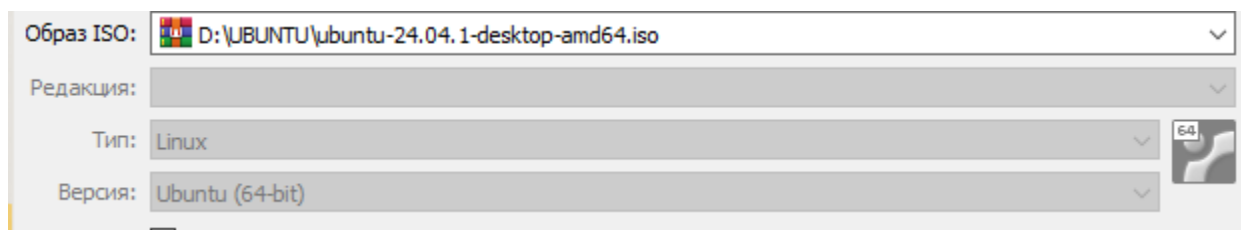


Рис.2

Создана новая виртуальная машина в VirtualBox с параметрами: тип ОС Linux, версия: Ubuntu (64-bit).

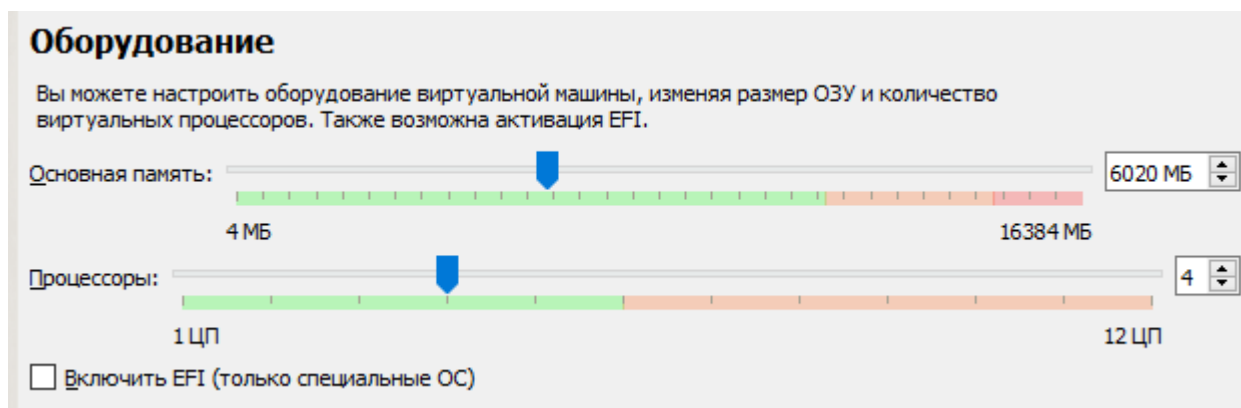


Рис.3

Было выделено 6 ГБ оперативной памяти и 4 виртуальных процессоров.

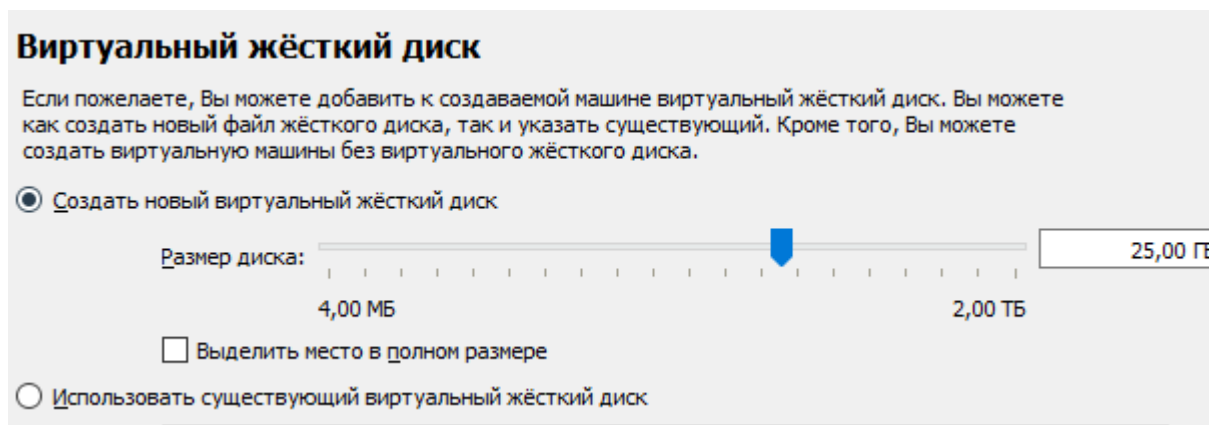


Рис.4

Создан виртуальный жесткий диск с объемом в 25 ГБ.

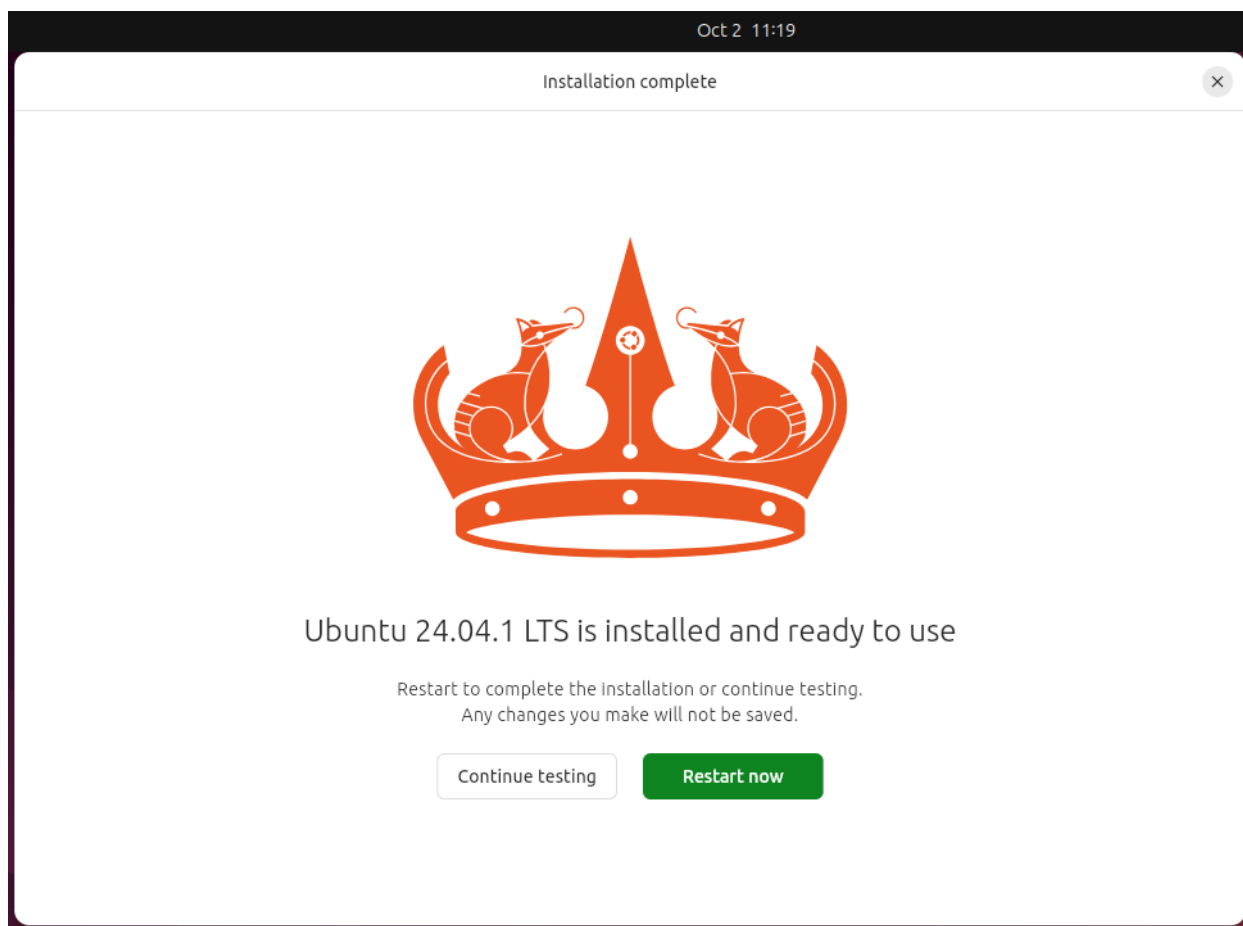
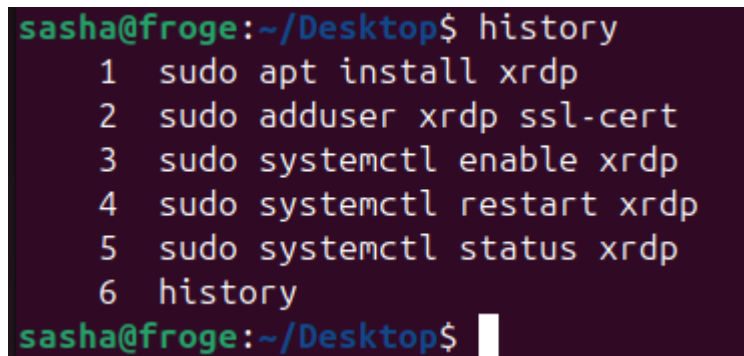


Рис.5

После запуска виртуальной машины была выполнена установка Ubuntu, следуя инструкциям мастера установки.

## 2. Предварительная настройка системы

Сначала, необходимо установить xrdp, для удобного подключения и возможности копировать и вставлять команды в терминале.



```
sasha@froge:~/Desktop$ history
1  sudo apt install xrdp
2  sudo adduser xrdp ssl-cert
3  sudo systemctl enable xrdp
4  sudo systemctl restart xrdp
5  sudo systemctl status xrdp
6  history
sasha@froge:~/Desktop$
```

Рис.6

Для начала необходимо установить xrdp посредством команды **sudo apt install xrdp**. После того, как все установится, нужно создать пользователя командой **sudo adduser xrdp ssl-cert**, параметр ssl-cert отвечает защищенный канал передачи данных. В конце включаем автоматическую запуску сервиса xrdp при запуске системы.

sudo apt install xrdp

sudo adduser xrdp ssl-cert

sudo systemctl enable xrdp

sudo systemctl restart xrdp

sudo systemctl status xrdp

history

```
sasha@froge:~$ sudo systemctl status xrdp
[sudo] password for sasha:
● xrdp.service - xrdp daemon
   Loaded: loaded (/usr/lib/systemd/system/xrdp.service; enabled; preset: ena>
   Active: active (running) since Wed 2024-10-02 11:26:20 MSK; 11min ago
     Docs: man:xrdp(8)
           man:xrdp.ini(5)
  Main PID: 4657 (xrdp)
    Tasks: 2 (limit: 6971)
  Memory: 14.0M (peak: 15.0M)
     CPU: 1.451s
   CGroup: /system.slice/xrdp.service
           └─4657 /usr/sbin/xrdp
             └─6844 /usr/sbin/xrdp

Oct 02 11:35:53 froge xrdp[6844]: [INFO ] xrdp_wm_log_msg: sesman connect ok
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] sesman connect ok
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] sending login info to session manager>
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] xrdp_wm_log_msg: login successful for>
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] login successful for user sasha on di>
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] loaded module 'libxup.so' ok, interfa>
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] started connecting
Oct 02 11:35:53 froge xrdp[6844]: [INFO ] lib_mod_connect: connecting via UNIX >
```

Рис.6

Пропишем **sudo systemctl status xrdp**, чтобы проверить работу xrdp.

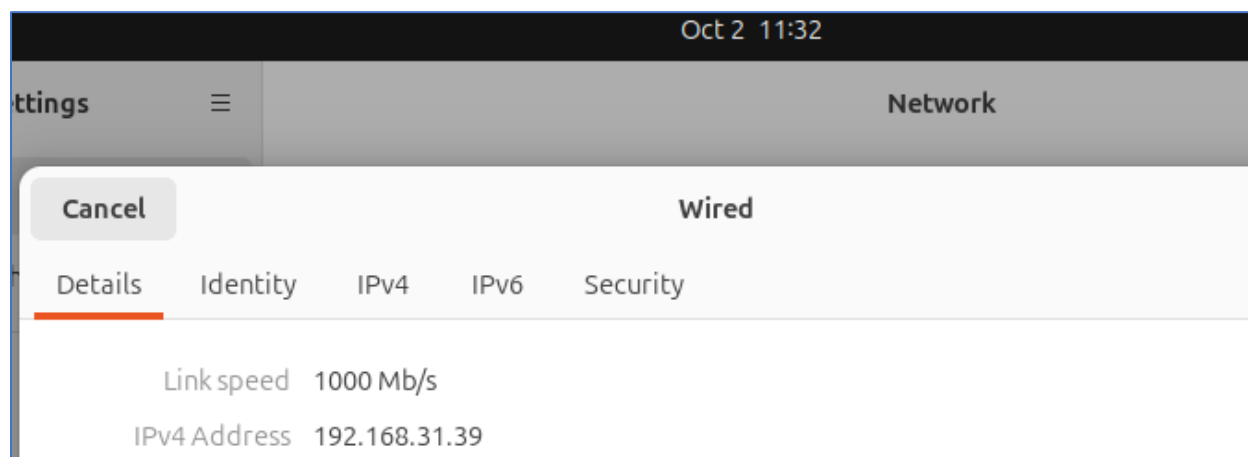


Рис.7

Далее, узнаем адрес виртуальной машины в настройках или посредством команды **ip a**.

```
sasha@froge: ~/Desktop
sasha@froge:~/Desktop$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:44:05:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.31.39/24 brd 192.168.31.255 scope global dynamic noprefixroute enp0s3
        valid_lft 43156sec preferred_lft 43156sec
sasha@froge:~/Desktop$
```

Рис.8

После того как был запущен xrdp и определен ip адрес, подключаемся к виртуальной машине при помощи программы “Подключение к удаленному рабочему столу”.

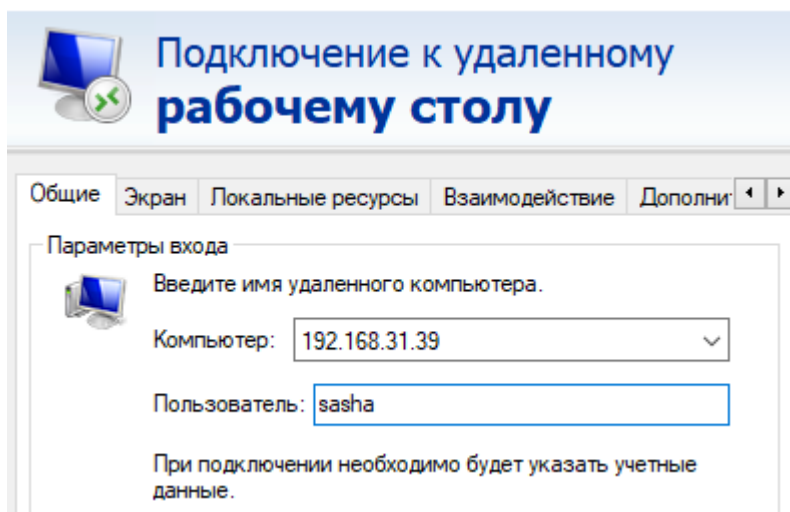


Рис.9

Прописываем ip адрес и пользователя под которым необходимо зайти.



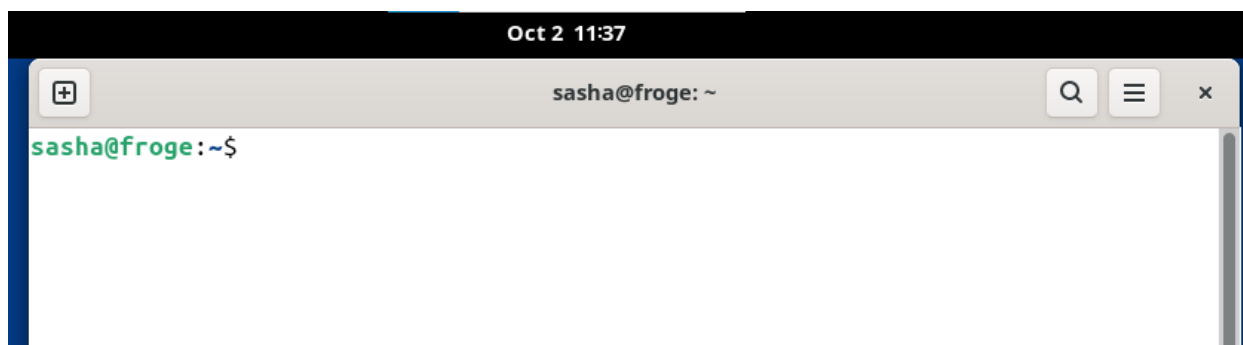


Рис.10

После выполнения описанных выше действия, можно подключаться к виртуальной машине посредством программы “Подключение к удаленному рабочему столу”, а также иметь возможность вставлять и копировать текст.

- 1 sudo apt install xrdp
- 2 sudo adduser xrdp ssl-cert
- 3 sudo systemctl enable xrdp
- 4 sudo systemctl restart xrdp
- 5 sudo systemctl status xrdp
- 6 history
- 7 ip a
- 8 history

### Создание пользователя ansible

Для того чтобы создать пользователя с домашней директорией /home/ansible, необходимо прописать команду

**sudo adduser ansible --home /home/ansible** после чего зададим пароль “Qq1234!Aa!”.

```
sasha@froge:~$ sudo adduser ansible --home /home/ansible
info: Adding user `ansible' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `ansible' (1001) ...
info: Adding new user `ansible' (1001) with group `ansible (1001)' ...
info: Creating home directory `/home/ansible' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for ansible
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
info: Adding new user `ansible' to supplemental / extra groups `users' ...
info: Adding user `ansible' to group `users' ...
sasha@froge:~$
```

---

Рис.11

Создали пользователя ansible и задали ему пароль.

```
sasha@froge:~$ sudo passwd ansible
New password:
Retype new password:
passwd: password updated successfully
sasha@froge:~$
```

---

Рис.12

Далее, сменим пароль пользователю на “!1234!Qq” используя команду **sudo passwd ansible**.

11 sudo passwd ansible

```
sasha@froge:~$ sudo -i
root@froge:~# sudo passwd root
New password:
Retype new password:
passwd: password updated successfully
root@froge:~#
```

---

Рис.13

Сменим root пароль на !1234!Qq командой **sudo -i sudo passwd**

```
root@froge:~# sudo groupadd wheel
root@froge:~#
```

Рис.14

Создадим группу wheel командой

**sudo groupadd wheel.**

```
sasha:x:1000:
xrdp:x:124:
ansible:x:1001:
wheel:x:1002:
root@froge:~#
```

Рис.15

Проверим создалась ли группа командой

**cat /etc/group** , после чего в терминале отобразятся все созданные группы.

```
root@froge:~# sudo mkdir /admin
root@froge:~# sudo chown ansible:wheel /admin
root@froge:~# history
 1  exit
 2  sudo passwd root
 3  cat /etc/group
 4  sudo groupadd wheel
 5  cat /etc/group
 6  history
 7  clear
 8  sudo mkdir /admin
 9  sudo chown ansible:wheel /admin
10  history
root@froge:~#
```

Рис.16

Создадим директорию /admin командой

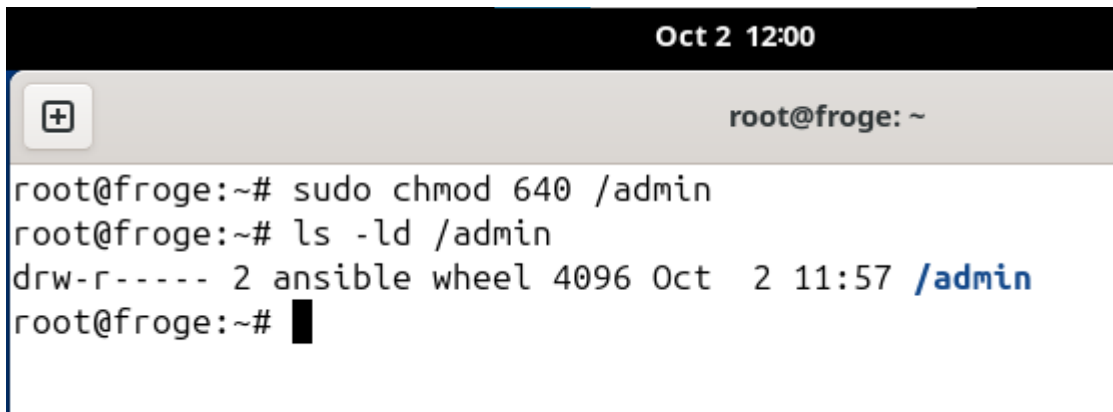
**sudo mkdir /admin**

```
9 sudo chown ansible:wheel /admin
10 history
root@froge:~#
```

Рис.17

Далее, сделаем пользователя ansible владельцем директории /admin и назначим его в группу wheel командой

**sudo chown ansible:wheel /admin**

A terminal window titled "Oct 2 12:00" with a tab labeled "root@froge: ~". The terminal shows the following commands and output:

```
root@froge:~# sudo chown ansible:wheel /admin
root@froge:~# ls -ld /admin
drw-r----- 2 ansible wheel 4096 Oct  2 11:57 /admin
root@froge:~#
```

Рис.18

После чего, установим права на директорию /admin: чтение и запись для пользователя ansible, только чтение для группы wheel

**sudo chmod 640 /admin**

**ls -ld /admin** для просмотра права на каталог

Далее отредактируем файл /etc/sudoers при помощи nano, чтобы разрешить группе wheel использовать sudo с паролем

```
Oct 2 12:04
root@froge: /home/sasha
GNU nano 7.2 /etc/sudoers *
# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
%wheel  ALL=(ALL:ALL) ALL
# See sudoers(5) for more information on "@include" directives:

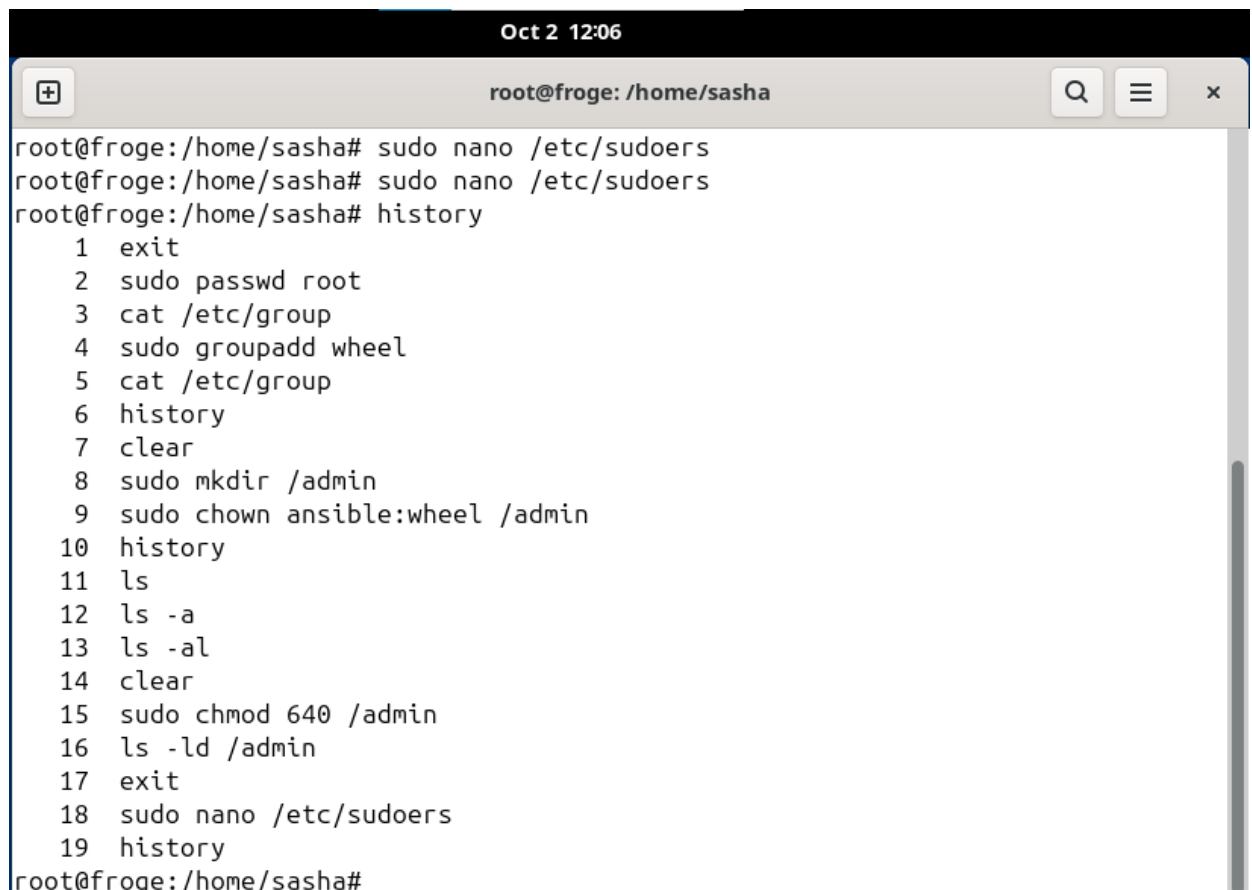
@includedir /etc/sudoers.d

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify  ^/ Go To Line
```

Рис.19

При помощи команды **sudo nano /etc/sudoers sudo visudo**, зайдём в файл и пропишем

**%wheel ALL=(ALL:ALL) ALL**, после чего сохраним изменения (ctrl + o) и выйдем (ctrl + x)

A terminal window titled "root@froge: /home/sasha" with a timestamp "Oct 2 12:06". The window shows a series of commands entered by the user, with the last command being "history". The output of the "history" command is a numbered list of 19 commands. The terminal window has a standard Linux desktop interface with a title bar and window controls.

```
Oct 2 12:06
root@froge: /home/sasha
root@froge:/home/sasha# sudo nano /etc/sudoers
root@froge:/home/sasha# sudo nano /etc/sudoers
root@froge:/home/sasha# history
  1  exit
  2  sudo passwd root
  3  cat /etc/group
  4  sudo groupadd wheel
  5  cat /etc/group
  6  history
  7  clear
  8  sudo mkdir /admin
  9  sudo chown ansible:wheel /admin
 10  history
 11  ls
 12  ls -a
 13  ls -al
 14  clear
 15  sudo chmod 640 /admin
 16  ls -ld /admin
 17  exit
 18  sudo nano /etc/sudoers
 19  history
root@froge:/home/sasha#
```

Рис.20

root@froge:/home/sasha# history

- 1 exit
- 2 sudo passwd root
- 3 cat /etc/group
- 4 sudo groupadd wheel
- 5 cat /etc/group
- 6 history
- 7 clear
- 8 sudo mkdir /admin
- 9 sudo chown ansible:wheel /admin
- 10 history
- 11 ls
- 12 ls -a
- 13 ls -al

```
14 clear
15 sudo chmod 640 /admin
16 ls -ld /admin
17 exit
18 sudo nano /etc/sudoers
19 history
```

### Предварительная настройка системы

Создадим нового пользователя командой

**sudo adduser newuser**

```
sasha@froge:~$ sudo adduser newuser
[sudo] password for sasha:
info: Adding user `newuser' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `newuser' (1003) ...
info: Adding new user `newuser' (1003) with group `newuser (1003)' ...
info: Creating home directory `/home/newuser' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for newuser
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
info: Adding new user `newuser' to supplemental / extra groups `users' ...
info: Adding user `newuser' to group `users' ...
```

Рис.21

Сначала, добавим пользователя в группу sudo, создадим новую группу и добавим пользователя в нее и проверим его членства в группах

**sudo usermod -aG sudo newuser**

**sudo groupadd newgroup**

**sudo usermod -aG newgroup newuser**

**groups newuser**

```
sasha@froge:~$ sudo usermod -aG sudo newuser
sasha@froge:~$ sudo groupadd newgroup
sasha@froge:~$ sudo usermod -aG newgroup newuser
sasha@froge:~$ groups newuser
newuser : newuser sudo users newgroup
sasha@froge:~$
```

Рис.22

```
34 sudo adduser newuser
35 sudo su
36 sudo adduser newuser
37 sudo usermod -aG sudo newuser
38 sudo groupadd newgroup
39 sudo usermod -aG newgroup newuser
40 groups newuser
41 history
sasha@froge:~$
```

Рис.23



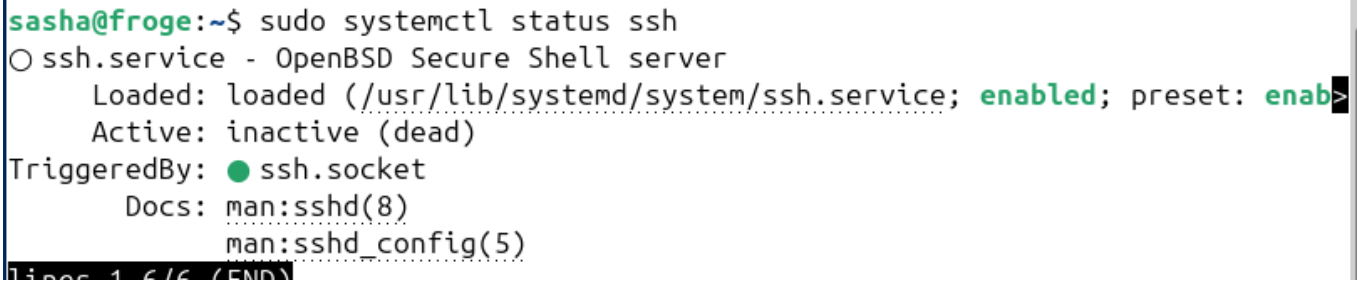
### 3. Установка SSH-сервер

Для начала, установим SSH-сервер, затем проверим, работает ли служба SSH, и включена ли она для автоматического запуска при загрузке.

```
apt install openssh-server
```

```
sudo systemctl status ssh
```

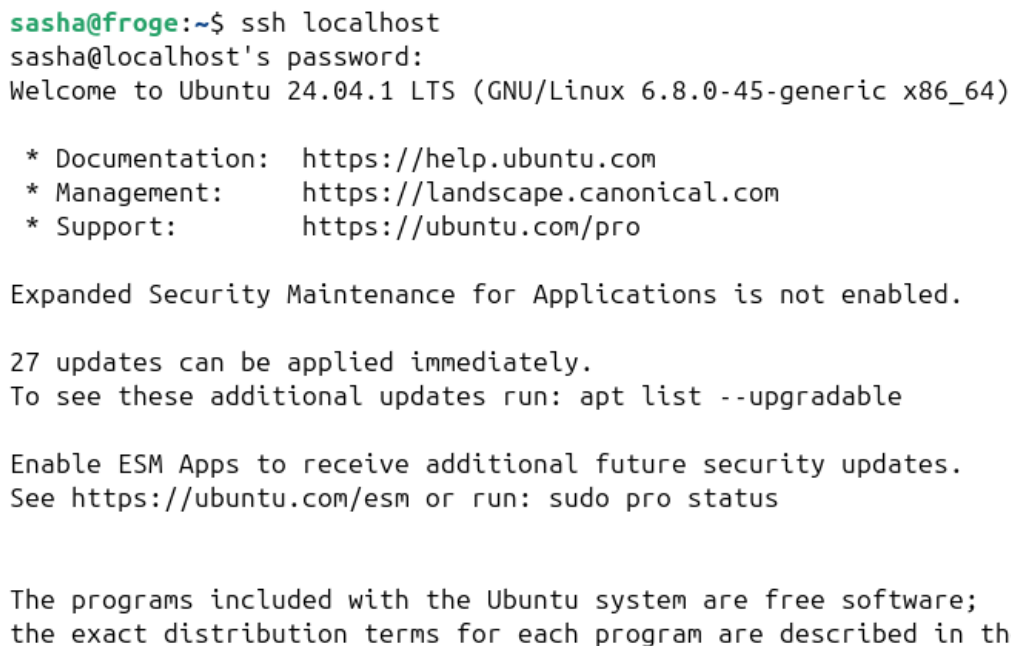
```
sudo systemctl enable ssh
```



```
sasha@froge:~$ sudo systemctl status ssh
○ ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enab>
   Active: inactive (dead)
 TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
        man:sshd_config(5)
lines 1-6/6 (END)
```

Рис.24

Проверка работы ssh сервера.



```
sasha@froge:~$ ssh localhost
sasha@localhost's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

27 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
```

Рис.25

Пропишем **ssh localhost** для установления SSH-сессии с локальным компьютером .

```
19  sudo apt install openssh-server
20  sudo systemctl enable sshd
21  sudo systemctl enable ssh
22  sudo systemctl status ssh
23  history
sasha@froge:~$
```

Рис.24

### Настроим конфигурацию SSH

Необходимо отредактировать файл конфигурации SSH (sshd\_config), который находится в /etc/ssh/sshd\_config, командой

**sudo nano /etc/ssh/sshd\_config**

Изменим порт SSH со стандартного (22), на ( 2222) для большей безопасности.



The screenshot shows a terminal window titled "Oct 2 12:30" with the prompt "root@froge: /home/sasha". The terminal is running the nano text editor, editing the file "/etc/ssh/sshd\_config". The file content is as follows:

```
GNU nano 7.2 /etc/ssh/sshd_config *

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr
# 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.

Include /etc/ssh/sshd_config.d/*.conf

Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
```

At the bottom of the terminal, there is a status bar with the following keyboard shortcuts:


<b>^G</b> Help	<b>^O</b> Write Out	<b>^W</b> Where Is	<b>^K</b> Cut	<b>^T</b> Execute	<b>^C</b> Location
<b>^X</b> Exit	<b>^R</b> Read File	<b>^\</b> Replace	<b>^U</b> Paste	<b>^J</b> Justify	<b>^_</b> Go To Line

Рис.25

Изменим в файле конфигурации “Port 22” на “Port 2222”

Также, изменим **PermitRootLogin**, который разрешает или запрещает вход под root.

**PermitRootLogin no**



```
Oct 2 12:32
root@froge: /home/sasha
GNU nano 7.2 /etc/ssh/sshd_config *

# Authentication:

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

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Рис.26

Пропишем в файле конфигурации PermitRootLogin no

Изменим **PasswordAuthentication**, который разрешает аутентификацию при помощи пароля

**PasswordAuthentication yes**

```
Oct 2 12:33
root@froge: /home/sasha
GNU nano 7.2 /etc/ssh/sshd_config *
# 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
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes
#PermitEmptyPasswords no
```

Рис.27

После внесения изменений, сохраним файл при помощи (Ctrl + O), затем Enter, и выйдем с помощью (Ctrl + X).

После изменения конфигурации перезапустим SSH-сервис:

**sudo systemctl restart ssh**

```
sasha@froge:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enab>
   Active: active (running) since Wed 2024-10-02 13:15:59 MSK; 5s ago
   TriggeredBy: ● ssh.socket
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 2966 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2968 (sshd)
     Tasks: 1 (limit: 6971)
    Memory: 1.2M (peak: 1.5M)
       CPU: 17ms
    CGroup: /system.slice/ssh.service
           └─2968 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Oct 02 13:15:59 froge systemd[1]: Starting ssh.service - OpenBSD Secure Shell s>
Oct 02 13:15:59 froge sshd[2968]: Server listening on :: port 2222.
Oct 02 13:15:59 froge systemd[1]: Started ssh.service - OpenBSD Secure Shell se>
lines 1-17/17 (END)
```

Рис.28

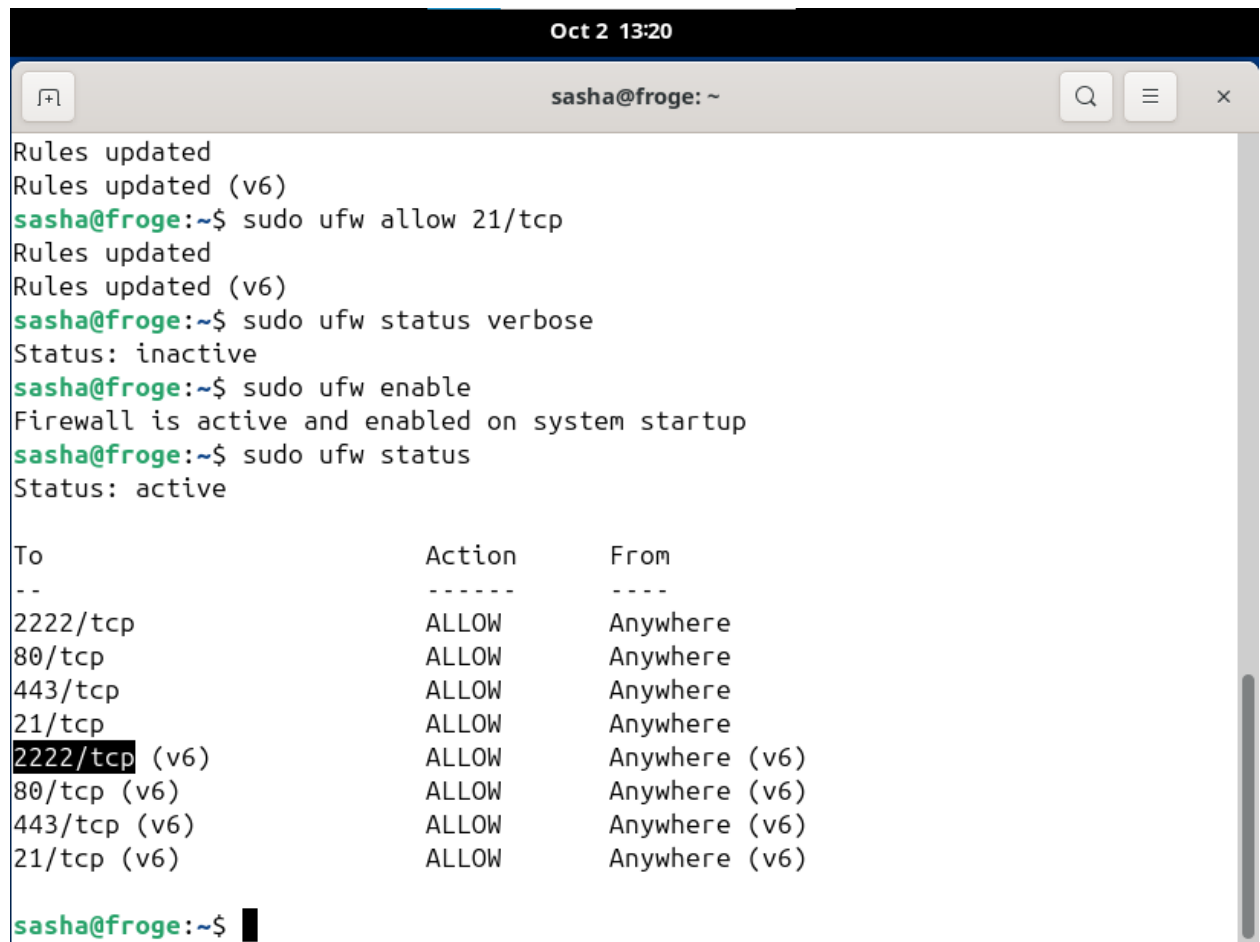
**Настроим брандмауэр UFW (Uncomplicated Firewall), необходимо разрешить SSH:**

**Sudo ufw allow ssh**

## **sudo ufw allow 2222/tcp**

Также, сразу проверим статус UFW, чтобы убедиться, что правило применено

### **sudo ufw status**



A terminal window titled 'sasha@froge: ~' with a timestamp 'Oct 2 13:20'. The terminal shows the following commands and output:

```
Rules updated
Rules updated (v6)
sasha@froge:~$ sudo ufw allow 21/tcp
Rules updated
Rules updated (v6)
sasha@froge:~$ sudo ufw status verbose
Status: inactive
sasha@froge:~$ sudo ufw enable
Firewall is active and enabled on system startup
sasha@froge:~$ sudo ufw status
Status: active
```

To	Action	From
--	-----	----
2222/tcp	ALLOW	Anywhere
80/tcp	ALLOW	Anywhere
443/tcp	ALLOW	Anywhere
21/tcp	ALLOW	Anywhere
2222/tcp (v6)	ALLOW	Anywhere (v6)
80/tcp (v6)	ALLOW	Anywhere (v6)
443/tcp (v6)	ALLOW	Anywhere (v6)
21/tcp (v6)	ALLOW	Anywhere (v6)

sasha@froge:~\$ █

Рис.29

После того, как были прописаны команды выше, можно подключиться к серверу SSH с другого компьютера с помощью команды:

**Ssh -p 2222 sasha@192.168.31.39**

```
sasha@froge: ~
Microsoft Windows [Version 10.0.19045.4894]
(с) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\froge>ssh -p 2222 sasha@192.168.31.39
sasha@192.168.31.39's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

27 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Wed Oct  2 13:22:12 2024 from 192.168.31.227
sasha@froge:~$ history
 1 sudo apt install xrdp
 2 sudo adduser xrdp ssl-cert
 3 sudo systemctl enable xrdp
 4 sudo systemctl restart xrdp
 5 sudo systemctl status xrdp
 6 history
 7 ip a
 8 clear
 9 ls
10 history
11 clear
12 sudo
13 sudo su
14 sudo passwd root
15 history
sasha@froge:~$
```

Рис.30  
серверу SSH с другого компьютера



## Подключение к Mobaextern по ssh

Для начала, установим Mobaextern с официального сайта.

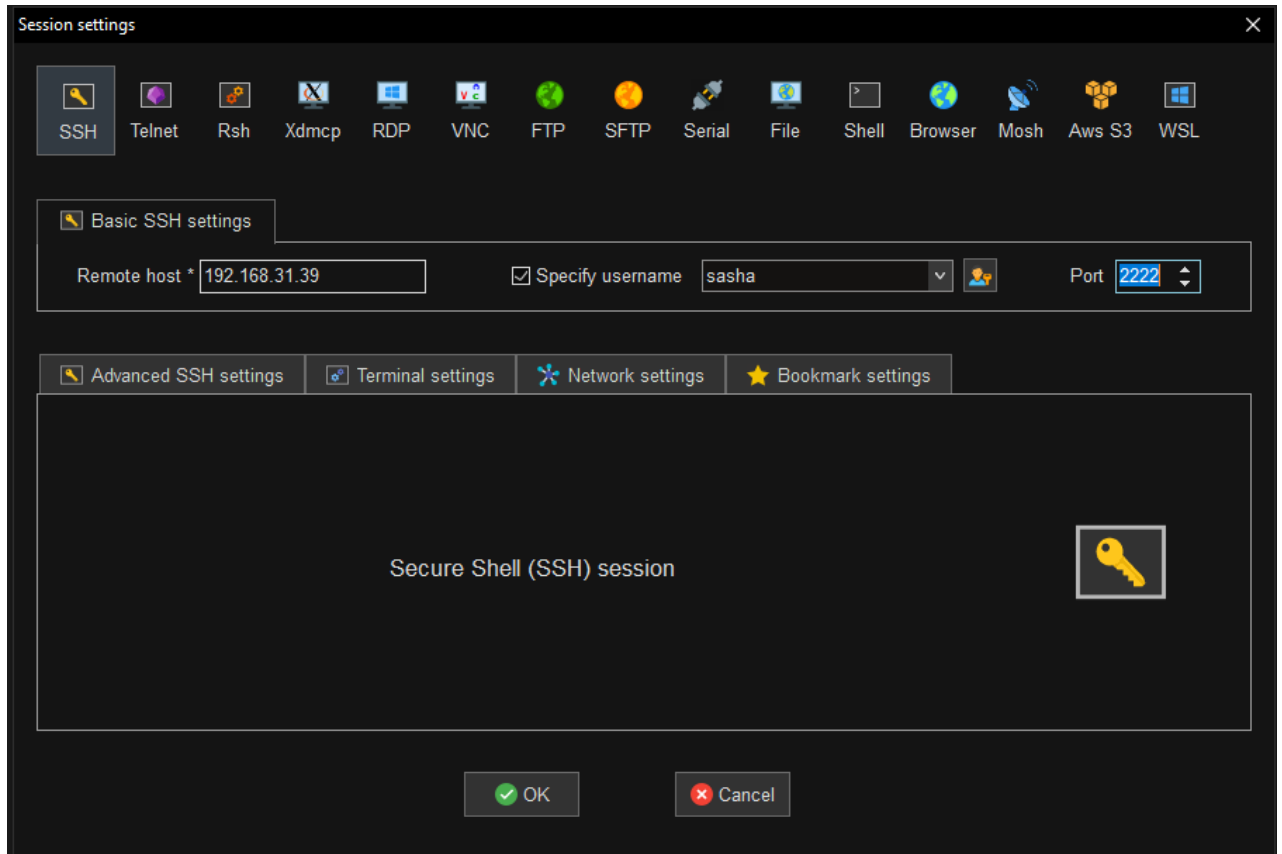


Рис.31

После того как зайдём в приложение необходимо нажать кнопку “ssh”, прописать ip адрес, имя пользователя и порт.

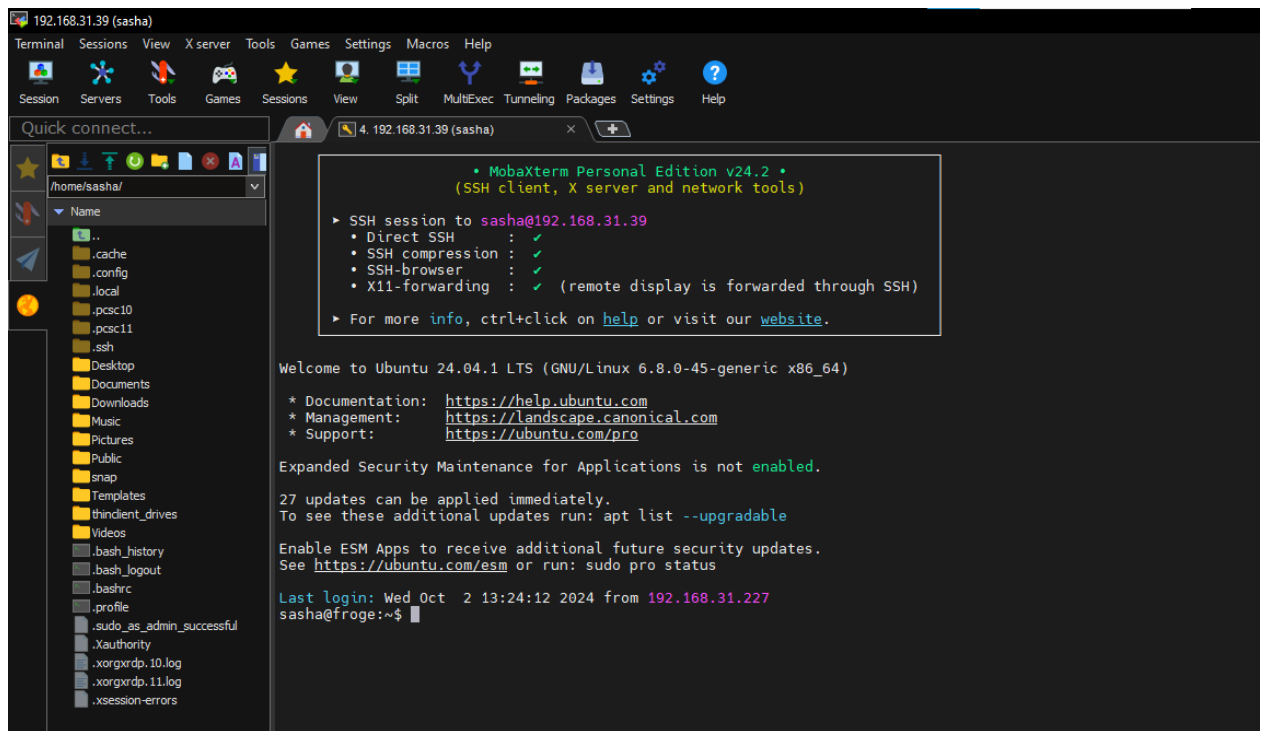


Рис.32

Далее, необходимо ввести пароль, после чего будет получен доступ.

## Вариант 12

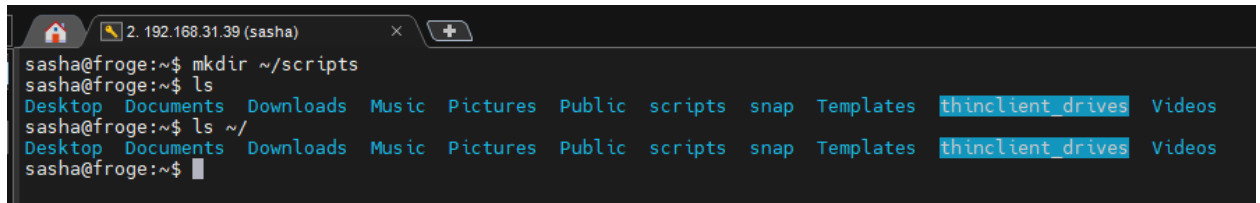
1. Создайте директорию ~/scripts.
2. Переместите файл script.sh из домашнего каталога в директорию ~/scripts.
3. Добавьте директорию ~/scripts в переменную окружения PATH.

Сначала создадим директорию ~/scripts командой

**mkdir ~/scripts**

и проверим её существование командой

**ls**



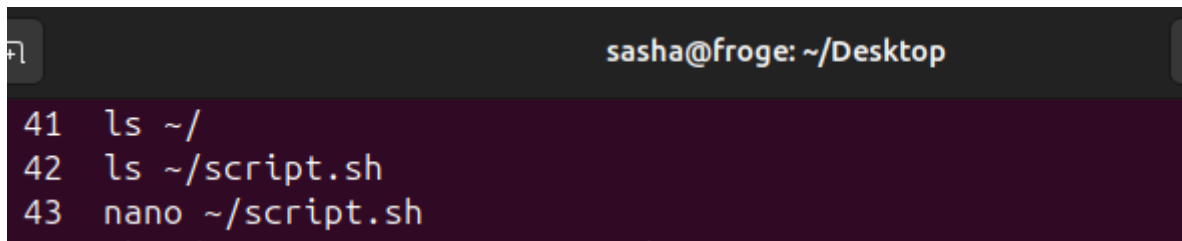
```
sasha@froge:~$ mkdir ~/scripts
sasha@froge:~$ ls
Desktop Documents Downloads Music Pictures Public scripts snap Templates thinclient_drives Videos
sasha@froge:~$ ls ~/
Desktop Documents Downloads Music Pictures Public scripts snap Templates thinclient_drives Videos
sasha@froge:~$
```

Рис.33

Так как ,по умолчанию в Ubuntu файл script.sh не существует. Создадим его при помощи

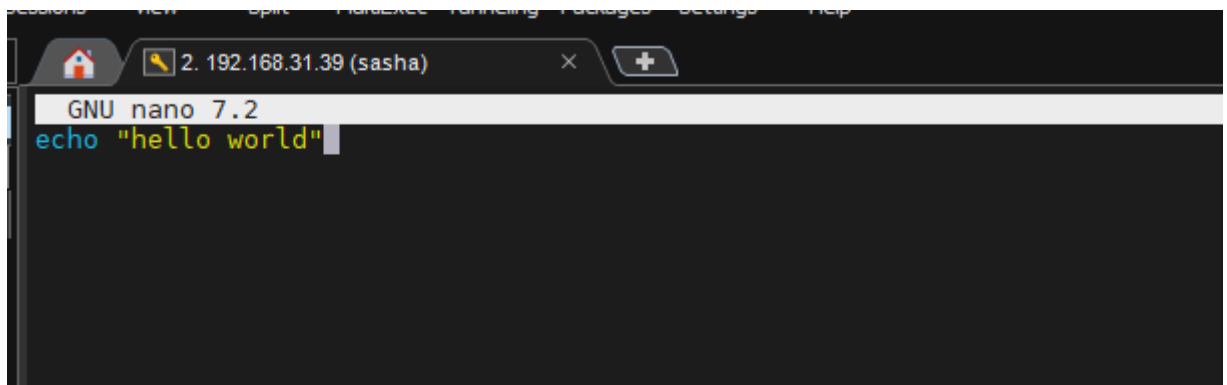
**nano ~/script.sh**

и напомним внутри него простой скрипт, который выводит hello world.



```
sasha@froge: ~/Desktop
41 ls ~/
42 ls ~/script.sh
43 nano ~/script.sh
```

Рис.34



```
GNU nano 7.2
echo "hello world"
```

Рис.35

Содержимое файла.

Далее, переместим файл script.sh из домашнего каталога в директорию

`~/scripts`

`mv ~/script.sh ~/scripts/`

проверим командой `ls` и перейдем в директорию `cd scripts/`

```
sasha@froge:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  scripts  script.sh  snap  Templates  thinclient_drives  Videos
sasha@froge:~$ mv ~/script.sh ~/scripts/
sasha@froge:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  scripts  snap  Templates  thinclient_drives  Videos
sasha@froge:~$ cd scripts/
sasha@froge:~/scripts$ ls
script.sh
sasha@froge:~/scripts$
```

Рис.36

Добавим директорию `~/scripts` в переменную окружения `PATH`. Для этого нужно изменить файл конфигурации оболочки. Сначала нужно открыть файл командой

`nano ~/.bashrc`

```
fi
. ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
export PATH="$PATH:HOME/scripts"
```

Рис.37

В конце файла добавим строку

`export PATH="$PATH:$HOME/scripts"`

Строка выше добавляет `~/scripts` в начало переменной `PATH`. При запуске командная оболочка будет в первую очередь искать исполняемые файлы в `~/scripts`, а затем в остальных каталогах.

```
sasha@froge:~$ cd scripts/
sasha@froge:~/scripts$ ls
script.sh
sasha@froge:~/scripts$ nano ~/.bashrc
sasha@froge:~/scripts$ nano ~/.bashrc
sasha@froge:~/scripts$ source ~/.bashrc
sasha@froge:~/scripts$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:HOME/scripts
sasha@froge:~/scripts$ ^C
sasha@froge:~/scripts$ nano ~/.bashrc
sasha@froge:~/scripts$ source ~/.bashrc
sasha@froge:~/scripts$ echo $PATH
/home/sasha/scripts:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:HOME/scripts
sasha@froge:~/scripts$
```

Рис.38

Применим изменения, загрузив обновленный файл командой **source ~/.bashrc**

Далее, командой

**Echo \$PATH** Выведем содержимое переменного окружения.

```
sasha@froge:~/scripts$ ^C
sasha@froge:~/scripts$ nano ~/.bashrc
sasha@froge:~/scripts$ source ~/.bashrc
sasha@froge:~/scripts$ echo $PATH
/home/sasha/scripts:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
sasha@froge:~/scripts$ cd ~
sasha@froge:~$ ls ~/scripts
script.sh
sasha@froge:~$ ls ~/scripts/script.sh
/home/sasha/scripts/script.sh
sasha@froge:~$
```

Рис.39

Устанавливаем права на выполнение

```
65  chmod +x ~/scripts/script.sh
66  script.sc
67  script.sh
68  clear
69  script.sh
70  history
sasha@froge:~$
```

Рис.40

Запускаем файл

```
sessions view Split MultiExec Tunneling Packages Settings
2. 192.168.31.39 (sasha) x +
sasha@froge:~$ script.sh
hello world
sasha@froge:~$
```

Рис.41

## **Заключение**

В ходе выполнения практической работы, была проведена установка Ubuntu 24 в VirtualBox ,выполнена предварительная настройка системы, настроен SSH-сервер, и выполнены задания по вариантам.