

Содержание

Задача	1
Ход работы	2
Установка Linux на виртуальную машину	2
Настройка SSH	5
Отключение авторизации по паролю	8
Получение информации о сетевом оборудовании	9
Заключение	12

Задача

Изучить основные понятия и сетевые сервисы в ОС Linux

Шаги:

1. Установить ОС Linux на виртуальную машину на хостинге
2. Подключиться к ней по ssh
3. Ввести команды ip (с параметрами) и netstat, рассказать, что они показывают
4. Настроить подключение по ssh с помощью ключей, настроить параметры безопасного подключения

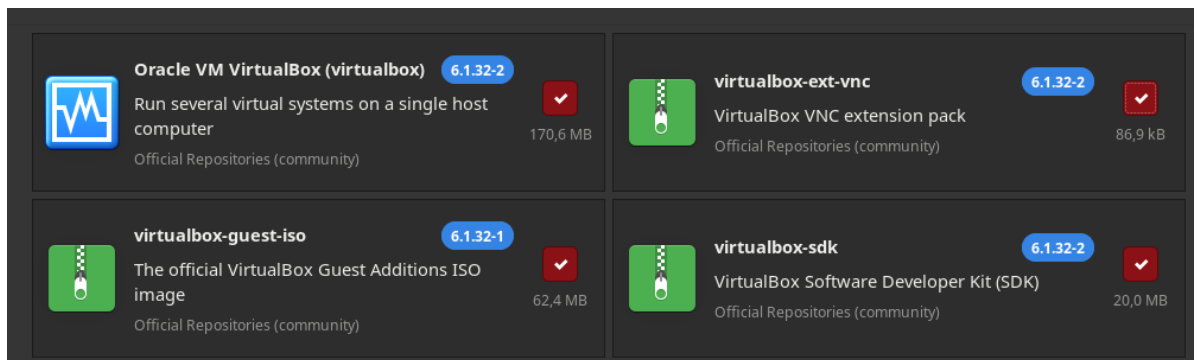
Ход работы

Установка Linux на виртуальную машину

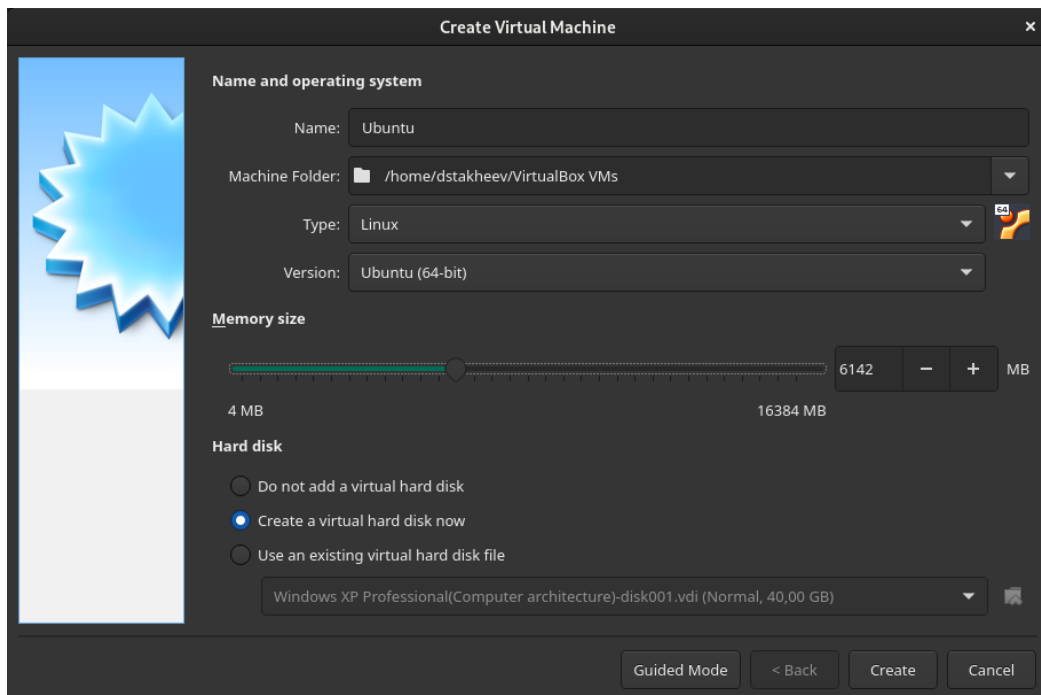
*Инструкция для Manjaro Linux

1. Устанавливаем Oracle VirtualBox Manager и extension pack командой

```
$ sudo pacman -S virtualbox virtualbox-ext-vnc virtualbox-guest-iso virtualbox-sdk
```



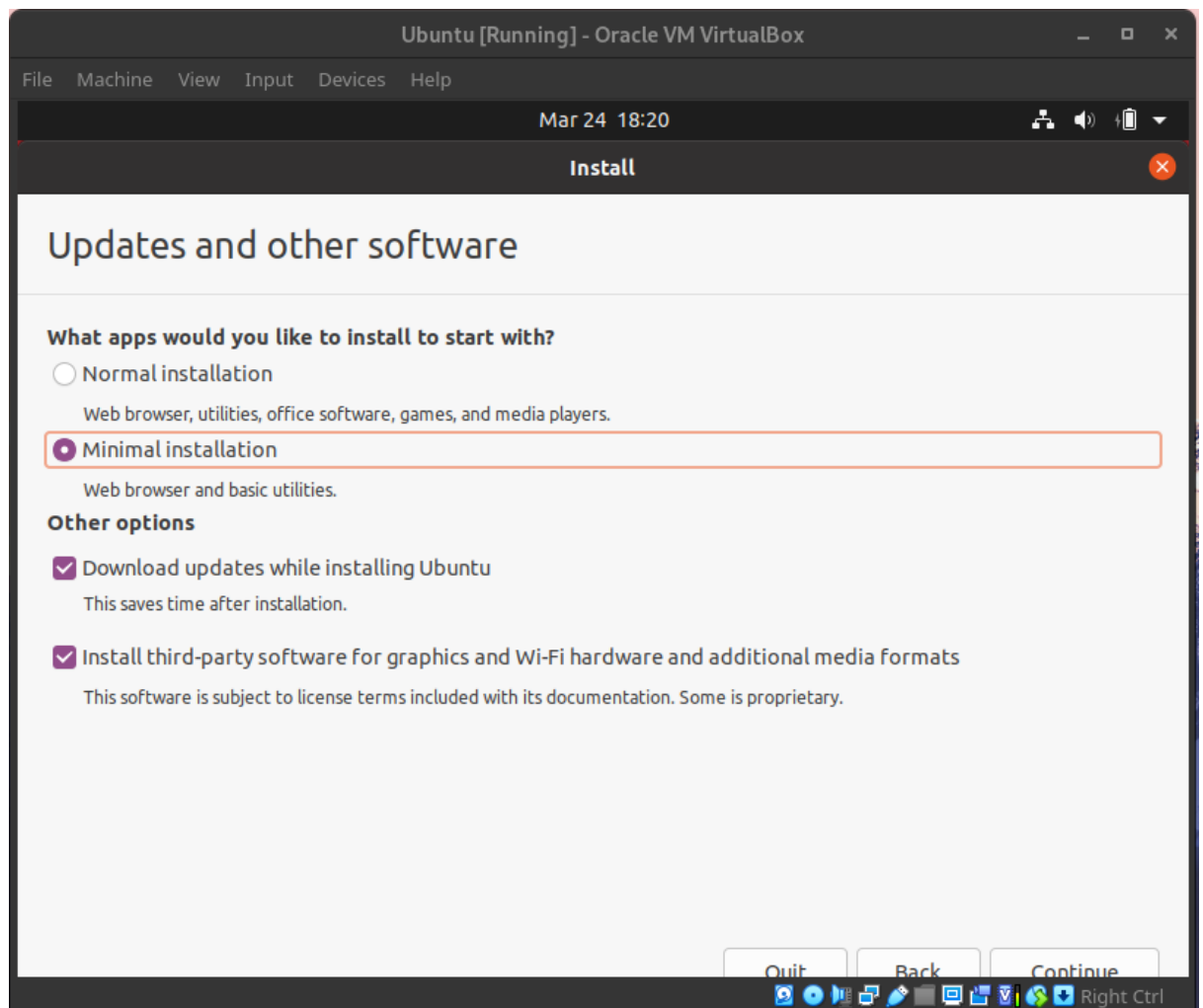
2. Скачиваем образ Ubuntu с официального сайта (<https://ubuntu.com>)
3. В VirtualBox создаём новую виртуальную машину и указываем путь до скачанного образа



4. Загружаемся в систему и приступаем к установке "Install Ubuntu"

5. Выбираем язык системы и раскладку клавиатуры

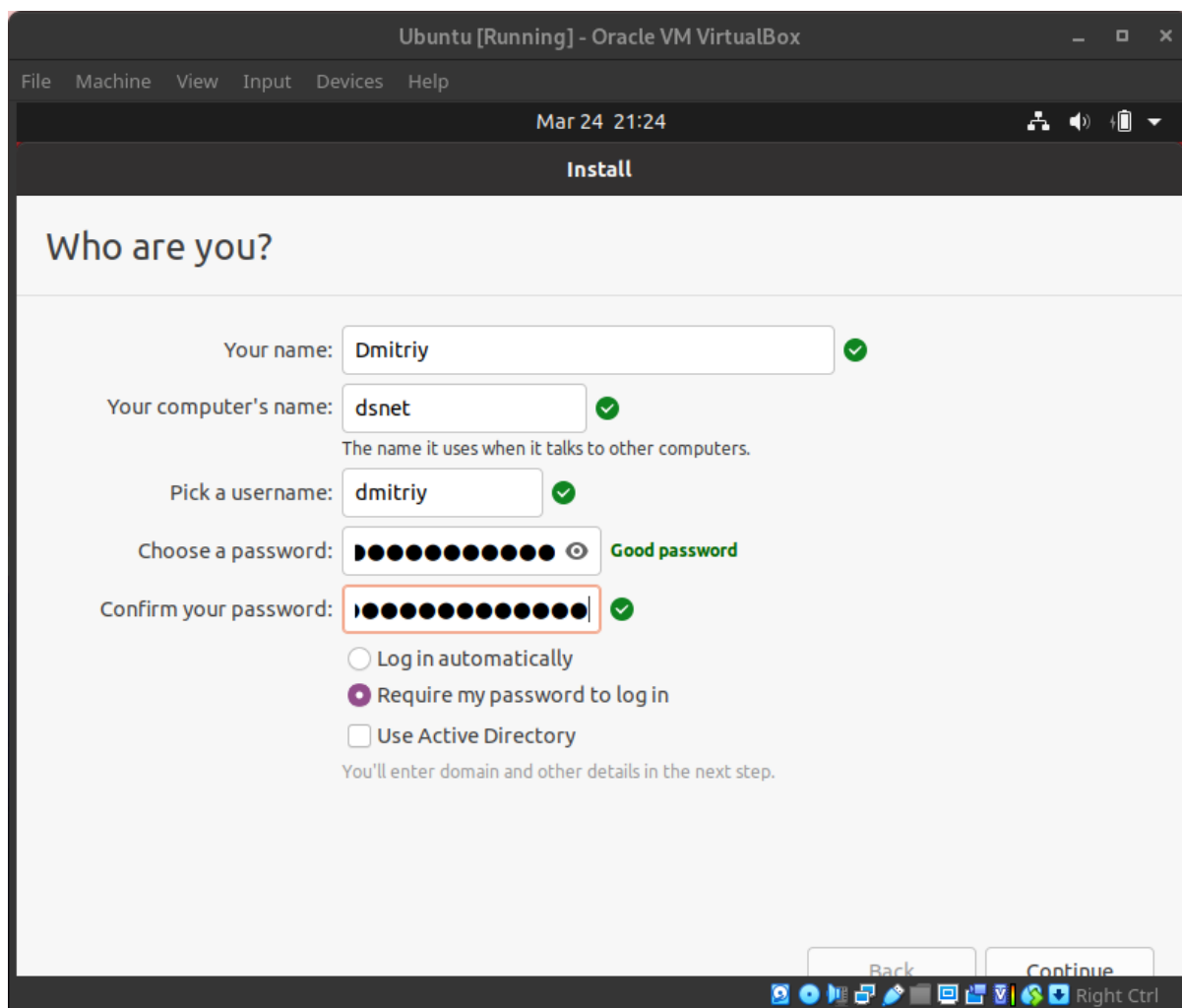
6. Выбираем конфигурацию системы



7. Назначаем диск, на который будет установлена Ubuntu

8. Выбираем часовой пояс

9. Создаём пользователя



После этого начнётся установка, в конце которой будет предложено перезапустить виртуальную машину.

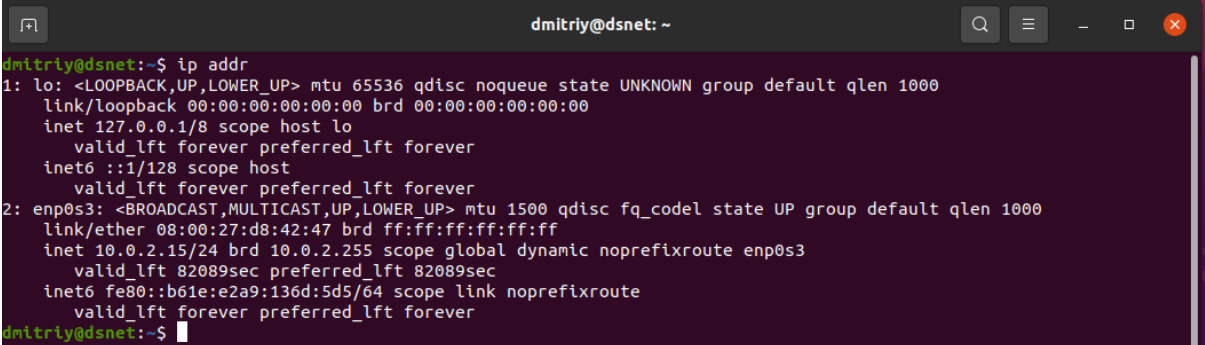
10. После перезагрузки проводим начальную конфигурацию системы (добавление аккаунтов, настройка параметров конфиденциальности)

11. Выполняем обновление пакетов командой

```
$ sudo apt-get update && sudo apt-get upgrade
```

Настройка SSH

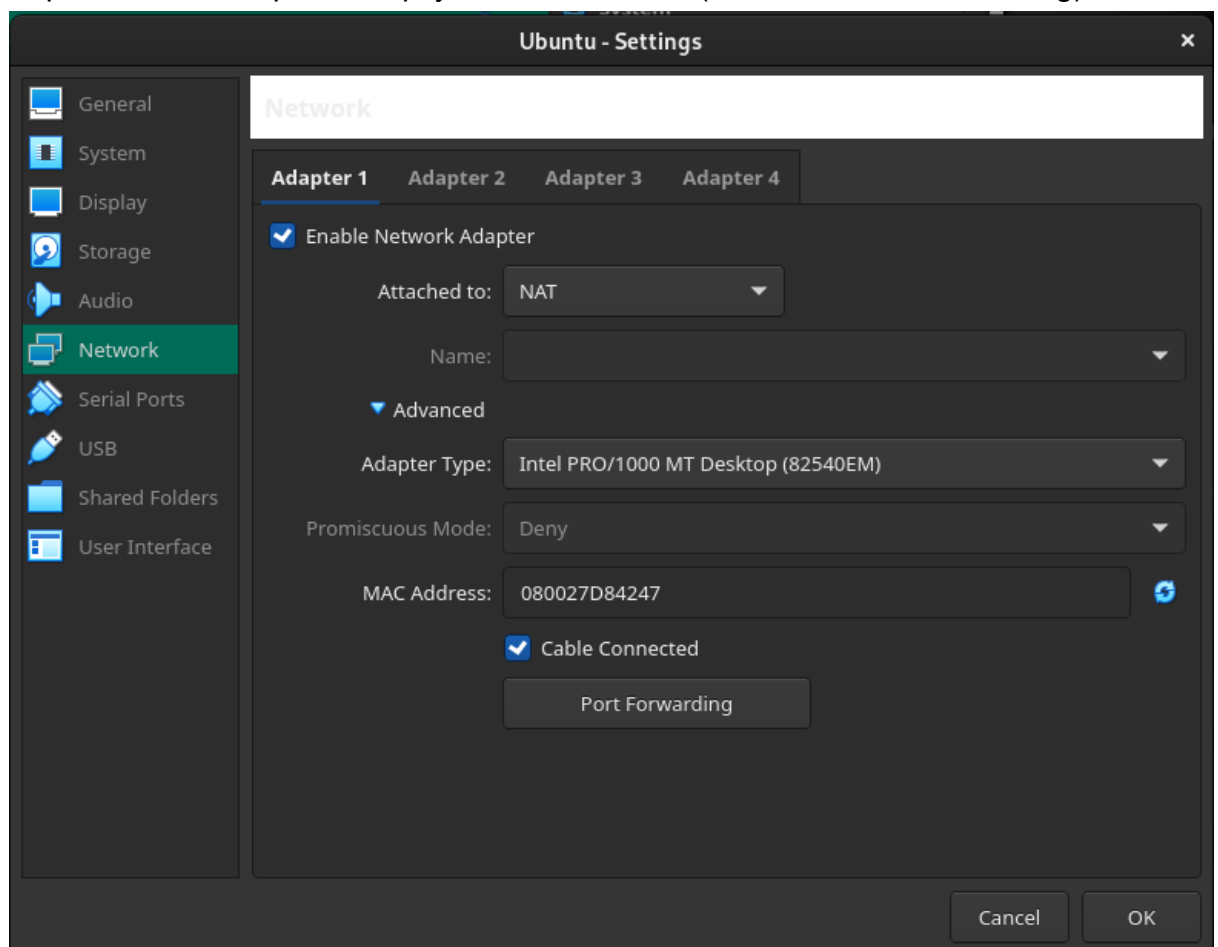
1. Вводим в терминале команду `ip addr` и узнаем `ip` адрес



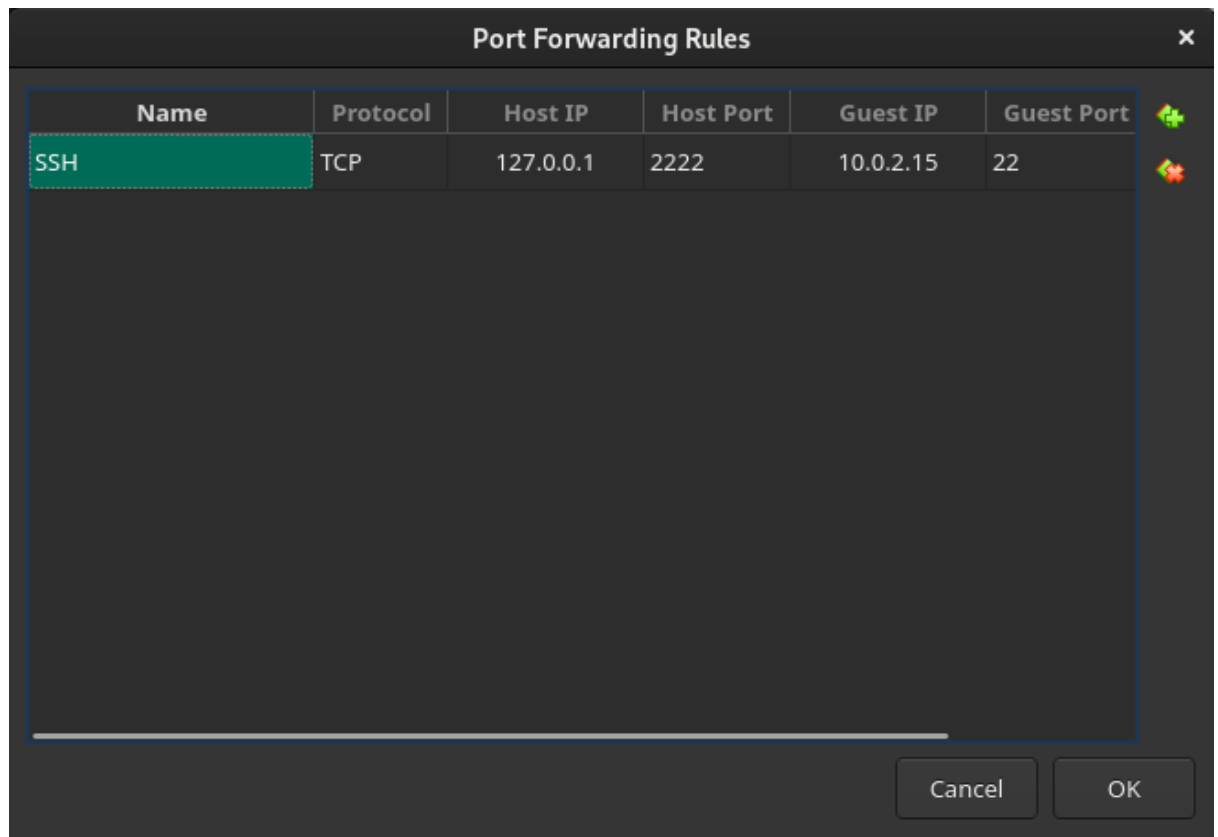
```
dmntriy@dsnet: ~  
dmntriy@dsnet:~$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:d8:42:47 brd ff:ff:ff:ff:ff:ff  
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3  
        valid_lft 82089sec preferred_lft 82089sec  
    inet6 fe80::b61e:e2a9:136d:5d5/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
dmntriy@dsnet:~$
```

В данном случае `ip` адрес 10.0.2.15

2. Выключаем виртуальную машину
3. Переходим в настройки виртуальной машины (Network- Port Forwarding)



4. Нажимаем на пиктограмму "+" и вводим данные:
Name - SSH
Protocol - TCP
Host IP - оставляем пустым или указываем 127.0.0.1
Host port - любой свободный порт, например 2222
Guest IP - адрес, который был получен в шаге № 1
Guest Port - порт SSH на VM (по умолчанию 22)



5. Сохраняем конфигурацию (OK)
6. Запускаем виртуальную машину и устанавливаем openssh-server командой

```
$ sudo apt install openssh-server
```

7. Проверяем работу службы SSH командой

```
$ service ssh status
```

```
dmিত্রি@dsnet:~$ service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e>
   Active: active (running) since Sat 2022-04-30 14:37:18 MSK; 3min 42s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 788 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 824 (sshd)
     Tasks: 1 (limit: 7039)
    Memory: 4.3M
     CGroup: /system.slice/ssh.service
            └─824 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
```

По умолчанию запущена после установки, в противном случае включаем командой

```
$ sudo systemctl status ssh
```

8. Для Ubuntu* настраиваем брандмауэр UFW командой (разрешаем доступ по SSH)

```
$ sudo ufw allow ssh
```

Проверить состояние SSH можно командой

```
$ sudo ufw status
```

9. Подключаемся к виртуальной машине командой

```
$ ssh user@localhost -p 2222
```

Где user - имя пользователя, 2222 - указанный в конфигурации порт. При авторизации указываем пароль пользователя.

10. Генерируем ssh-ключи в основной системе командой

```
$ ssh-keygen
```

11. По умолчанию ключи будут сгенерированы в директорию ~/.ssh
Проверяем корректность создания командой

```
$ ls ~/.ssh
```

Где id_rsa - приватный ключ, id_rsa.pub - публичный

12. Копируем сгенерированный публичный ключ на сервер (виртуальную машину) командой

```
$ ssh-copy-id user@localhost
```

User - имя пользователя, localhost - ip-адрес сервера

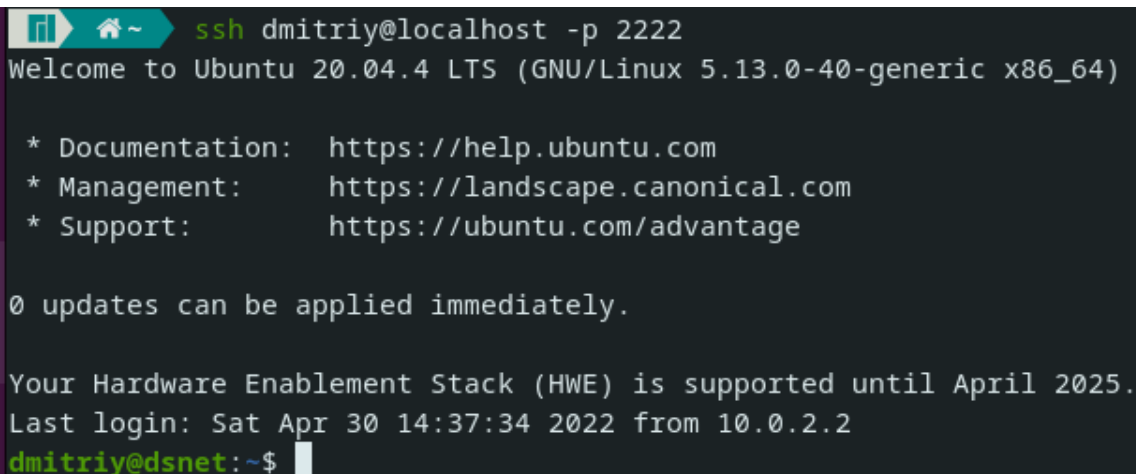
*В случае виртуальной машины необходимо указать порт

```
$ ssh-copy-id -p 2222 user@localhost
```

13. Проверяем корректность подключения командой

```
$ ssh user@localhost -p 2222
```

Подключение было выполнено успешно без ввода пароля пользователя



```
ssh dmitriy@localhost -p 2222
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.13.0-40-generic x86_64)

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

0 updates can be applied immediately.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Sat Apr 30 14:37:34 2022 from 10.0.2.2
dmitriy@dsnet:~$
```

Отключение авторизации по паролю

Для обеспечения безопасности отключим возможность подключения по ssh при помощи пароля

14. Открываем конфигурационный файл ssh командой

```
$ sudo vi /etc/ssh/sshd_config
```


15. Раскомментируем параметр "PasswordAuthentication" и изменим его значение с "yes" на "no"

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

16. Перезагрузим сервис ssh командой

```
$ sudo service ssh restart
```

Получение информации о сетевом оборудовании

Получим основную информацию о сетевом оборудовании при помощи следующих команд:

Команда `ip` - утилита, позволяющая настроить сетевые интерфейсы, `ip` адреса, таблицы маршрутизации, `arp` и `masc` таблицы.

`$ ip` - получение справки по команде

`$ ip addr` - Отображения списка всех сетевых интерфейсов и связанных с ними IP адресов (тип интерфейса - `local loop`(для связи внутри сети)(`lo`)/`ether`(обычное имя реального интерфейса), флаги работы - `qdisc`, `noqueue`, IPv4-адреса - `inet`, IPv6-адреса - `inet6`). Также команда может быть использована для добавления/удаления `ip` адресов (с модификаторами `add` и `del` соответственно)

```
dmitriy@dsnet:~$ ip addr  
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  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:d8:42:47 brd ff:ff:ff:ff:ff:ff  
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3  
        valid_lft 85585sec preferred_lft 85585sec  
    inet6 fe80::b61e:e2a9:136d:5d5/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever
```

`$ ip link` - просмотр всех сетевых интерфейсов, вывод аналогичен команде `ip addr`, только без соответствующих `ip` адресов. Может быть использована для отключения/включения интерфейса (модификатор `set`), настройки MTU (размера одного пакета), MAC-адресации и т.д.

```

dmitriy@dsnet:~$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
    link/ether 08:00:27:d8:42:47 brd ff:ff:ff:ff:ff:ff

```

\$ ip route - команда, выводящая таблицу маршрутизации, где default - вариант по умолчанию (ip адрес цели или маска подсети), via - через какой шлюз мы можем добраться до этой цели, dev - сетевой интерфейс, с помощью которого будет доступен этот шлюз, proto - информирует о том, кем был установлен маршрут (dhcp, kernel), metric - приоритет маршрута (чем меньше - тем выше приоритет), src - ip адрес, с которого отправляются пакеты.

```

dmitriy@dsnet:~$ ip route
default via 10.0.2.2 dev enp0s3 proto dhcp metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15 metric 100

```

Команда netstat - утилита, предоставляющая информацию о сетевых подключениях, таблицах arp, mas и маршрутизации, статистику интерфейсов и т.д.

\$ netstat -a - получить список всех портов (в т.ч. те, которые не являются прослушиваемыми)

```

dmitriy@dsnet:~$ netstat -a
Active Internet connections (servers and established)

```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	localhost:domain	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:ssh	0.0.0.0:*	LISTEN
tcp	0	0	localhost:ipp	0.0.0.0:*	LISTEN
tcp	0	0	dsnet:52290	32.121.122.34.bc.g:http	TIME_WAIT
tcp	0	0	dsnet:ssh	_gateway:50098	ESTABLISHED
tcp6	0	0	[::]:ssh	[::]:*	LISTEN
tcp6	0	0	ip6-localhost:ipp	[::]:*	LISTEN
udp	0	0	localhost:domain	0.0.0.0:*	
udp	0	0	dsnet:bootpc	_gateway:bootps	ESTABLISHED
udp	0	0	0.0.0.0:631	0.0.0.0:*	
udp	0	0	0.0.0.0:mdns	0.0.0.0:*	
udp	0	0	0.0.0.0:42254	0.0.0.0:*	
udp6	0	0	[::]:39596	[::]:*	
udp6	0	0	[::]:mdns	[::]:*	
raw6	0	0	[::]:ipv6-icmp	[::]:*	7

```

Active UNIX domain sockets (servers and established)

```

Proto	RefCnt	Flags	Type	State	I-Node	Path
unix	2	[ACC]	STREAM	LISTENING	20057	/run/uuid/request
unix	2	[ACC]	STREAM	LISTENING	29780	@/tmp/dbus-b7D366yY

Запрос можно выполнить для определённого протокола - tcp или udp (netstat -at и -au соответственно)

\$ netstat -r - получение таблицы маршрутизации

```
dmitriy@dsnet:~$ netstat -r
Kernel IP routing table
Destination      Gateway          Genmask         Flags   MSS Window  irtt Iface
default          _gateway        0.0.0.0         UG      0 0        0 enp0s3
10.0.2.0         0.0.0.0         255.255.255.0   U        0 0        0 enp0s3
link-local       0.0.0.0         255.255.0.0     U        0 0        0 enp0s3
```

\$ netstat -s - вывод статистики для каждого протокола (общее количество пакетов определённого типа, число отклонённых/доставленных пакетов и т.д.)

```
dmitriy@dsnet:~$ netstat -s
Ip:
  Forwarding: 2
  13520 total packets received
  1 with invalid addresses
  0 forwarded
  0 incoming packets discarded
  13517 incoming packets delivered
  9090 requests sent out
  20 outgoing packets dropped
Icmp:
  40 ICMP messages received
  0 input ICMP message failed
  ICMP input histogram:
    destination unreachable: 40
  40 ICMP messages sent
  0 ICMP messages failed
  ICMP output histogram:
    destination unreachable: 40
```

\$ netstat -i - вывод информации о сетевых интерфейсах (RX-OK и TX-OK колонки показывают, сколько пакетов были получены или переданы без ошибок, повредились (RX-ERR/TXERR), сколько было потеряно (RX-DRP/TX-DRP), и сколько было потеряно из-за переполнения (RX-OVR/TX-OVR)). Флаги на скриншоте: B - широковещательный адрес, M - интерфейс получает все пакеты, L - loopback устройство, R - Интерфейс работает, U - Интерфейс активен.

```
dmitriy@dsnet:~$ netstat -i
Kernel Interface table
Iface    MTU     RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
enp0s3   1500    15327     0     0 0        9323     0     0     0 BMRU
lo       65536    429     0     0 0         429     0     0     0 LRU
```

Команда может быть расширена (netstat -ie):

```
dmitriy@dsnet:~$ netstat -ie
Kernel Interface table
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::b61e:e2a9:136d:5d5 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:d8:42:47 txqueuelen 1000 (Ethernet)
    RX packets 15772 bytes 19526409 (19.5 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 9704 bytes 1294253 (1.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 446 bytes 39672 (39.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 446 bytes 39672 (39.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Заключение

В ходе выполнения работы была установлена OS Linux на виртуальную машину, настроено безопасное подключение по SSH, а также получена основная информация о сетевом оборудовании.