Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Факультет инфокоммуникационных технологий

Направление подготовки 11.03.02

Лабораторная работа №4

Элементы безопасности в Linux

Выполнил:

Дощенников Никита Андреевич

Группа: К3221

Проверил:

Береснев Артем Дмитриевич

Санкт-Петербург

2025

Цель работы:
Получить практические навыки работы с сетевой подсистемой в Linux, научится управлять пользователями, правами на файлы и каталоги, научиться настраивать сетевые интерфейсы, NAT и настраивать ssh.

Часть 1. Подготовка конфигурации.
У себя на машине я создал новую виртуальную сеть:

```
virsh net-define /tmp/intnet.xml
virsh net-autostart intnet
virsh net-start intnet
```

Содержание файла `/tmp/intnet.xml`:

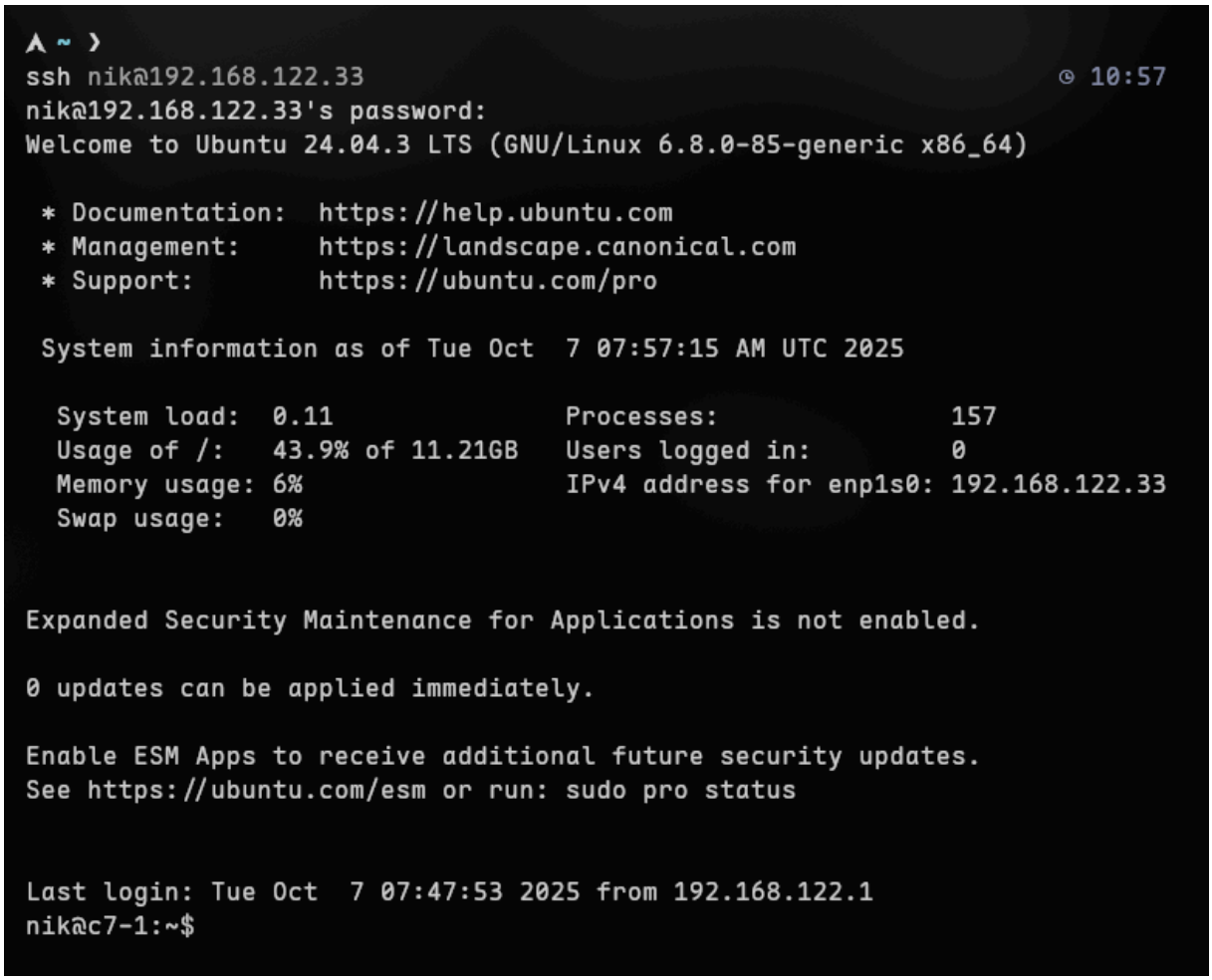
```
<network>
  <name>intnet</name>
  <forward mode='none'/>
</network>
```

К первой машине я добавил дополнительный интерфейс, подключенный к сети intnet:

```
virsh attach-interface --domain ubuntu24.04 --type network --
source intnet --model virtio --config --live
```

Вторую машину я полностью перевел во внутреннюю сеть:

```
virsh detach-interface --domain ubuntu24.04-clone --type
network --mac 52:54:00:76:1f:db --config --live
virsh attach-interface --domain ubuntu24.04-clone --type
network --source intnet --model virtio --config --live
```


~ ❯
ssh nik@192.168.122.33
10:57
nik@192.168.122.33's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
System information as of Tue Oct 7 07:57:15 AM UTC 2025
System load: 0.11
Usage of /: 43.9% of 11.21GB
Memory usage: 6%
Swap usage: 0%
Processes: 157
Users logged in: 0
IPv4 address for enp1s0: 192.168.122.33
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
Last login: Tue Oct 7 07:47:53 2025 from 192.168.122.1
nik@c7-1:~$

```
nik@c7-1:~$ ssh nik@10.0.0.2
nik@10.0.0.2's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)

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

 System information as of Tue Oct  7 07:58:07 AM UTC 2025

  System load: 0.0               Memory usage: 6%   Processes:       162
  Usage of /:  44.4% of 11.21GB  Swap usage:   0%   Users logged in: 1


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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


Last login: Tue Oct  7 07:48:02 2025 from 10.0.0.1
nik@c7-2:~$
```

На `c7-1`:

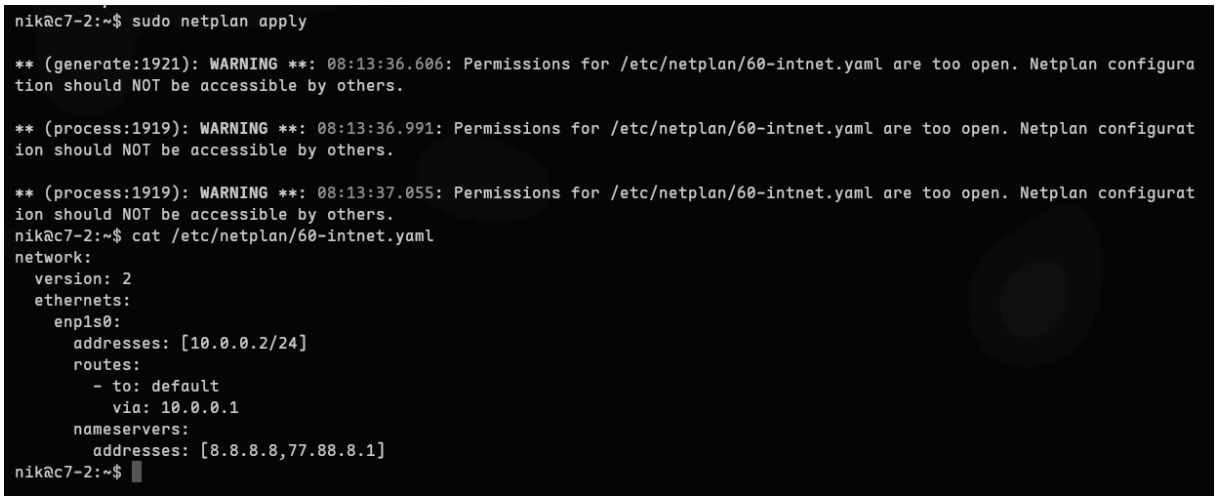
```
nik@c7-1:~$ sudo nvim /etc/netplan/60-intnet.yaml
nik@c7-1:~$ sudo netplan apply

** (generate:2292): WARNING **: 08:07:57.285: Permissions for /etc/netplan/60-intnet.yaml are too open. Netplan configura
tion should NOT be accessible by others.

** (process:2290): WARNING **: 08:07:57.598: Permissions for /etc/netplan/60-intnet.yaml are too open. Netplan configurat
ion should NOT be accessible by others.

** (process:2290): WARNING **: 08:07:57.682: Permissions for /etc/netplan/60-intnet.yaml are too open. Netplan configurat
ion should NOT be accessible by others.
nik@c7-1:~$ cat /etc/netplan/60-intnet.yaml
network:
  version: 2
  ethernets:
    enp8s0:
      addresses: [10.0.0.1/24]
nik@c7-1:~$
```

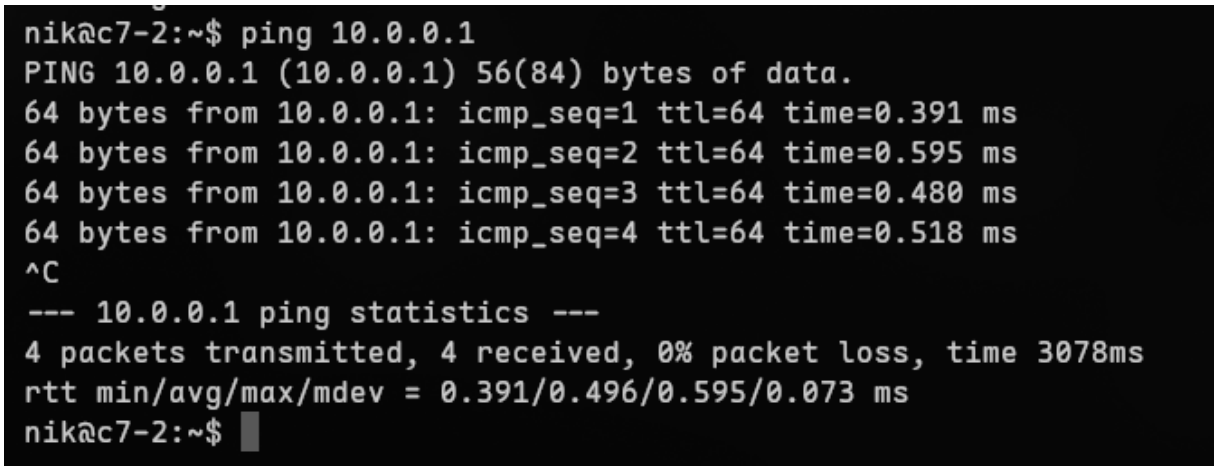
```
nik@c7-1:~$ ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.401 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.403 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.388 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.513 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.567 ms
^C
--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4122ms
rtt min/avg/max/mdev = 0.388/0.454/0.567/0.072 ms
nik@c7-1:~$
```

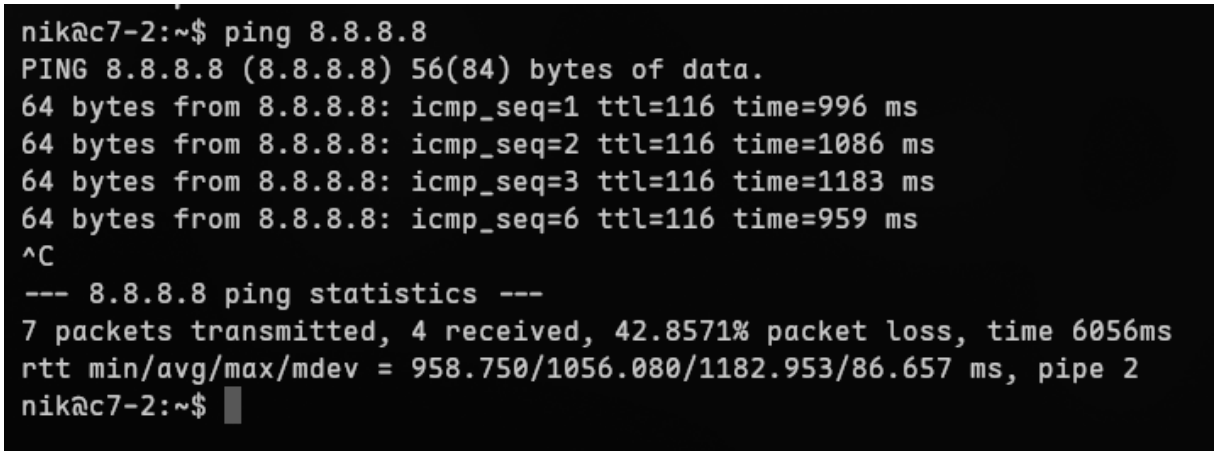
```
nik@c7-1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=704 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=117 time=830 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=117 time=1033 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=117 time=893 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=117 time=1023 ms
^C
--- 8.8.8.8 ping statistics ---
12 packets transmitted, 5 received, 58.3333% packet loss, time 11172ms
rtt min/avg/max/mdev = 703.785/896.690/1033.467/123.649 ms, pipe 2
nik@c7-1:~$
```

```
nik@c7-1:~$ sudo nvim /etc/sysctl.conf
nik@c7-1:~$ cat /etc/sysctl.conf | grep net.ipv4.ip_forward=1
net.ipv4.ip_forward=1
nik@c7-1:~$ sudo sysctl -p
net.ipv4.ip_forward = 1
nik@c7-1:~$ sudo iptables -t nat -A POSTROUTING -o enp1s0 -j MASQUERADE
nik@c7-1:~$
```

На `c7-2`:







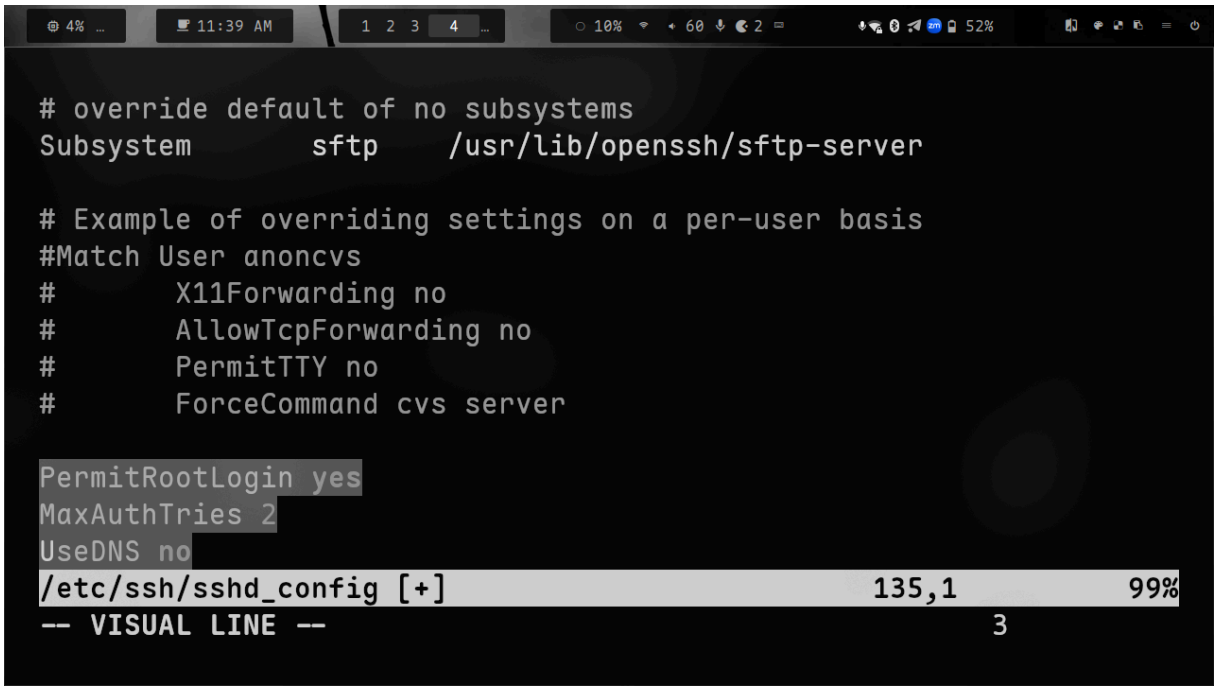
Часть 2. Создание пользователей и настройка OpenSSH Server (sshd).

Я создал пользователя на каждой машине:

```
nik@c7-1:~$ sudo adduser nik-c7-1
info: Adding user `nik-c7-1' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `nik-c7-1' (1001) ...
info: Adding new user `nik-c7-1' (1001) with group `nik-c7-1 (1001)' ...
info: Creating home directory `/home/nik-c7-1' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for nik-c7-1
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 `nik-c7-1' to supplemental / extra groups `users' ...
info: Adding user `nik-c7-1' to group `users' ...
nik@c7-1:~$ id nik-c7-1
uid=1001(nik-c7-1) gid=1001(nik-c7-1) groups=1001(nik-c7-1),100(users)
nik@c7-1:~$
```

```
nik@c7-2:~$ sudo adduser nik-c7-2
[sudo] password for nik:
info: Adding user `nik-c7-2' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `nik-c7-2' (1001) ...
info: Adding new user `nik-c7-2' (1001) with group `nik-c7-2 (1001)' ...
info: Creating home directory `/home/nik-c7-2' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for nik-c7-2
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 `nik-c7-2' to supplemental / extra groups `users' ...
info: Adding user `nik-c7-2' to group `users' ...
nik@c7-2:~$ id nik-c7-2
uid=1001(nik-c7-2) gid=1001(nik-c7-2) groups=1001(nik-c7-2),100(users)
nik@c7-2:~$
```

Затем я настроил `ssh` на обеих машинах:

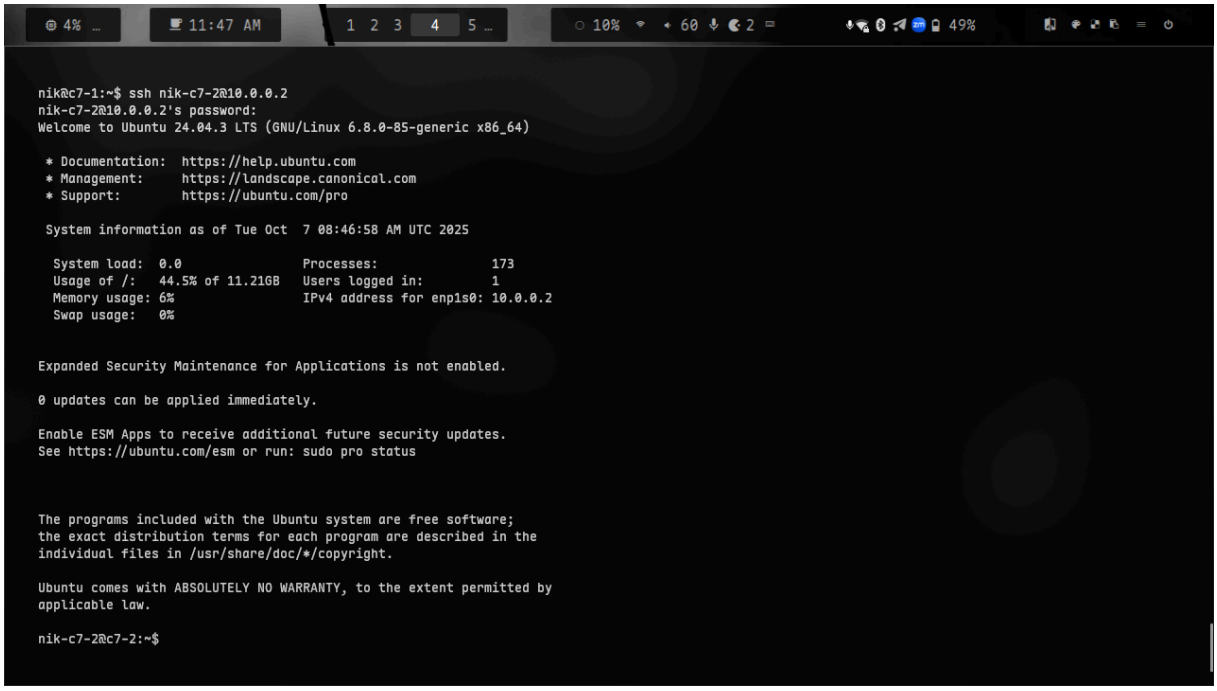


И перезапустил daemon:

```
nik@c7-1:~$ sudo nvim /etc/ssh/sshd_config
nik@c7-1:~$ sudo systemctl restart ssh
nik@c7-1:~$ sudo systemctl status ssh
WARNING: terminal is not fully functional
Press RETURN to continue
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled;>
     Active: active (running) since Tue 2025-10-07 08:43:45 UTC; 8>
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 2552 ExecStartPre=/usr/sbin/sshd -t (code=exited, sta>
   Main PID: 2554 (sshd)
      Tasks: 1 (limit: 4605)
     Memory: 1.2M (peak: 1.4M)
        CPU: 20ms
     CGroup: /system.slice/ssh.service
             └─2554 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-10>

Oct 07 08:43:45 c7-1 systemd[1]: Starting ssh.service - OpenBSD Se>
Oct 07 08:43:45 c7-1 sshd[2554]: Server listening on 0.0.0.0 port >
Oct 07 08:43:45 c7-1 sshd[2554]: Server listening on :: port 22.
Oct 07 08:43:45 c7-1 systemd[1]: Started ssh.service - OpenBSD Sec>
nik@c7-1:~$
```

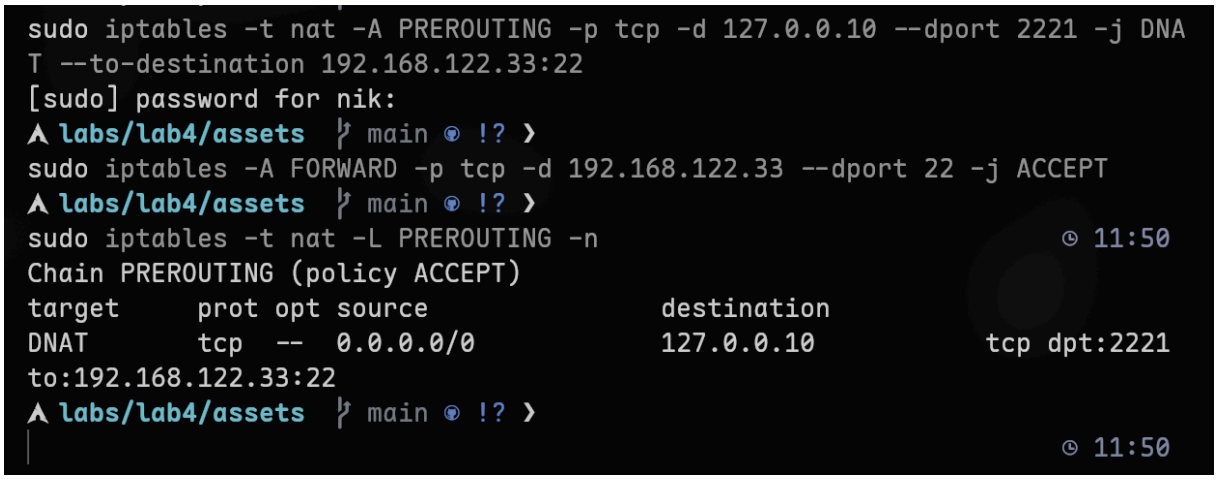
Все те же действия я проделал на второй машине.

Попробовал подключиться с `c7-1` на `c7-2`:



Часть 3. Подключение к виртуальной машине c7-1 по ssh через NAT VirtualBox.

У себя на машине:



```
ssh nik@192.168.122.33
nik@192.168.122.33's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)

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

 System information as of Tue Oct  7 09:58:13 AM UTC 2025

  System load:  1.72               Processes:               166
  Usage of /:   44.0% of 11.21GB   Users logged in:         0
  Memory usage: 5%                 IPv4 address for enp1s0: 192.168.122.33
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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


Last login: Tue Oct  7 07:57:15 2025 from 192.168.122.1
nik@c7-1:~$ ls
nik@c7-1:~$
```

```
4% …  01:00 PM  1 2 3 4 5 …  10%  60  2  18%

ssh nik-c7-1@192.168.122.33                                          12:59
nik-c7-1@192.168.122.33's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)

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

 System information as of Tue Oct  7 09:59:53 AM UTC 2025

  System load:  0.32              Processes:              165
  Usage of /:   44.1% of 11.21GB  Users logged in:        0
  Memory usage: 6%                IPv4 address for enp1s0: 192.168.122.33
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

nik-c7-1@c7-1:~$
```

Копирование файла с `c7-2` на `c7-1`:

Копирование с `c7-2` на `c7-1`:

```
nik@c7-1:~$ scp nik@10.0.0.2:/home/nik/part.sh /home/nik/
nik@10.0.0.2's password:
part.sh
nik@c7-1:~$ cat part.sh
#!/bin/bash

for disk in /dev/sd{a,b,c,d,e}; do
  sudo parted -s $disk mklabel gpt
  sudo parted -s $disk mkpart primary 0% 100%
done

nik@c7-1:~$ ssh nik@10.0.0.2
nik@10.0.0.2's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)

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

 System information as of Tue Oct  7 10:12:11 AM UTC 2025

  System load:  0.1               Processes:               165
  Usage of /:   44.5% of 11.21GB  Users logged in:         1
  Memory usage: 6%                IPv4 address for enp1s0: 10.0.0.2
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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


Last login: Tue Oct  7 10:09:10 2025 from 10.0.0.1
nik@c7-2:~$ ls
info.sh  part.sh  test.txt
nik@c7-2:~$ cat part.sh
#!/bin/bash

for disk in /dev/sd{a,b,c,d,e}; do
  sudo parted -s $disk mklabel gpt
  sudo parted -s $disk mkpart primary 0% 100%
done

nik@c7-2:~$
```

Часть 4. Установка и настройка NAT в iptables.

Я установил `iptables`:

```
nik@c7-1:~$ sudo apt install -y iptables
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
iptables is already the newest version (1.8.10-3ubuntu2).
iptables set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
nik@c7-1:~$
```

Настроил клиентский NAT, то есть настроил связь между `10.0.0.0/24` и `enp1s0` через роутер `c7-1`.

```
nik@c7-1:~$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
nik@c7-1:~$ sudo iptables -t nat -A POSTROUTING -o enp1s0 -s 10.0.0.0/24 -j MASQUERADE
nik@c7-1:~$
```

Затем я разрешил форвардинг трафика на `c7-1`:

```
nik@c7-1:~$
nik@c7-1:~$ sudo iptables -A FORWARD -d 10.0.0.0/24 -j ACCEPT
nik@c7-1:~$ sudo iptables -A FORWARD -s 10.0.0.0/24 -j ACCEPT
nik@c7-1:~$
```

После этого я проверил работу интернета с `c7-2`:

```
nik@c7-2:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=9 ttl=116 time=870 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=116 time=867 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=116 time=688 ms
64 bytes from 8.8.8.8: icmp_seq=15 ttl=116 time=500 ms
^C
--- 8.8.8.8 ping statistics ---
15 packets transmitted, 4 received, 73.3333% packet loss, time 14288ms
rtt min/avg/max/mdev = 499.626/731.227/869.839/152.669 ms
nik@c7-2:~$
```

Я настроил проброс порта:

```
nik@c7-1:~$ sudo iptables -t nat -A PREROUTING -i enp1s0 -p tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
nik@c7-1:~$
```

На `c7-1`:

```
nik@c7-1:~$ sudo iptables -t nat -A PREROUTING -i enp1s0 -p tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
[sudo] password for nik:
nik@c7-1:~$
```

Локально:

```
~ ❯
ssh -p 55022 nik@192.168.122.33
nik@192.168.122.33's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-85-generic x86_64)

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

 System information as of Tue Oct  7 10:55:42 AM UTC 2025

  System load:  0.0                Processes:               166
  Usage of /:   44.5% of 11.21GB   Users logged in:         1
  Memory usage: 6%                 IPv4 address for enp1s0: 10.0.0.2
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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


Last login: Tue Oct  7 10:51:01 2025 from 192.168.122.1
nik@c7-2:~$
```

Текущие правила:

```
nik@c7-1:~$
nik@c7-1:~$ sudo iptables -L -v -n
[sudo] password for nik:
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
   96 28473 ACCEPT     0    --  *      *       0.0.0.0/0            10.0.0.0/24
  137 20835 ACCEPT     0    --  *      *       10.0.0.0/24          0.0.0.0/0
    0     0 ACCEPT     0    --  enp8s0 enp1s0  10.0.0.0/24          0.0.0.0/0
    0     0 ACCEPT     0    --  enp1s0 enp8s0  0.0.0.0/0            10.0.0.0/24          state RELATED,ESTABLISHED

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
nik@c7-1:~$ sudo iptables -t nat -L -v -n
Chain PREROUTING (policy ACCEPT 43 packets, 10193 bytes)
 pkts bytes target     prot opt in     out     source               destination
    2   120 DNAT       6    --  enp1s0 *       0.0.0.0/0            0.0.0.0/0            tcp dpt:55022 to:10.0.0.2:22
    0     0 DNAT       6    --  enp1s0 *       0.0.0.0/0            0.0.0.0/0            tcp dpt:55022 to:10.0.0.2:22
    0     0 DNAT       6    --  enp1s0 *       0.0.0.0/0            0.0.0.0/0            tcp dpt:55022 to:10.0.0.2:22

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain POSTROUTING (policy ACCEPT 34 packets, 2764 bytes)
 pkts bytes target     prot opt in     out     source               destination
   15  1074 MASQUERADE  0    --  *      enp1s0  10.0.0.0/24         0.0.0.0/0
nik@c7-1:~$
```

Настройки не сбрасываются после перезагрузки:

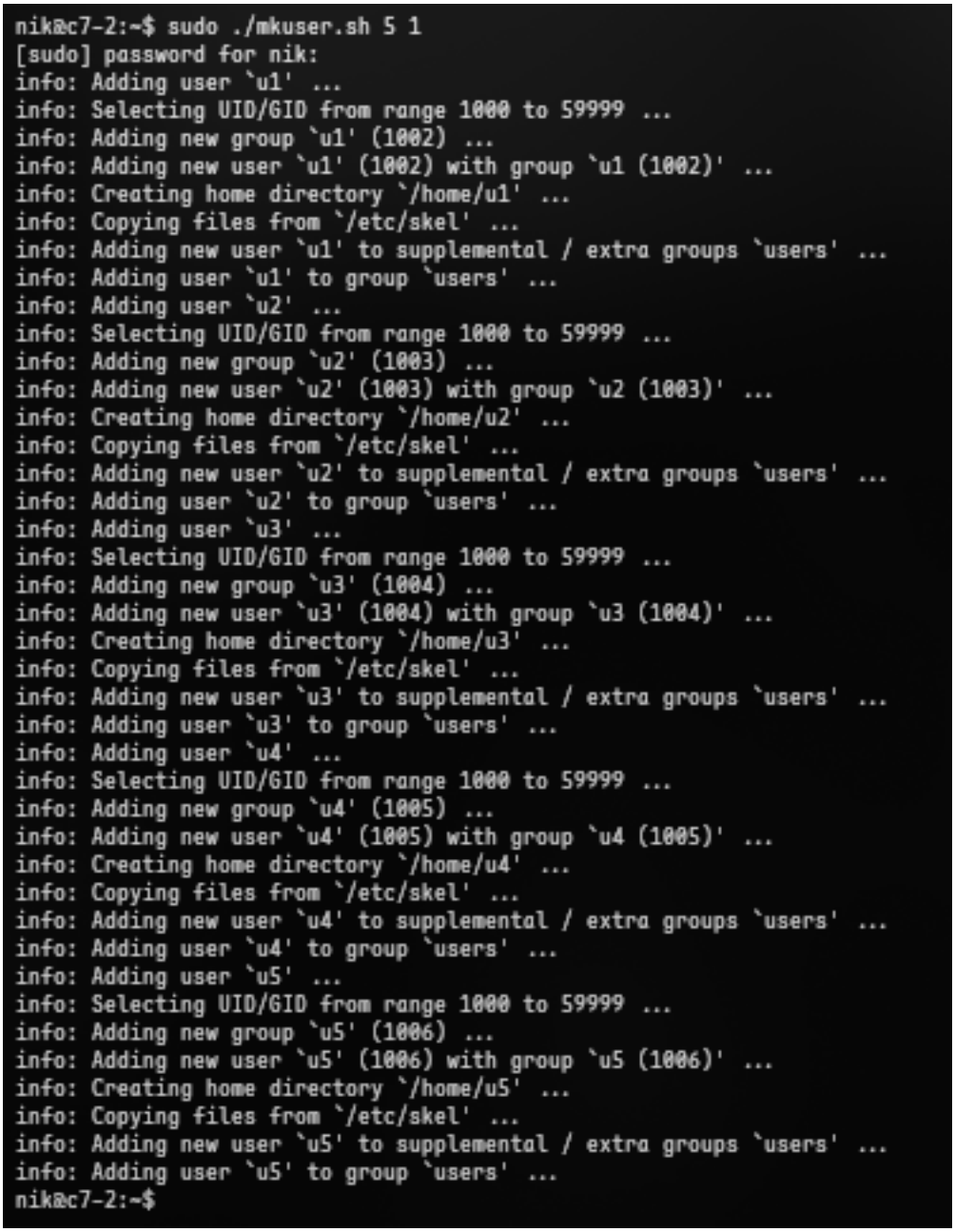
```
nik@c7-1:~$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Oct  7 10:59:54 2025
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A FORWARD -d 10.0.0.0/24 -j ACCEPT
-A FORWARD -s 10.0.0.0/24 -j ACCEPT
-A FORWARD -s 10.0.0.0/24 -i enp8s0 -o enp1s0 -j ACCEPT
-A FORWARD -d 10.0.0.0/24 -i enp1s0 -o enp8s0 -m state --state RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Tue Oct  7 10:59:54 2025
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Oct  7 10:59:54 2025
*nat
:PREROUTING ACCEPT [45:10572]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [43:3419]
-A PREROUTING -i enp1s0 -p tcp -m tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
-A PREROUTING -i enp1s0 -p tcp -m tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
-A PREROUTING -i enp1s0 -p tcp -m tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
-A POSTROUTING -s 10.0.0.0/24 -o enp1s0 -j MASQUERADE
COMMIT
# Completed on Tue Oct  7 10:59:54 2025
nik@c7-1:~$
```

Часть 5. Настройка прав на файлы и каталоги.

Я создал скрипт `mkuser.sh`:

```
#!/usr/bin/env bash
set -euo pipefail
```

```
count="${1:?}"
start="${2:?}"
for ((i=0;i<count;i++)); do
  n=$((start+i))
  u="u${n}"
  p="DerParol${n}"
  id -u "$u" >/dev/null 2>&1 || adduser --disabled-password --
gecos "" "$u"
  echo "${u}:${p}" | chpasswd
done
```



```
nik@c7-2:~$ getent passwd | grep '^u[1-5]:'
u1:x:1002:1002:,,,:/home/u1:/bin/bash
u2:x:1003:1003:,,,:/home/u2:/bin/bash
u3:x:1004:1004:,,,:/home/u3:/bin/bash
u4:x:1005:1005:,,,:/home/u4:/bin/bash
u5:x:1006:1006:,,,:/home/u5:/bin/bash
nik@c7-2:~$
```

Затем я создал группу:

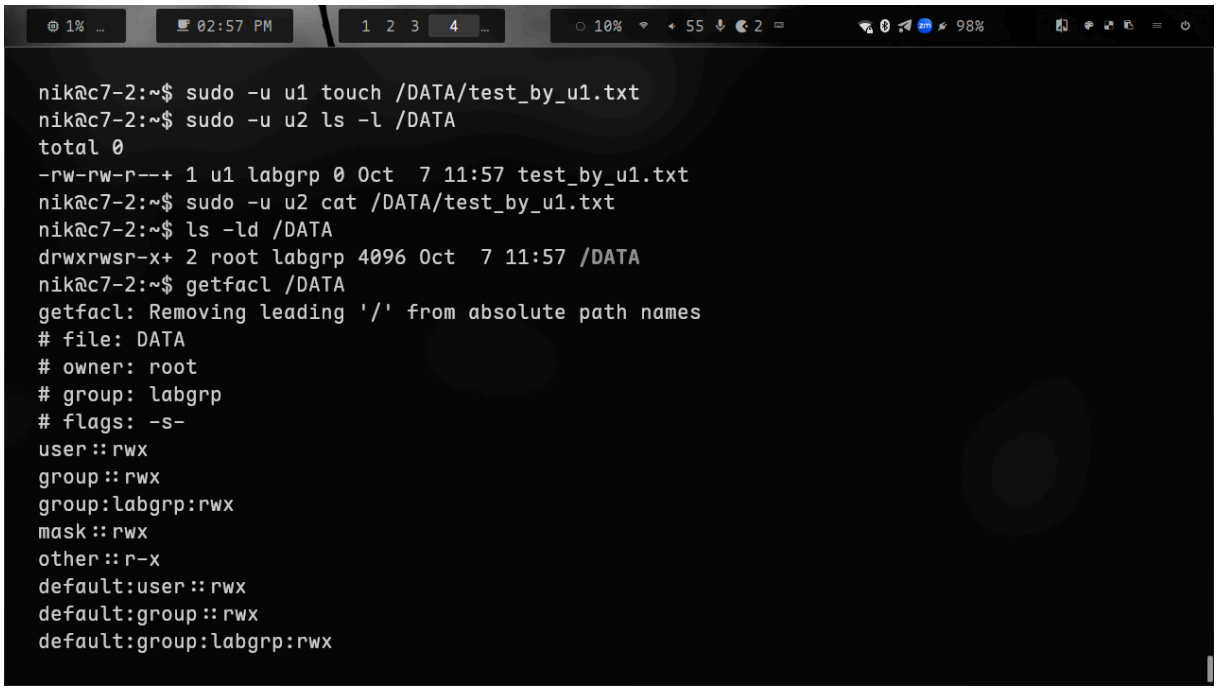
```
nik@c7-2:~$
nik@c7-2:~$ sudo groupadd labgrp
nik@c7-2:~$ getent group labgrp
labgrp:x:1007:
nik@c7-2:~$
```

Членам группы я выдал полный доступ, а остальным только чтение:

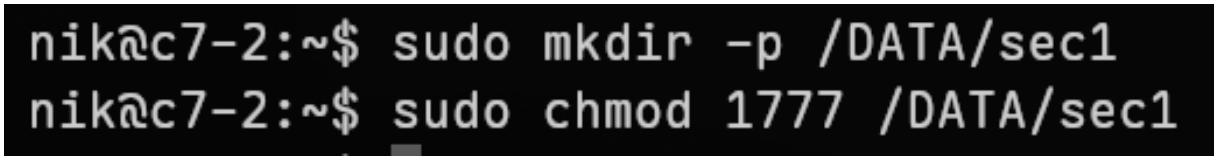
```
nik@c7-2:~$ sudo mkdir -p /DATA
nik@c7-2:~$ sudo chown root:labgrp /DATA
nik@c7-2:~$ sudo chmod 2775 /DATA
```

```
nik@c7-2:~$ sudo setfacl -m g:labgrp:rwx,o:rx /DATA
[sudo] password for nik:
nik@c7-2:~$ sudo setfacl -d -m g:labgrp:rwx,o:rx /DATA
nik@c7-2:~$ sudo usermod -aG labgrp u1
```

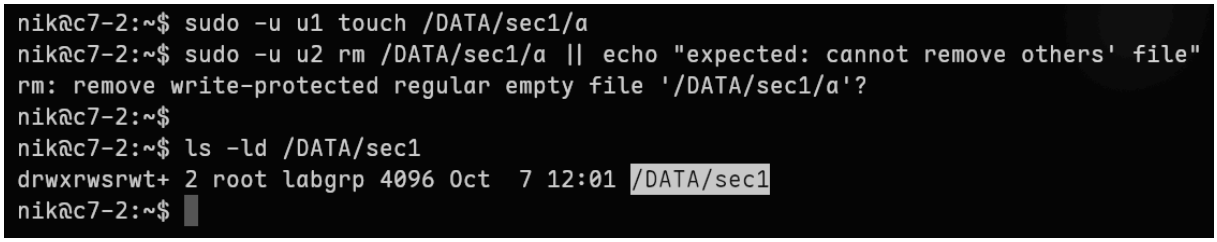
Проверим:



Я сделал так, чтобы в `/DATA/sec1` любой мог писать, но удалять - только свои файлы:



Проверим:



`/DATA/sec2` настроен на полный доступ для спец. пользователя, для пользователей `uN` только чтение, для прочих нельзя.

```
nik@c7-2:~$ id nik-c7-2
uid=1001(nik-c7-2) gid=1001(nik-c7-2) groups=1001(nik-c7-2),100(users)
nik@c7-2:~$ SPEC_USER="nik-c7-2"
nik@c7-2:~$ sudo mkdir -p /DATA/sec2
nik@c7-2:~$ sudo groupadd -f sec2readers
nik@c7-2:~$ sudo chgrp sec2readers /DATA/sec2
nik@c7-2:~$ sudo usermod -aG sec2readers u1
nik@c7-2:~$ sudo usermod -aG sec2readers u2
nik@c7-2:~$ sudo usermod -aG sec2readers u3
nik@c7-2:~$ sudo usermod -aG sec2readers u4
nik@c7-2:~$ sudo usermod -aG sec2readers u5
nik@c7-2:~$ sudo chown "${SPEC_USER}":sec2readers /DATA/sec2
nik@c7-2:~$ sudo chmod 2750 /DATA/sec2
nik@c7-2:~$ sudo setfacl -m u:"${SPEC_USER}":rwx,g:sec2readers:rx,o:--- /DATA/sec2
nik@c7-2:~$ sudo setfacl -d -m u:"${SPEC_USER}":rwx,g:sec2readers:r--,o:--- /DATA/sec2
nik@c7-2:~$
```

Проверим:

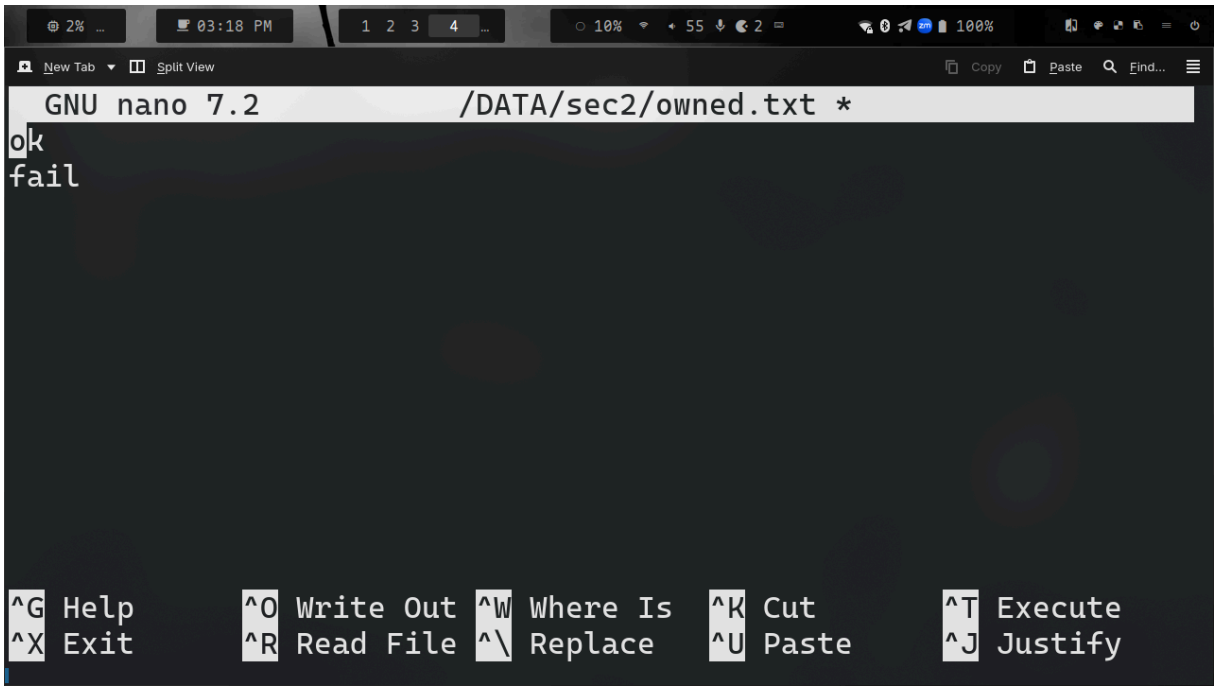
```
nik@c7-2:~$ sudo -u "${SPEC_USER}" bash -lc 'echo ok > /DATA/sec2/owned.txt'
nik@c7-2:~$ sudo -u u1 cat /DATA/sec2/owned.txt
ok
<ATA/sec2/owned.txt' || echo "expected: read-only for group"
nik@c7-2:~$ sudo -u u1 ls /DATA/sec2 >/dev/null || echo "should list: group has x"
nik@c7-2:~$ sudo -u nobody ls /DATA/sec2 || echo "expected: permission denied for others"
ls: cannot open directory '/DATA/sec2': Permission denied
expected: permission denied for others
nik@c7-2:~$
```

В `/DATA/sec3` скопировал `nano` и любой пользователь смог изменять с помощью его файлы в нем.

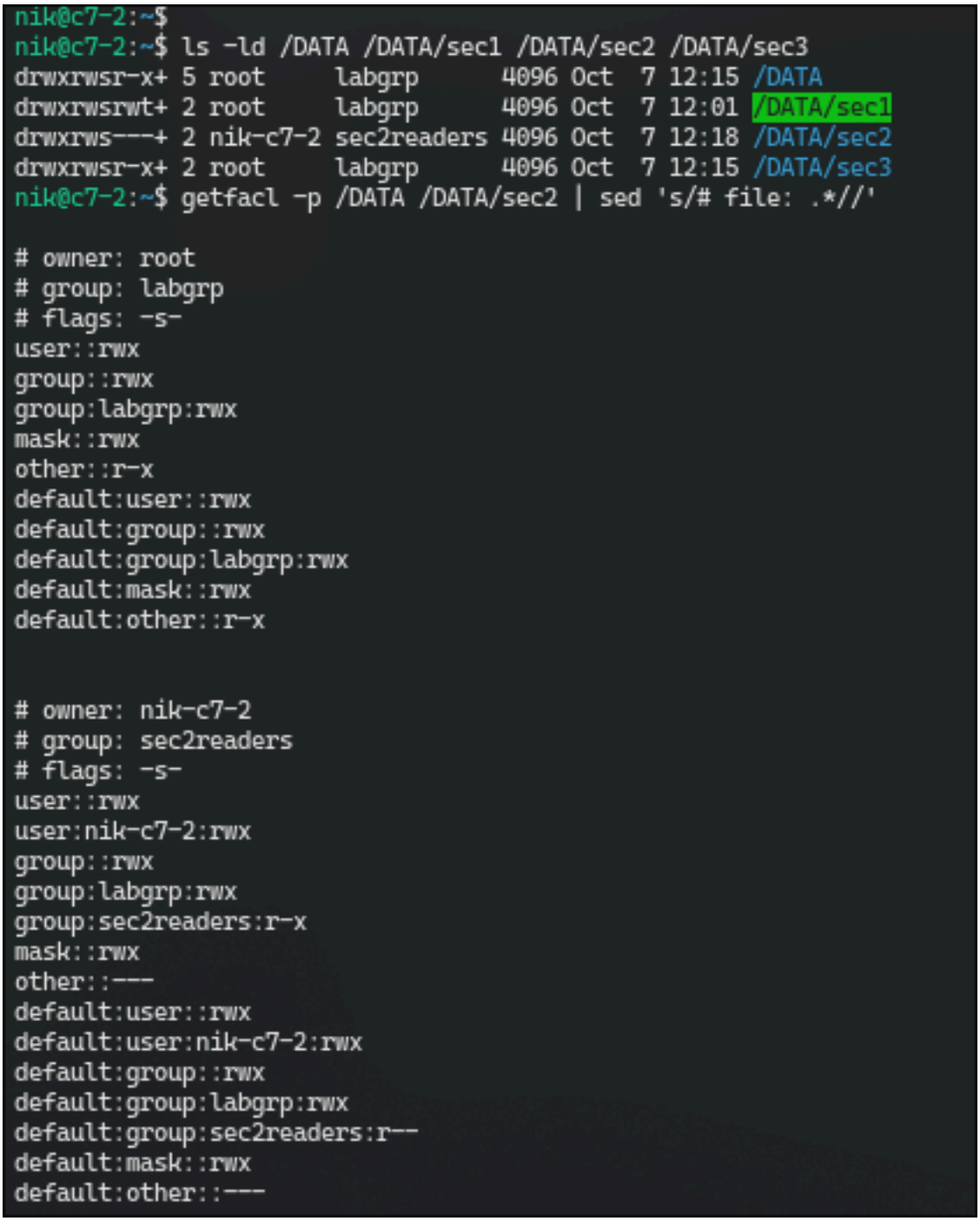
```
nik@c7-2:~$ sudo mkdir -p /DATA/sec3
nik@c7-2:~$ sudo cp /bin/nano /DATA/sec3/nano
nik@c7-2:~$ sudo chown root:root /DATA/sec3/nano
nik@c7-2:~$ sudo chmod 4755 /DATA/sec3/nano
```

Проверим:

```
nik@c7-2:~$ sudo -u u1 /DATA/sec3/nano /DATA/sec2/owned.txt
[sudo] password for nik:
```

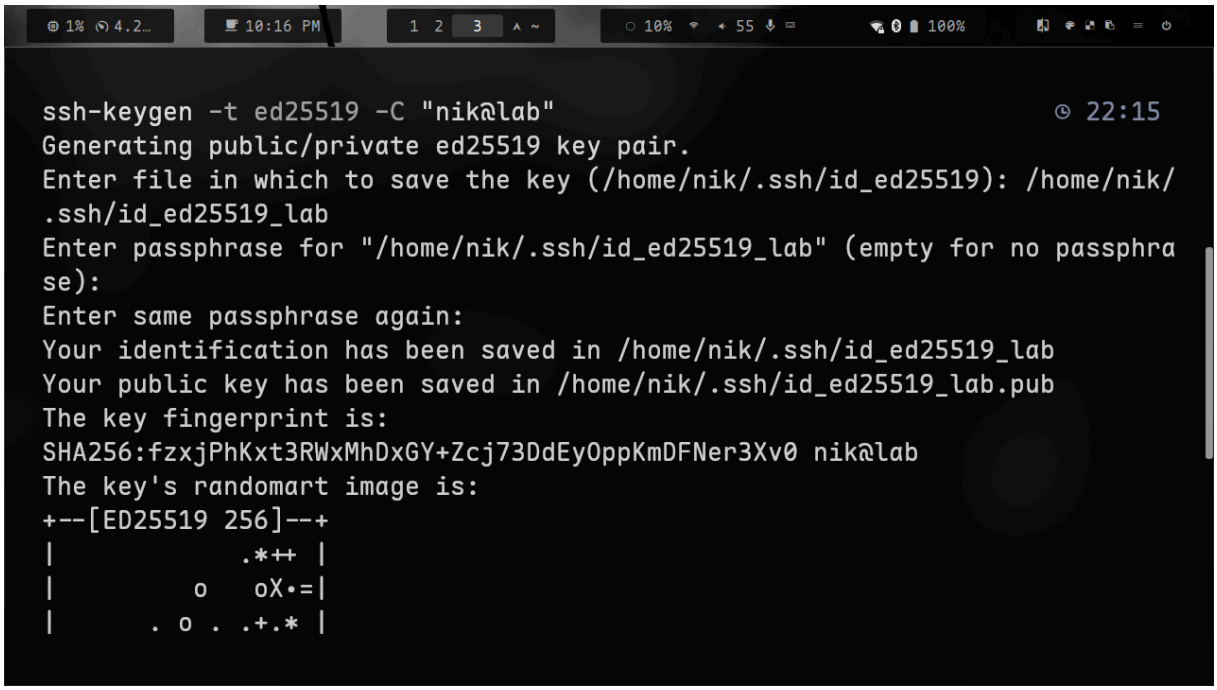

2% ...
03:18 PM
1 2 3 4 ...
10% 55 2
100%
New Tab
Split View
Copy
Paste
Find...
GNU nano 7.2 /DATA/sec2/owned.txt *
ok
fail
^G Help
^O Write Out
^W Where Is
^K Cut
^T Execute
^X Exit
^R Read File
^\ Replace
^U Paste
^J Justify

Права:

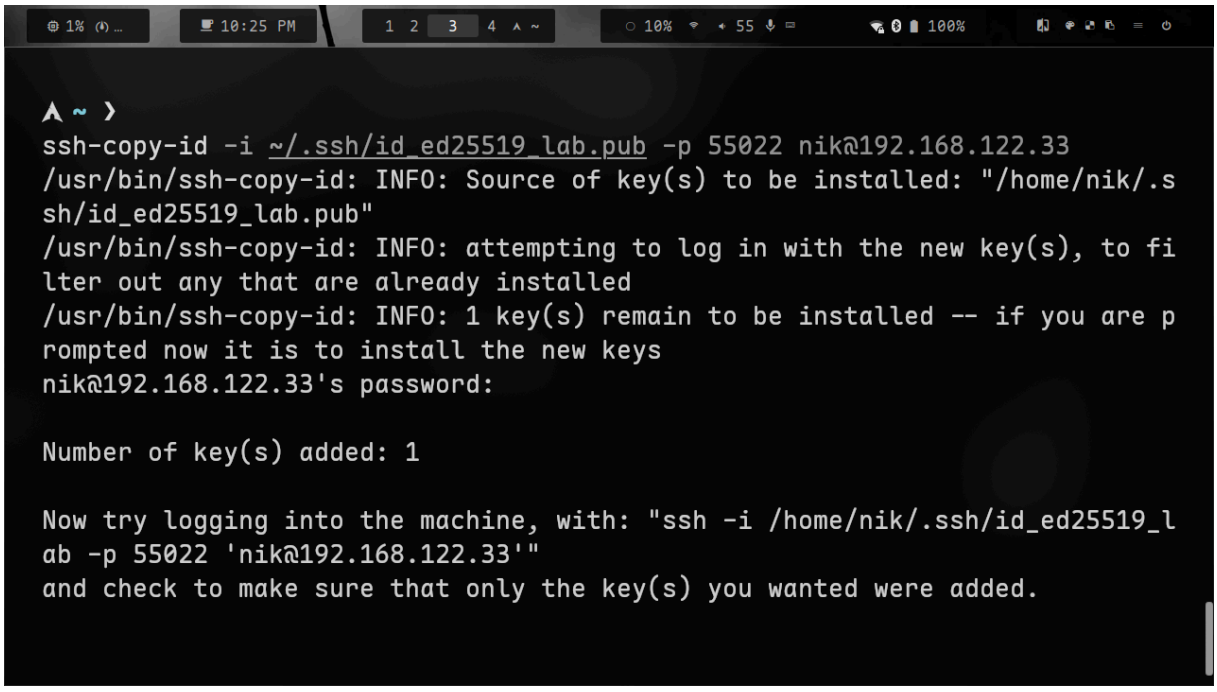


Часть 6. Настройка аутентификации по ключу.

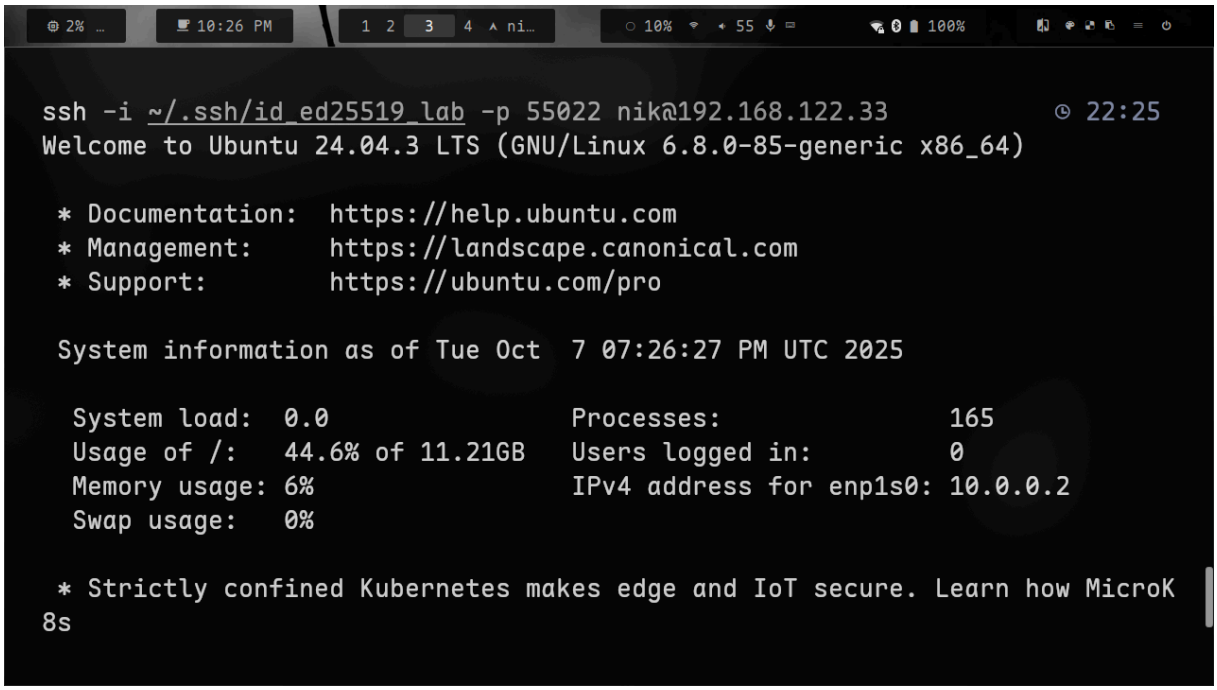
Локально я создал пару ssh ключей:



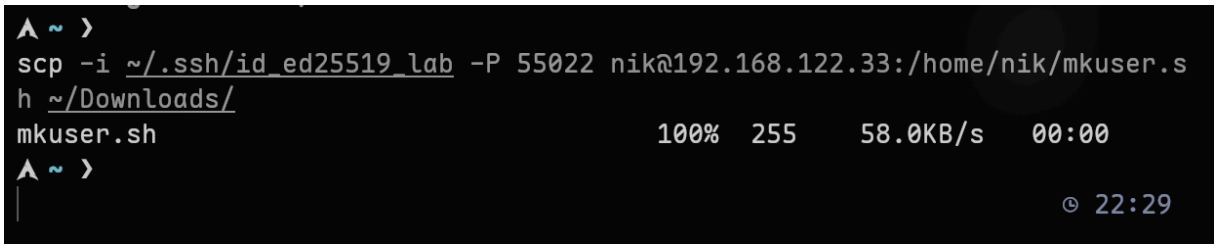
Я передал его на `c7-2`:



И теперь могу заходить без пароля:

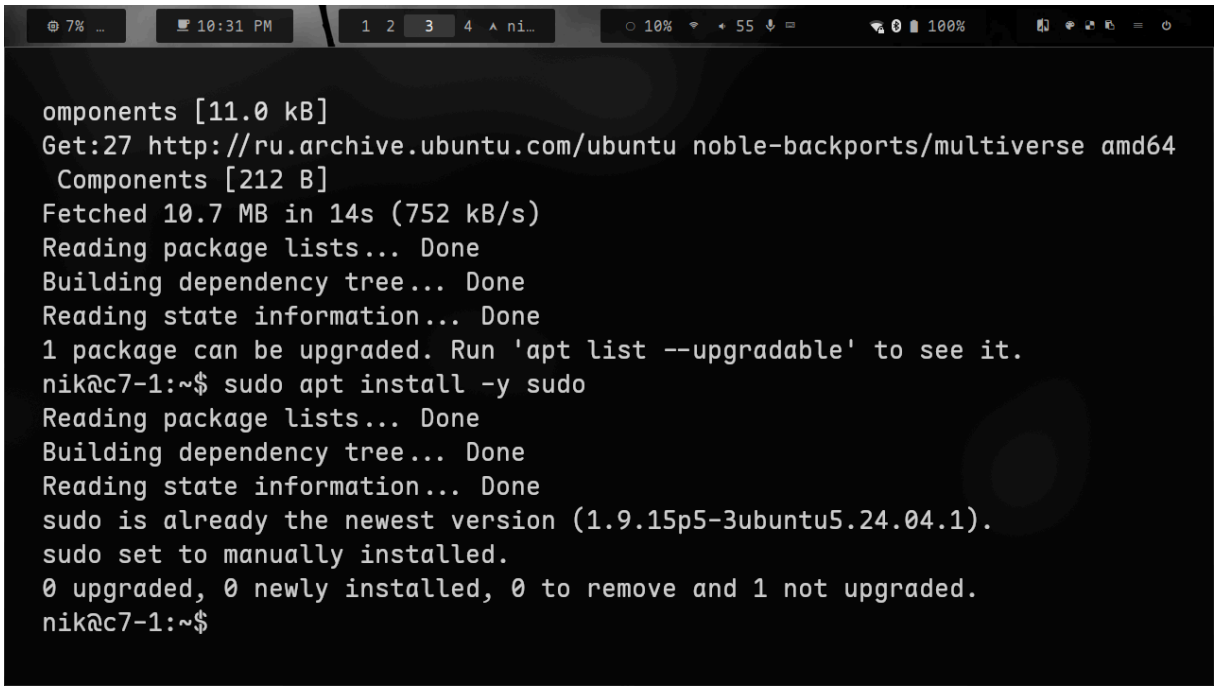


И скопировал файл к себе на компьютер:

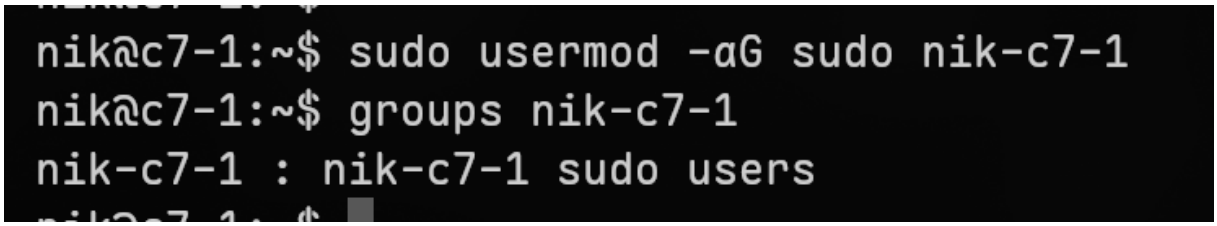


Часть 7. Sudo.

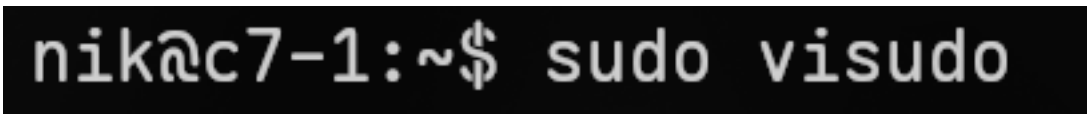
Я установил `sudo` на `c7-1`:

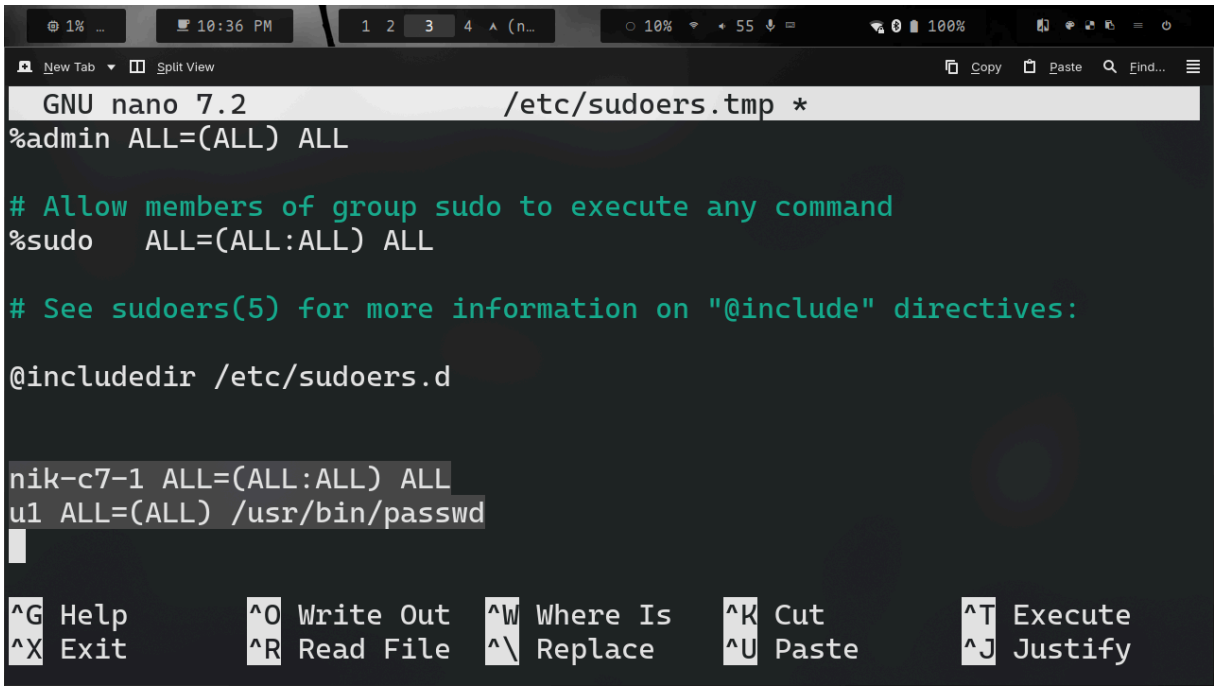


Разрешил пользователю из части 2 п.1 повышать привелегии до root.



Ограничил права первого из созданных в части 5 пользователей:





Проверил работу прав:

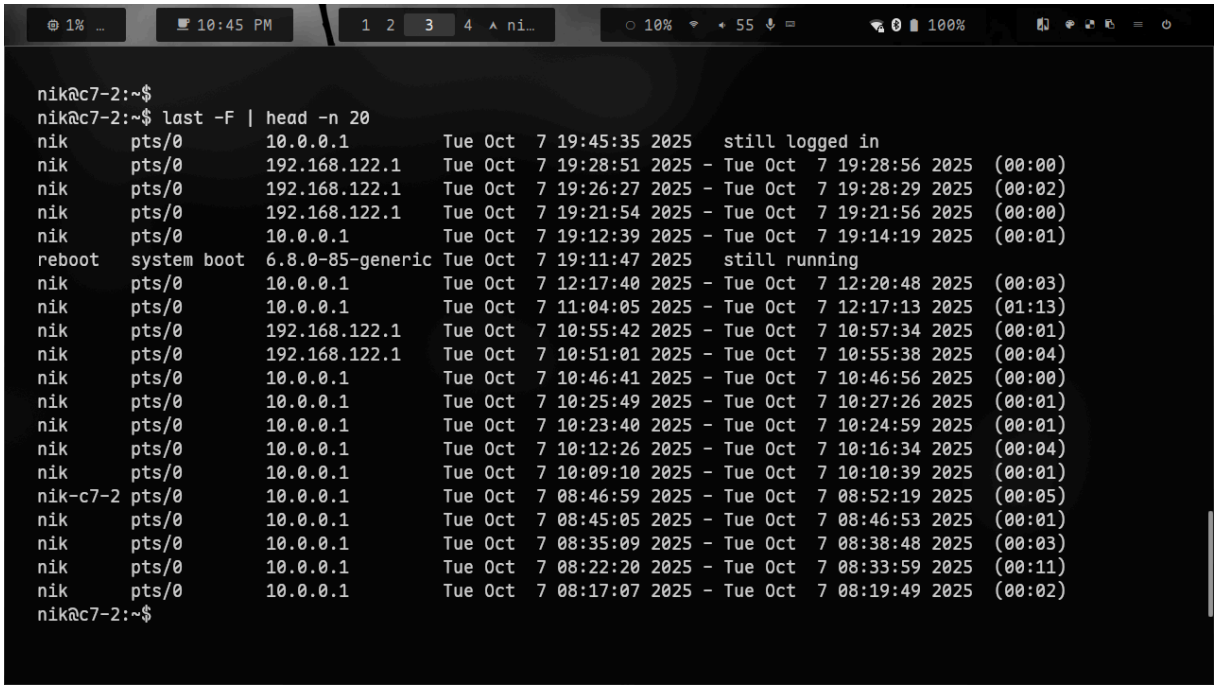
```
nik@c7-1:~$ su - nik-c7-1
Password:
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

nik-c7-1@c7-1:~$ sudo ls /root
[sudo] password for nik-c7-1:
nik-c7-1@c7-1:~$ sudo ls /root
nik-c7-1@c7-1:~$
```

```
nik@c7-1:~$ su - u1
Password:
u1@c7-1:~$ sudo passwd u1
[sudo] password for u1:
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
```

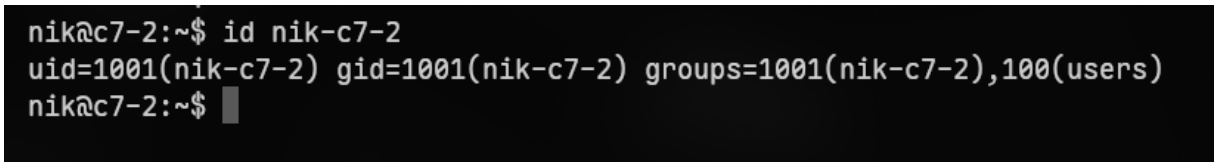
```
u1@c7-1:~$ sudo ls /root
Sorry, user u1 is not allowed to execute '/usr/bin/ls /root' as root on c7-
1.
u1@c7-1:~$
```

Часть 8. Получение информации о пользователях.

Я вывел входы пользователей в систему за текущий месяц:



Вывел информацию о пользователе из части 2 п. 1:



Ответы на вопросы.

В части 4 вы использовали готовые команды для настройки NAT. Поясните какие параметры передаются в ключах команды iptables.

```
iptables -t nat -A POSTROUTING -o enp1s0 -s 10.0.0.0/24 -j
MASQUERADE
```

`-t nat` — выбрать таблицу NAT, `-A POSTROUTING` — добавить правило в цепочку POSTROUTING, `-o enp1s0` — указывает выходной интерфейс, `-s 10.0.0.0/24` — диапазон внутренних адресов, для которых применяется правило, `-j MASQUERADE` — действие: подменить IP-адрес источника на внешний адрес интерфейса. Аналогично, при `-j SNAT --to-source <ip>` адрес подменяется на указанный вручную.

При создании ключей ssh программа-генератор предлагает ввести пароль. Зачем он нужен и для чего используется?

Пароль защищает приватный ключ. Даже если файл ключа попадёт в чужие руки, без пароля им нельзя воспользоваться.

При первом подключении по ssh к новому серверу вам выводится хэш и программа предлагает принять его или отклонить. Зачем это нужно?

Это fingerprint публичного ключа сервера. Он нужен, чтобы убедиться, что мы подключаемся к правильному серверу, а не к подменённому. После подтверждения отпечаток сохраняется в `~/.ssh/known_hosts`. Если при следующем подключении хэш изменился — SSH предупредит о возможной подмене.

Как на сервере ssh определить сколько подключений по ssh есть и от каких пользователей?

```
ss -tuna | grep ':22'
```

или

```
who
```

или

```
ps aux | grep sshd
```

`sshd` показывает каждое активное соединение, `who` — кто вошёл в систему, с каких ip, и сколько пользователей сейчас в системе.

Если у двух пользователей в Linux будут одинаковые пароли, то сможем ли мы понять это по данным в файле `/etc/shadow`? Почему?

Нет. Файл `/etc/shadow` хранит хэши паролей с солью, то есть у каждого пароля добавляется случайное значение перед хэшированием. Даже если два пользователя используют одинаковый пароль, их хэши будут разными.

Заполните таблицу, описывающую действие различных атрибутов прав (r, w, x) и атрибутов безопасности (suid, sgid, stiky bit) при назначении их файлу или каталогу. В таблице должны быть следующие столбцы:

атрибут	**сокращенное название**	**значение действия для файла**	**значение действия для каталога**
read	r	разрешает чтение файла	разрешает просматривать содержимое каталога `ls`
write	w	разрешает изменять или удалять файл	разрешает создавать и удалять файлы внутри каталога
execute	x	разрешает выполнять файл как программу	разрешает входить в каталог `cd`
setuid	s (u+s)	файл выполняется с правами владельца	не применяется
setgid	s (g+s)	файл выполняется с правами группы	новые файлы внутри каталога наследуют группу каталога
sticky	t	не применяется	удалять файлы может только владелец или root (пример: /tmp)

В Linux существует расширенные права на файлы или каталоги. Работать с ними можно с помощью утилит satfacl и getfacl. Приведите пример команды, с помощью которой мы можем дать конкретному пользователю все права на файл, не делая его владельцем и не добавляя его в группы.

Чтобы дать пользователю `vlad` все права на файл `/DATA/info.txt`, не меняя владельца и групп:

```
sudo setfacl -m u:vlad:rwx /DATA/info.txt
```

Проверить можно при помощи:

```
getfacl /DATA/info.txt
```

Вывод.
В ходе лабораторной работы я изучил основы администрирования linux-систем: настройка nat, управление пользователями и правами доступа, работа с acl, организация аутентификации по ssh-ключам и настройка sudo. На практике реализовано взаимодействие двух виртуальных машин, проброс портов и защита доступа.