

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №7
по дисциплине «Сети и телекоммуникации»
Тема: Сетевые экраны. IPTABLES

Студент гр. 1304

Шаврин А.П.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2023

Цель работы.

Изучение принципов работы с сетевыми экранами. Освоить блокирование и разрешение приема и отправки пакетов с помощью iptables, а также настройку логирования событий.

Задание.

Для выполнения лабораторной работы необходимо настроить три виртуальные машины Ub1, Ub2 и Ub3 так, чтобы они находились в одной подсети.

Кроме того, для некоторых пунктов необходимо установить дополнительные службы на виртуальные машины: apache2, ftpd – и выполнить следующие задачи:

1. Заблокировать доступ по IP-адресу ПК Ub1 к Ub3. Продемонстрировать результаты с попыткой подключения Ub1 и Ub2 к Ub3.
2. Заблокировать доступ по 21-му порту на Ub1. Продемонстрировать возможность доступа по ssh на Ub1 и невозможность доступа по 21-му порту.
3. Разрешить доступ только по ssh на Ub2. Предоставить результат.
4. Запретить ICMP-запросы на IP-адрес 8.8.8.8 двумя способами. Необходимо создать два правила: в цепочке INPUT и цепочке OUTPUT. С помощью Wireshark на хосте нужно продемонстрировать разницу между двумя способами блокировки и сделать вывод о том, какой вариант эффективнее.
5. Полностью запретить доступ к Ub3. Разрешить доступ по ICMP-протоколу.
6. Запретить подключение к Ub1 по порту 80. Настроить логирование попыток подключения по 80-му порту. Продемонстрировать результаты логирования.
7. Заблокировать доступ по 80-му порту к Ub3 с Ub1 по его MAC-адресу. Продемонстрировать результат, сменить MAC-адрес на Ub3 и продемонстрировать успешное подключение к Ub3 по 80-му порту.

8. Полностью закрыть доступ к Ub1. Разрешить доступ для Ub3 к Ub1, используя диапазон портов 20–79. В результате необходимо показать невозможность подключения к 80 порту и возможность – к ssh или ftp.

9. Разрешить только одно ssh-подключение к Ub3. Продемонстрировать результат попытки подключения с Ub2 при наличии открытой ssh-сессии с Ub1 к Ub3.

Выполнение работы.

Были развёрнуты три виртуальные машины с адресами 192.168.0.1/24 (ub1), 192.168.0.2/24 (ub2), 192.168.0.3/24 (ub3). Доступность машин приведена на рисунках 1 – 3.

```
alex@ub1:~$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.398 ms
^C
--- 192.168.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.398/0.398/0.398/0.000 ms
alex@ub1:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.544 ms
^C
--- 192.168.0.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.544/0.544/0.544/0.000 ms
```

Рисунок 1. Доступность ub2 и ub3 с ub1

```
alex@ub2:~$ ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.470 ms
^C
--- 192.168.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.470/0.470/0.470/0.000 ms
alex@ub2:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.397 ms
^C
--- 192.168.0.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.397/0.397/0.397/0.000 ms
```

Рисунок 2. Доступность ub1 и ub3 с ub2

```
alex@ub3:~$ ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.441 ms
^C
--- 192.168.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.441/0.441/0.441/0.000 ms
alex@ub3:~$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.379 ms
^C
--- 192.168.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.379/0.379/0.379/0.000 ms
```

Рисунок 3. Доступность ub1 и ub2 с ub3

1. Заблокируем доступ по IP-адресу от Ub1 к Ub3 (см. рис. 4). Для этого на Ub1 объявим правило в цепочке OUTPUT на IP-адрес Ub3, после чего проверим доступность Ub3 с Ub1 и Ub3 с Ub2 (см. рис. 5 - 6).

Запрет был поставлен с помощью команды:

```
sudo iptables -A OUTPUT -d 192.168.0.3 -j DROP
```

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
DROP      all  --  anywhere             192.168.0.3
```

Рисунок 4. Таблица правил на ub1.

```
alex@ub1:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
^C
--- 192.168.0.3 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Рисунок 5. Не доступность ub3 с ub1

```
alex@ub2:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.532 ms
^C
--- 192.168.0.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.532/0.532/0.532/0.000 ms
```

Рисунок 6. Доступность ub3 с ub2

2. Заблокируем доступ к 21-му порту на Ub1 (см. рис. 7). Далее проверим доступность Ub1 с Ub3 через ssh и через порт 21 (см. рис. 8-9).

Для блокировки доступа была применена команда:

```
sudo iptables -A INPUT -p tcp --dport 21 -j REJECT
```

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
REJECT     tcp  --  anywhere             tcp dpt:ftp reject-with icmp-port-unre
achable

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Рисунок 7. Таблица правил на ub1.

```
alex@ub3:~$ sudo nc -vz 192.168.0.1 21
[sudo] password for alex:
nc: connect to 192.168.0.1 port 21 (tcp) failed: Connection refused
alex@ub3:~$
```

Рисунок 8. Не доступность ub1 с ub3 по 21 порту.

```
alex@ub3:~$ ssh 192.168.0.1
The authenticity of host '192.168.0.1 (192.168.0.1)' can't be established.
ECDSA key fingerprint is SHA256:JETJEP86ZJR8jL+2w4c6wjn0601+zKL1+X7IxiPvtwQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.1' (ECDSA) to the list of known hosts.
alex@192.168.0.1's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-186-generic x86_64)

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

112 packages can be updated.
81 updates are security updates.

Last login: Tue Apr 11 20:07:07 2023
alex@ub1:~$
```

Рисунок 9. Доступность ub1 с ub3 по ssh

3. Разрешим доступ только по ssh на Ub2. Для этого разрешим пакеты через ssh и запретим все остальные (см. рис. 10). Далее проверим доступность Ub2 с Ub1 при помощи команды ping и доступность Ub2 с Ub1 по ssh (см. рис. 11-12).

Команды представлены ниже:

```
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

```
sudo iptables -A INPUT -j REJECT
```

```
alex@ub2:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination            tcp dpt:ssh
ACCEPT    tcp  --  anywhere              anywhere               tcp dpt:ssh
REJECT     all  --  anywhere              anywhere               reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Рисунок 10. Таблица правил на ub2.

```
alex@ub1:~$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
From 192.168.0.2 icmp_seq=1 Destination Port Unreachable
^C
--- 192.168.0.2 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms
```

Рисунок 11. Не доступность ub2 с ub1.

```
alex@ub1:~$ ssh 192.168.0.2
The authenticity of host '192.168.0.2 (192.168.0.2)' can't be established.
ECDSA key fingerprint is SHA256:mIuDPd7x74/H1KMjzSNcrlrDxoxUn7DW/hNWYxI/4Xw.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.2' (ECDSA) to the list of known hosts.
alex@192.168.0.2's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-186-generic x86_64)

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

112 packages can be updated.
81 updates are security updates.

Last login: Tue Apr 11 20:04:56 2023
alex@ub2:~$
```

Рисунок 12. Доступность ub2 с ub1 по ssh.

4. Запретим на Ub1 ICMP запросы на 8.8.8.8 через правило в цепочке INPUT (см. рис. 13), а затем обновим таблицы и запретим через правило в цепочке OUTPUT (см. рис. 16). Для анализа трафика вместо Wireshark будем использовать версию данной программы для CLI – TShark.

Команды представлены ниже:

```
sudo iptables -A INPUT -p icmp -s 8.8.8.8 -j REJECT
```

```
sudo iptables -A OUTPUT -p icmp -d 8.8.8.8 -j REJECT
```

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination            reject-with icmp-port-unreachable
REJECT     icmp -- dns.google             anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Рисунок 13. Правила на ub1 при цепочке INPUT.

```
alex@ub1:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4009ms
```

Рисунок 14. Запрет ICMP запросов к 8.8.8.8 с Ub1 при цепочке INPUT

```
alex@ub1:~$ sudo tshark -i enp0s8 icmp
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s8'
  1 0.000000000 192.168.31.149 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0x2ef6, seq=33/8448,
    ttl=64
  2 0.022802606      8.8.8.8 → 192.168.31.149 ICMP 98 Echo (ping) reply    id=0x2ef6, seq=33/8448,
    ttl=107 (request in 1)
  3 0.022824838 192.168.31.149 → 8.8.8.8      ICMP 126 Destination unreachable (Port unreachable)
  4 1.002341291 192.168.31.149 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0x2ef6, seq=34/8704,
    ttl=64
  5 1.025745410      8.8.8.8 → 192.168.31.149 ICMP 98 Echo (ping) reply    id=0x2ef6, seq=34/8704,
    ttl=107 (request in 4)
  6 1.025767612 192.168.31.149 → 8.8.8.8      ICMP 126 Destination unreachable (Port unreachable)
  7 2.002355023 192.168.31.149 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0x2ef6, seq=35/8960,
    ttl=64
  8 2.025444775      8.8.8.8 → 192.168.31.149 ICMP 98 Echo (ping) reply    id=0x2ef6, seq=35/8960,
    ttl=107 (request in 7)
  9 2.025471685 192.168.31.149 → 8.8.8.8      ICMP 126 Destination unreachable (Port unreachable)
^C 10 3.010985795 192.168.31.149 → 8.8.8.8      ICMP 98 Echo (ping) request  id=0x2ef6, seq=36/921
6, ttl=64
 11 3.030184466      8.8.8.8 → 192.168.31.149 ICMP 98 Echo (ping) reply    id=0x2ef6, seq=36/9216,
    ttl=107 (request in 10)
 12 3.030217759 192.168.31.149 → 8.8.8.8      ICMP 126 Destination unreachable (Port unreachable)
12 packets captured
alex@ub1:~$
```

Рисунок 15. Отправка и прием ICMP запросов к 8.8.8.8 с Ub1 при цепочке INPUT (утилита TShark)

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
REJECT     icmp -- anywhere             dns.google             reject-with icmp-port-unreachable
```

Рисунок 16. Правила на ub1 при цепочке OUTPUT.

```
alex@ub1:~$ ping 8.8.8.8 -c 2
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 192.168.31.149 icmp_seq=1 Destination Port Unreachable
From 192.168.31.149 icmp_seq=1 Destination Port Unreachable

--- 8.8.8.8 ping statistics ---
0 packets transmitted, 0 received, +2 errors
```

Рисунок 17. Запрет ICMP запросов к 8.8.8.8 с Ub1 при цепочке OUTPUT

```
alex@ub1:~$ sudo tshark -i enp0s8 icmp
Running as user "root" and group "root". This could be dangerous.
Capturing on 'enp0s8'
```

Рисунок 18. Отправка и прием ICMP запросов к 8.8.8.8 с Ub1 при цепочке OUTPUT (утилита TShark)

При использовании цепочки INPUT запросы отправляются на 8.8.8.8 и оттуда же приходят ответы, которые затем отклоняются. При использовании цепочки OUTPUT пакеты вовсе не отправляются.


Таким образом вариант через цепочку OUTPUT эффективнее, так как лишние ICMP запросы не нагружают сеть.

5. Полностью запретим доступ к Ub3, однако оставим возможность доступа по ICMP протоколу (см. рис. 19). Также проверим доступность с Ub2 на Ub3 через ssh и при помощи команды ping (см. рис. 20-21).

Команды представлены ниже:

```
sudo iptables -A INPUT -p icmp -j REJECT
```

```
sudo iptables -A INPUT -j REJECT
```




```
alex@ub3:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     icmp -- anywhere             anywhere
REJECT     all  -- anywhere             anywhere             reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

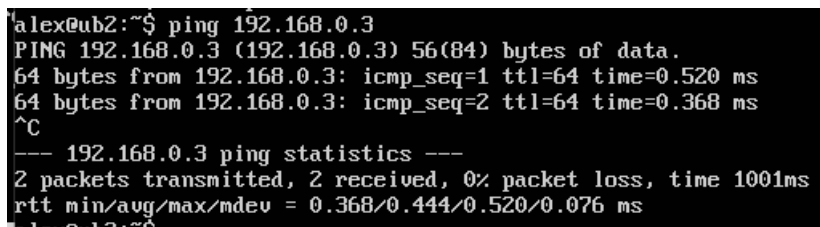
Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Рисунок 19. Правила на ub3.



```
alex@ub2:~$ ssh 192.168.0.3
ssh: connect to host 192.168.0.3 port 22: Connection refused
alex@ub2:~$
```

Рисунок 20. Недоступность ub3 с ub2 через ssh



```
alex@ub2:~$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data.
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.520 ms
64 bytes from 192.168.0.3: icmp_seq=2 ttl=64 time=0.368 ms
^C
--- 192.168.0.3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.368/0.444/0.520/0.076 ms
alex@ub2:~$
```

Рисунок 21. Доступность ub3 с ub2 при команде ping (ICMP протокол)

6. Запретим подключение к Ub1 по порту 80, а также настроим логирование попыток данного подключения (см. рис. 22). По итогу выполнения запросов к Ub1 с Ub2 по порту 80 были составлены логи, находящиеся в файле /var/log/kern.log, в котором записаны отклоненные попытки подключения.

Команды представдены ниже:

```
Sudo iptables -A INPUT -p tcp -dport 80 -j LOG --log-prefix "80portlog"
```

```
Sudo iptables -A INPUT -p tcp -dport 80 -j REGECT
```



```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
LOG        tcp  --  anywhere              anywhere           tcp dpt:http LOG level warning prefix
'80portlog'
REJECT     tcp  --  anywhere              anywhere           tcp dpt:http reject-with icmp-port-unr
eachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Рисунок 22. Правила на ub1.

```
alex@ub2:~$ sudo nc -vz 192.168.0.1 80
nc: connect to 192.168.0.1 port 80 (tcp) failed: Connection refused
```

Рисунок 23. Недоступность ub1 с ub2 через 80 порт

```
root@ub1:/var/log# cat /var/log/kern.log | grep "80portlog"
Apr 11 21:45:57 ub1 kernel: [ 2345.333402] 80portlog IN=enp0s3 OUT= MAC=08:00:27:ae:e8:df:08:00:27:94
:40:bd:08:00 SRC=192.168.0.2 DST=192.168.0.1 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=46141 DF PROTO=TCP
SPT=49196 DPT=80 WINDOW=29200 RES=0x00 SYN URG=0
```

Рисунок 24. Лог попытки подключения.

7. Заблокируем доступ по порту 80 с Ub1 к Ub3 по его MAC-адресу. Затем изменим у Ub3 MAC-адрес и снова проверим доступ. На ub3 был запущен apache2.

Команда:

```
Sudo iptables -A INPUT -p tcp -sport 80 -m mac -mac-source
08:00:27:0F:A5:F8
```

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
REJECT     tcp  --  anywhere              anywhere           tcp spt:http MAC 08:00:27:0F:A5:F8 rej
ect-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Рисунок 25. Правила на ub1.

```
alex@ub1:~$ sudo nc -vz 192.168.0.3 80
^C
alex@ub1:~$
```

Рисунок 26. Не доступность ub3 с ub1 по 80 порту.

```
alex@ub1:~$ sudo nc -vz 192.168.0.3 80
Connection to 192.168.0.3 80 port [tcp/http] succeeded!
```

Рисунок 27. Доступность ub3 с ub1 по 80 порту после смены MAC-адреса.

8. Полностью закроем доступ к Ub1, но разрешим доступ для Ub3 к Ub1 через диапазон портов 20-79 (см. рис. 28). Затем проведем проверку недоступности соединения через порт 80 и доступности соединения через порт 22, т.е. через ssh. (см. рис. 29-30)

Команды:

```
Sudo iptables -A INPUT -p tcp -s 192.168.0.3/24 -dport 20:79 -j ACCEPT
```

```
Sudo iptables -A INPUT -j REJECT
```

```
alex@ub1:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination          tcp dpts:ftp-data:finger
ACCEPT     tcp  --  192.168.0.3            anywhere             reject-with icmp-port-unreachable
REJECT     all  --  anywhere              anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
alex@ub1:~$
```

Рисунок 28. Правила на ub1.

```
alex@ub3:~$ sudo nc -vz 192.168.0.1 80
nc: connect to 192.168.0.1 port 80 (tcp) failed: Connection refused
```

Рисунок 29. Недоступность с ub3 к ub1 по 80 порту.

```
alex@ub3:~$ ssh 192.168.0.1
alex@192.168.0.1's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-186-generic x86_64)

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

112 packages can be updated.
81 updates are security updates.

Last login: Tue Apr 11 21:19:28 2023
alex@ub1:~$
```

Рисунок 30. Доступность с ub3 к ub1 по 22 порту (ssh).

9. Разрешим только одно ssh-подключение к Ub3 (см. рис. 31). Затем, при созданном ssh подключении к Ub3 с Ub1, произведем попытку создания еще одного ssh подключения с Ub2 на Ub3, и проверим недоступность второго соединения (см. рис. 32-33).

Команда:

```
Sudo iptables -A INPUT -p tcp --syn --dport 22 -m connlimit-above 1 --connlimit-mask 0 -j REJECT
```

```
alex@ub3:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination          tcp dpt:ssh flags:FIN,SYN,RST,ACK/SYN
REJECT     tcp  --  anywhere              anywhere             #conn src/0 > 1 reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
alex@ub3:~$
```

Рисунок 31. Правила на ub1.

```
alex@ub1:~$ ssh 192.168.0.3
The authenticity of host '192.168.0.3 (192.168.0.3)' can't be established.
ECDSA key fingerprint is SHA256:JETJEP86ZjR8jL+2w4c6wjn0601+zKL1+X7IxiPwtwQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.3' (ECDSA) to the list of known hosts.
alex@192.168.0.3's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-186-generic x86_64)

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

112 packages can be updated.
81 updates are security updates.

Last login: Tue Apr 11 22:38:19 2023 from 192.168.0.2
alex@ub3:~$ _
```

Рисунок 32. Доступность с ub1 к ub3 по 22 порту (ssh)

```
alex@ub2:~$ ssh 192.168.0.3
ssh: connect to host 192.168.0.3 port 22: Connection refused
alex@ub2:~$
```

Рисунок 33. Недоступность с ub2 к ub3 по 22 порту (ssh)

Выводы.

Изучены принципы работы с сетевыми экранами. Освоены блокирование и разрешение приема и отправки пакетов с помощью iptables, а также настройка логирования событий.