

Отчет по лабораторной работе №7

Дисциплина: Администрирование сетевых подсистем

Иванов Сергей Владимирович

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Создание пользовательской службы firewalld	6
3.2	Перенаправление портов	9
3.3	Настройка Port Forwarding и Masquerading	9
3.4	Внесение изменений в настройки внутреннего окружения виртуальной машины	11
4	Ответы на контрольные вопросы	13
5	Выводы	15

Список иллюстраций

3.1	Запуск server	6
3.2	Создание файла	6
3.3	Файл ssh-custom.xml	7
3.4	Редактирование файла службы	7
3.5	Список доступных служб	8
3.6	Перезагрузка правил firewall	8
3.7	Добавление службы в FirewallD	8
3.8	Перезагрузка правил МЭ	9
3.9	Переадресация портов	9
3.10	Доступ клиента по SSH к серверу	9
3.11	Проверка возможностей перенаправления IPv4-пакетов	10
3.12	Перенаправление IPv4-пакетов на сервере	10
3.13	Включение маскардинга	10
3.14	Проверка доступности интернета	11
3.15	Создание каталогов для внесения изменений	12
3.16	Скрипт firewall.sh	12
3.17	Редактирование Vagrantfile	12

1 Цель работы

Получить навыки настройки межсетевого экрана в Linux в части переадресации портов и настройки Masquerading.

2 Задание

1. Настройте межсетевой экран виртуальной машины `server` для доступа к серверу по протоколу SSH не через 22-й порт, а через порт 2022 (см. разделы 7.4.1 и 7.4.2).
2. Настройте Port Forwarding на виртуальной машине `server` (см. разделы 7.4.3).
3. Настройте маскерадинг на виртуальной машине `server` для организации доступа клиента к сети Интернет (см. раздел 7.4.3).
4. Напишите скрипт для Vagrant, фиксирующий действия по расширенной настройке межсетевого экрана. Соответствующим образом внести изменения в Vagrantfile (см. раздел 7.4.4).

3 Выполнение лабораторной работы

3.1 Создание пользовательской службы firewalld

Загрузим операционную систему и перейдем в рабочий каталог с проектом: `cd /var/tmp/user_name/vagrant`. Запустим виртуальную машину `server`: `vagrant up server`. (рис. 1).

```
C:\Users\lserg>cd C:\work_asp\svivanov\vagrant
C:\work_asp\svivanov\vagrant>vagrant up server
Bringing machine 'server' up with 'virtualbox' provider
==> server: You assigned a static IP ending in ".1" or
```

Рис. 3.1: Запуск server

На виртуальной машине `server` войдем под пользователем и откроем терминал. Перейдем в режим суперпользователя. На основе существующего файла описания службы `ssh` создадим файл с собственным описанием:

```
cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/ssh-custom.xml
cd /etc/firewalld/services/ (рис. 2).
```

```
[svivanov@server.svivanov.net ~]$ sudo -i
[sudo] пароль для svivanov:
[root@server.svivanov.net ~]# cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/ssh-custom.xml
[root@server.svivanov.net ~]# cd /etc/firewalld/services/
[root@server.svivanov.net services]#
```

Рис. 3.2: Создание файла

Посмотрим содержимое файла службы: `cat /etc/firewalld/services/ssh-custom.xml`

Строка 1: Объявление XML-документа. Указывает версию XML (1.0) и кодировку (UTF-8).

Строка 2: Открывающий тег корневого элемента service. Все параметры службы определяются внутри этого тега.

Строка 3: Тег short содержит краткое имя службы (например, “SSH”), которое может использоваться в инструментах управления firewall.

Строка 4: Тег description содержит подробное описание службы, её назначения и условий использования.

Строка 5: Тег port определяет сетевой порт и протокол, связанные со службой.

Строка 6: Закрывающий тег корневого элемента service. Завершает определение службы. (рис. 3)

```
[root@server.svivanov.net services]# cat /etc/firewalld/services/ssh-custom.xml
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>SSH</short>
  <description>Secure Shell (SSH) is a protocol for logging into and executing commands on remote machines. It provides secure encrypted communications. If you plan on accessing your machine remotely via SSH over a firewalled interface, enable this option. You need the openssh-server package installed for this option to be useful.</description>
  <port protocol="tcp" port="22"/>
</service>
[root@server.svivanov.net services]#
```

Рис. 3.3: Файл ssh-custom.xml

Откроем файл описания службы на редактирование и заменим порт 22 на новый порт (2022). В этом же файле скорректируем описание службы для демонстрации, укажем, что это модифицированный файл службы. (рис. 4)

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>SSH</short>
  <description>This is modified file.</description>
  <port protocol="tcp" port="2022"/>
</service>
~
```

Рис. 3.4: Редактирование файла службы

Просмотрим список доступных FirewallD служб: `firewall-cmd --get-services`. Обратим внимание, что новая служба ещё не отображается в списке. (рис. 5)

```
[root@server.svivanov.net services]# firewall-cmd --get-services
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-client amqp amqps anno-1602 anno-1800
apcupsd aseqnet audit ausweisapp2 bacula bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp b
itcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkm
k-agent civilization-iv civilization-v cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unica
st dhcp dhcpv6 dhcpv6-client distcc dns dns-over-qtcp dns-over-tls docker-registry docker-swarm dropbox-lansync e
lasticsearch etcd-client etcd-server factorio finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps f
reeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpsd grafana gre high-availability
http http3 https ident imap imaps iperf2 iperf3 ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadm
in kdeconnect kerberos kibana klogon kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-control
-plane-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-services kube-scheduler kube-s
cheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-netw
ork llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minecraft minidlna mndp mongodb mosh
mountd mpd mqtt mqtt-tls ms-wbt mssql murmur mysql nbd nebula need-for-speed-most-wanted netbios-ns netdata-dashb
oard nfs nfs3 nmea-0183 nrpe ntp nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole ple
```

Рис. 3.5: Список доступных служб

Перегрузим правила межсетевого экрана с сохранением информации о состо-
янии и вновь выведем на экран список служб, а также список активных служб:

```
firewall-cmd --reload
firewall-cmd --get-services
firewall-cmd --list-services
```

Убедимся, что созданная служба отображается в списке доступных для
Firewalld служб, но не активирована. (рис. 6)

```
-https wireguard ws-discovery ws-discovery-client ws-discovery-host ws-
ttp wsmans wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zab
r zabbix-trapper zabbix-web-service zero-k zerotier
[root@server.svivanov.net services]# firewall-cmd --list-services
cockpit dhcp dhcpv6-client dns http https ssh
[root@server.svivanov.net services]#
```

Рис. 3.6: Перегрузка правил firewall

Добавим новую службу в Firewalld и выведем на экран список активных
служб:

```
firewall-cmd --add-service=ssh-custom
firewall-cmd --list-services (рис. 7)
```

```
[root@server.svivanov.net services]# firewall-cmd --add-service=ssh-custom
success
[root@server.svivanov.net services]# firewall-cmd --list-services
cockpit dhcp dhcpv6-client dns http https ssh ssh-custom
[root@server.svivanov.net services]#
```

Рис. 3.7: Добавление службы в Firewalld

Перегрузим правила межсетевого экрана с сохранением информации о состо-
янии:

firewall-cmd --add-service=ssh-custom --permanent

firewall-cmd --reload (рис. 8)

```
[root@server.svivanov.net services]# firewall-cmd --add-service=ssh-custom --permanent
success
[root@server.svivanov.net services]# firewall-cmd --reload
success
[root@server.svivanov.net services]#
```

Рис. 3.8: Перезагрузка правил МЭ

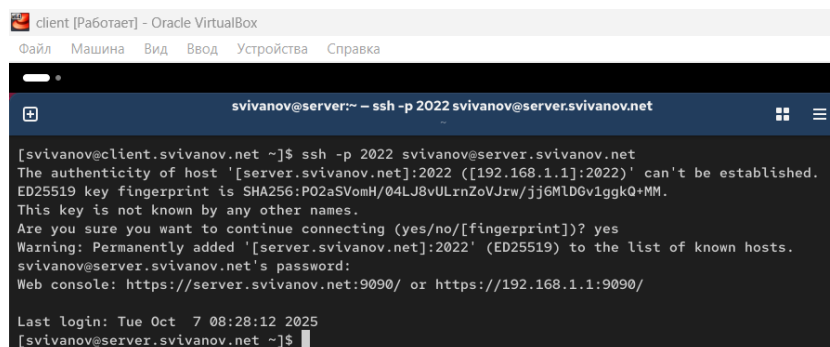
3.2 Перенаправление портов

Организуем на сервере переадресацию с порта 2022 на порт 22: firewall-cmd --add-forward-port=port=2022:proto=tcp:toport=22 (рис. 9)

```
[root@server.svivanov.net services]# firewall-cmd --add-forward-port=port=2022:proto=tcp:toport=22
success
[root@server.svivanov.net services]#
```

Рис. 3.9: Переадресация портов

На клиенте попробуем получить доступ по SSH к серверу через порт 2022: ssh -p 2022 user@server.user.net (рис. 10)



```
client [Работает] - Oracle VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

svivanov@server:~ - ssh -p 2022 svivanov@server.svivanov.net

[svivanov@client.svivanov.net ~]$ ssh -p 2022 svivanov@server.svivanov.net
The authenticity of host '[server.svivanov.net]:2022 ([192.168.1.1]:2022)' can't be established.
ED25519 key fingerprint is SHA256:P02aSVomH/04LJ8vULrnZoVJrw/jj6MD6viggkQ+MM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[server.svivanov.net]:2022' (ED25519) to the list of known hosts.
svivanov@server.svivanov.net's password:
Web console: https://server.svivanov.net:9090/ or https://192.168.1.1:9090/

Last login: Tue Oct 7 08:28:12 2025
[svivanov@server.svivanov.net ~]$
```

Рис. 3.10: Доступ клиента по SSH к серверу

3.3 Настройка Port Forwarding и Masquerading

На сервере посмотрим, активирована ли в ядре системы возможность перенаправления IPv4-пакетов пакетов: sysctl -a | grep forward. Видим что выключена. (рис. 11)

```
net.ipv4.conf.lo.mc_forwarding = 0
net.ipv4.ip_forward = 0
net.ipv4.ip_forward_update_priority = 1
net.ipv4.ip_forward_use_pmtu = 0
```

Рис. 3.11: Проверка возможностей перенаправления IPv4-пакетов

Включим перенаправление IPv4-пакетов на сервере:

echo "net.ipv4.ip_forward = 1" > /etc/sysctl.d/90-forward.conf

sysctl -p /etc/sysctl.d/90-forward.conf (рис. 12)

```
[root@server.svivanov.net services]# echo "net.ipv4.ip_forward = 1" > /etc/sysctl.d/90-forward.conf
[root@server.svivanov.net services]# sysctl -p /etc/sysctl.d/90-forward.conf
net.ipv4.ip_forward = 1
[root@server.svivanov.net services]#
```

Рис. 3.12: Перенаправление IPv4-пакетов на сервере

Включим маскарадинг на сервере:

firewall-cmd --zone=public --add-masquerade --permanent

firewall-cmd --reload (рис. 13)

```
[root@server.svivanov.net services]# firewall-cmd --zone=public --add-masquerade --permanent
success
[root@server.svivanov.net services]# firewall-cmd --reload
success
[root@server.svivanov.net services]#
```

Рис. 3.13: Включение маскарадинга

На клиенте проверим доступность выхода в Интернет. (рис. 14)

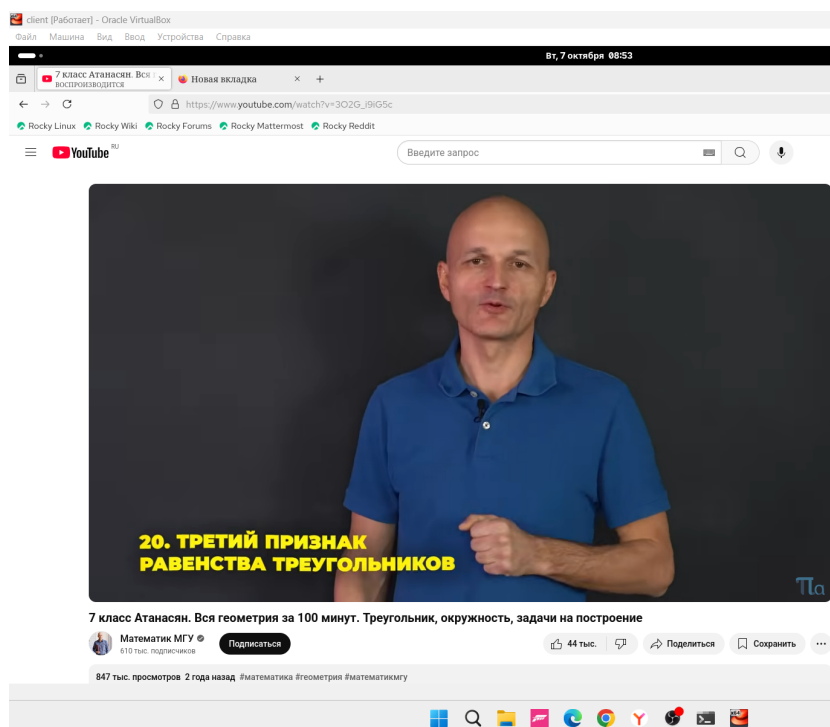


Рис. 3.14: Проверка доступности интернета

3.4 Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине `server` перейдем в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создадим в нём каталог `firewall`, в который поместим в соответствующие подкаталоги конфигурационные файлы `Firewalld`:

```
cd /vagrant/provision/server
```

```
mkdir -p /vagrant/provision/server/firewall/etc/firewalld/services
```

```
mkdir -p /vagrant/provision/server/firewall/etc/sysctl.d
```

```
cp -r /etc/firewalld/services/ssh-custom.xml /vagrant/provision/server/firewall/etc/firewalld/serv
```

```
cp -r /etc/sysctl.d/90-forward.conf /vagrant/provision/server/firewall/etc/sysctl.d/.
```

(рис. 15)

```
[root@server.svivanov.net services]# cd /vagrant/provision/server
[root@server.svivanov.net server]# mkdir -p /vagrant/provision/server/firewall/etc/firewalld/services
[root@server.svivanov.net server]# mkdir -p /vagrant/provision/server/firewall/etc/sysctl.d
[root@server.svivanov.net server]# cp -r /etc/firewalld/services/ssh-custom.xml /vagrant/provision/server/firewall/etc/firewalld/services/
[root@server.svivanov.net server]# cp -r /etc/sysctl.d/90-forward.conf /vagrant/provision/server/firewall/etc/sysctl.d/
[root@server.svivanov.net server]#
```

Рис. 3.15: Создание каталогов для внесения изменений

В каталоге /vagrant/provision/server создадим файл firewall.sh:

```
cd /vagrant/provision/server
```

```
touch firewall.sh
```

```
chmod +x firewall.sh
```

Открыв его на редактирование, пропишем в нём следующий скрипт. (рис. 16)

```
#!/bin/bash
echo "Provisioning script $0"
echo "Copy configuration files"
cp -R /vagrant/provision/server/firewall/etc/* /etc
echo "Configure masquerading"
firewall-cmd --add-service=ssh-custom --permanent
firewall-cmd --add-forward-port=port=2022:proto=tcp:toport=22 --permanent
firewall-cmd --zone=public --add-masquerade --permanent
firewall-cmd --reload
restorecon -vR /etc
~
```

Рис. 3.16: Скрипт firewall.sh

Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile необходимо добавить в разделе конфигурации для сервера: (рис. 17)

```
server.vm.provision "server firewall",
    type: "shell",
    preserve_order: true,
    path: "provision/server/firewall.sh"
```

Рис. 3.17: Редактирование Vagrantfile

4 Ответы на контрольные вопросы

1. Где хранятся пользовательские файлы firewalld?

Пользовательские файлы firewalld хранятся в:

/etc/firewalld/ - для пользовательских конфигураций (сервисы, зоны и т.д.)

Конкретно для служб:

/etc/firewalld/services/ - пользовательские файлы служб

/usr/lib/firewalld/services/ - системные файлы служб (предустановленные)

2. Какую строку надо включить в пользовательский файл службы, чтобы указать порт TCP 2022?

```
<port protocol="tcp" port="2022"/>
```

3. Какая команда позволяет вам перечислить все службы, доступные в настоящее время на вашем сервере?

```
firewall-cmd --list-services
```

Или для получения более подробной информации со всеми предустановленными службами:

```
firewall-cmd --get-services
```

4. В чем разница между трансляцией сетевых адресов (NAT) и маскардингом (masquerading)?

NAT (Network Address Translation):

- Статическое преобразование адресов
- Постоянное соответствие между внутренними и внешними адресами

- Используется когда есть выделенные внешние IP-адреса

Masquerading:

- Динамический NAT
- Использует IP-адрес интерфейса шлюза
- Автоматически подстраивается при изменении IP-адреса интерфейса
- Чаще используется при динамических IP-адресах (например, DHCP)

5. Какая команда разрешает входящий трафик на порт 4404 и перенаправляет его в службу ssh по IP-адресу 10.0.0.10?

`firewall-cmd --add-forward-port=port=4404:proto=tcp:toport=22:toaddr=10.0.0.10`

6. Какая команда используется для включения маскарадинга IP-пакетов для всех пакетов, выходящих в зону public?

`firewall-cmd --zone=public --add-masquerade`

5 Выводы

В ходе выполнения лабораторной работы мы приобрели навыки настройки межсетевого экрана в Linux в части переадресации портов и настройки Masquerading.