

# **Отчёт по лабораторной работе №7**

**Дисциплина: Администрирование сетевых подсистем**

Мишина Анастасия Алексеевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Создание пользовательской службы firewalld . . . . .	6
2.2	Перенаправление портов . . . . .	8
2.3	Настройка Port Forwarding и Masquerading . . . . .	9
2.4	Внесение изменений в настройки внутреннего окружения виртуальной машины . . . . .	10
<b>3</b>	<b>Выводы</b>	<b>12</b>
<b>4</b>	<b>Ответы на контрольные вопросы</b>	<b>13</b>

# Список иллюстраций

2.1	Создание собственного файла описания службы и просмотр	6
2.2	Редактирование файла описания службы . . . . .	7
2.3	Новая служба в списке доступных служб . . . . .	7
2.4	Добавление новой службы и просмотр списка активных служб, сохранение информации о состоянии . . . . .	8
2.5	Доступ по SSH к серверу через порт 2022 на клиенте . . . .	8
2.6	Доступ по SSH к серверу через порт 2022 на клиенте . . . .	8
2.7	Включение перенаправления пакетов и включение маскирования . . . . .	9
2.8	Создание каталогов и копирование конфигурационных файлов, создание скрипта firewall.sh . . . . .	10
2.9	Редактирование firewall.sh . . . . .	10

# Список таблиц

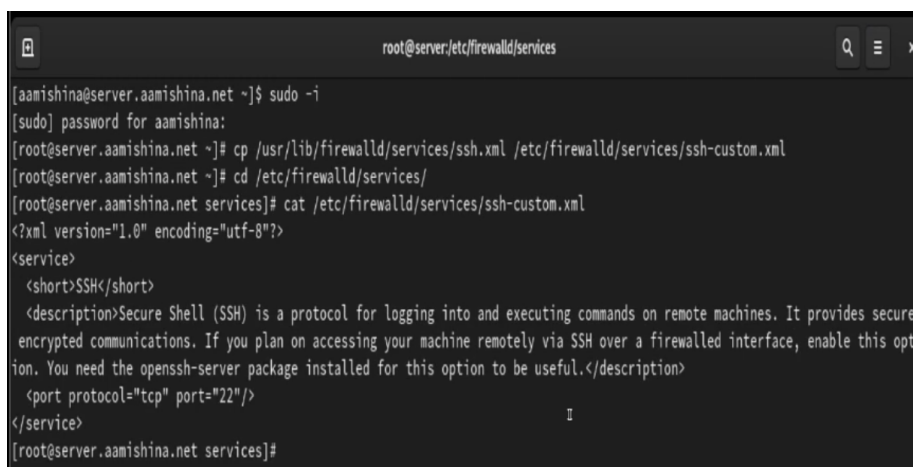
# 1 Цель работы

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

## 2 Выполнение лабораторной работы

### 2.1 Создание пользовательской службы firewalld

Запускаем ВМ через рабочий каталог. На ВМ server входим под собственным пользователем и переходим в режим суперпользователя. На основе существующего файла описания службы ssh создаем файл с собственным описанием. Просматриваем содержимое файла (рис. 2.1).



```
root@server:/etc/firewalld/services

[aamishina@server.aamishina.net ~]$ sudo -i
[sudo] password for aamishina:
[root@server.aamishina.net ~]# cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/ssh-custom.xml
[root@server.aamishina.net ~]# cd /etc/firewalld/services/
[root@server.aamishina.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>
I
[root@server.aamishina.net services]#
```

Рис. 2.1: Создание собственного файла описания службы и просмотр

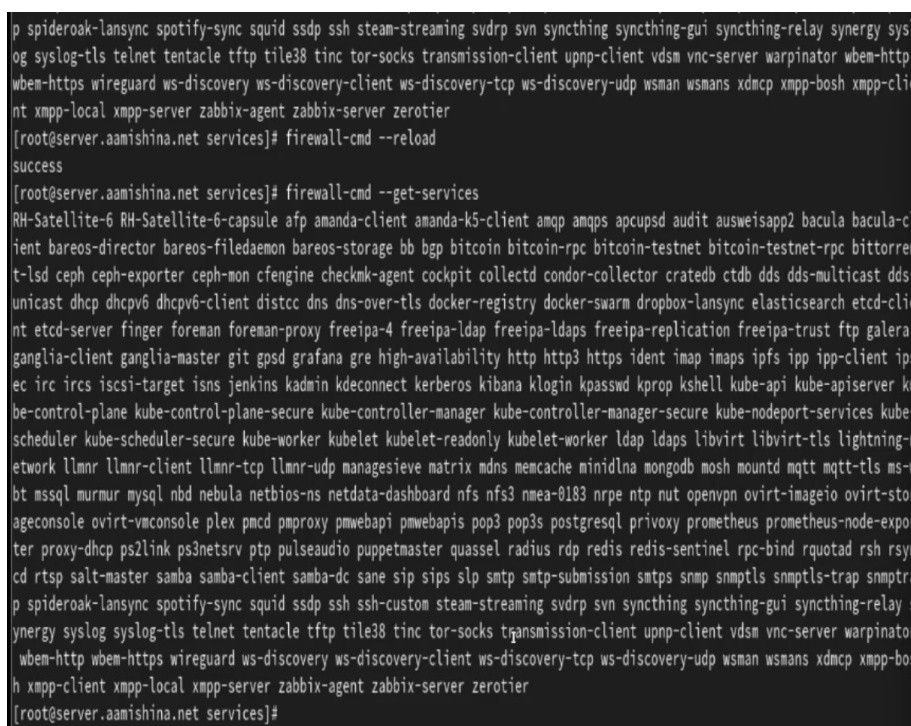
Открываем файл на редактирование и меняем порт 22 на порт 2022, в описании службы указав, что файл был модифицирован (рис. 2.2)



```
root@server:/etc/firewalld/services
<?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. File was modified.</description>
  <port protocol="tcp" port="2022"/>
</service>
```

Рис. 2.2: Редактирование файла описания службы

Просматриваем список доступных служб (новой службы пока нет). Перезагружаем правила межсетевого экрана, снова просматриваем список доступных служб и видим новую (рис. 2.3)



```
p spideroak-lansync spotify-sync squid sssd ssh steam-streaming svdrp svn syncthing syncthing-gui syncthing-relay synergy syslog
og syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsms vnc-server warpinator wbm-http
wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans xdmcp xmpp-bosh xmpp-clie
nt xmpp-local xmpp-server zabbix-agent zabbix-server zerotier
[root@server.aamishina.net services]# firewall-cmd --reload
success
[root@server.aamishina.net services]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 bacula bacula-cl
ient bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorren
t-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-
unicast dhcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-clie
nt etcd-server finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera
ganglia-client ganglia-master git gssd grafana gre high-availability http http3 https ident imap imaps ipfs ipp ipp-client ips
ec irc ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiserver ku
be-control-plane kube-control-plane-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-services kube-
scheduler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-n
etwork llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcached minidlna mongodb mosh mountd mqtt mqtt-tls ms-w
bt mssql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 nmap-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-stor
ageconsole ovirt-vmconsole plex pmpcd pmpoxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus prometheus-node-expor
ter proxy-dhcp ps2link ps3netdrv ptp pulseaudio puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyn
cd rtsp salt-master samba samba-client samba-dc sane sip sips slp smtp smtp-submission smtps snmp snmptls snmptls-trap snmptra
p spideroak-lansync spotify-sync squid sssd ssh ssh-custom steam-streaming svdrp svn syncthing syncthing-gui syncthing-relay s
ynergy syslog syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsms vnc-server warpinator
wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans xdmcp xmpp-bos
h xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server zerotier
[root@server.aamishina.net services]#
```

Рис. 2.3: Новая служба в списке доступных служб

Новая служба отображается в списке доступных, но пока не активирована. Добавляем новую службу в FirewallD и просматриваем список активных служб (служба появилась). Перезагружаем правила межсетевого

экрана с сохранением информации о состоянии (рис. 2.4)

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

Рис. 2.4: Добавление новой службы и просмотр списка активных служб, сохранение информации о состоянии

## 2.2 Перенаправление портов

Организовываем переадресацию с порта 2022 на порт 22 на сервере, введя команду: `firewall-cmd --add-forward-port=port=2022:proto=tcp:toport=22`.

На клиенте пробуем получить доступ по SSH через порт 2022. Доступ получен (рис. 2.5), (рис. 2.6).

```
aamishina@client:~
[aamishina@client.aamishina.net ~]$ ssh -p 2022 aamishina@server.aamishina.net
The authenticity of host '[server.aamishina.net]:2022 ([192.168.1.1]:2022)' can't be established.
ED25519 key fingerprint is SHA256:9t38xEDGq4lcr5/Tsoa80ISQoztxdmZFH71zorIzzZ7g.
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.aamishina.net]:2022' (ED25519) to the list of known hosts.
ssh_dispatch_run_fatal: Connection to 192.168.1.1 port 2022: Broken pipe
[aamishina@client.aamishina.net ~]$
```

Рис. 2.5: Доступ по SSH к серверу через порт 2022 на клиенте

```
aamishina@server:~
[aamishina@client.aamishina.net ~]$ ssh -p 2022 aamishina@server.aamishina.net
aamishina@server.aamishina.net's password:
Activate the web console with: systemctl enable --now cockpit.socket

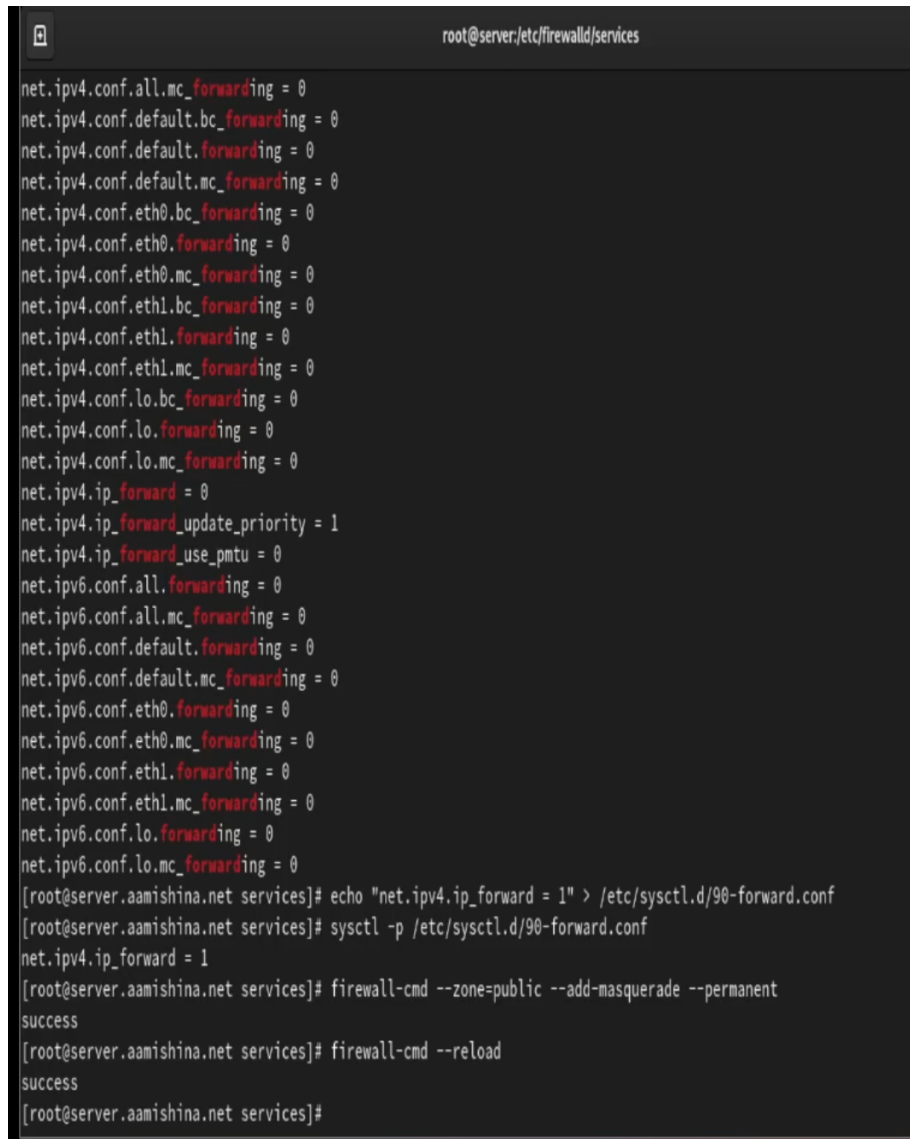
Last login: Tue Oct 15 20:48:05 2024 from 192.168.1.30
[aamishina@server.aamishina.net ~]$
```

Рис. 2.6: Доступ по SSH к серверу через порт 2022 на клиенте



## 2.3 Настройка Port Forwarding и Masquerading

На сервере просматриваем, активирована ли в ядре системы возможность перенаправления IPv4-пакетов. Включаем перенаправление пакетов на сервере. Включаем маскарading на сервере (рис. 2.7). Убеждаемся, что на клиенте доступен выход в интернет (веб-страницы в браузере загружаются успешно).



```
root@server:/etc/firewalld/services

net.ipv4.conf.all.mc_forwarding = 0
net.ipv4.conf.default.bc_forwarding = 0
net.ipv4.conf.default.forwarding = 0
net.ipv4.conf.default.mc_forwarding = 0
net.ipv4.conf.eth0.bc_forwarding = 0
net.ipv4.conf.eth0.forwarding = 0
net.ipv4.conf.eth0.mc_forwarding = 0
net.ipv4.conf.eth1.bc_forwarding = 0
net.ipv4.conf.eth1.forwarding = 0
net.ipv4.conf.eth1.mc_forwarding = 0
net.ipv4.conf.lo.bc_forwarding = 0
net.ipv4.conf.lo.forwarding = 0
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
net.ipv6.conf.all.forwarding = 0
net.ipv6.conf.all.mc_forwarding = 0
net.ipv6.conf.default.forwarding = 0
net.ipv6.conf.default.mc_forwarding = 0
net.ipv6.conf.eth0.forwarding = 0
net.ipv6.conf.eth0.mc_forwarding = 0
net.ipv6.conf.eth1.forwarding = 0
net.ipv6.conf.eth1.mc_forwarding = 0
net.ipv6.conf.lo.forwarding = 0
net.ipv6.conf.lo.mc_forwarding = 0

[root@server.aamishina.net services]# echo "net.ipv4.ip_forward = 1" > /etc/sysctl.d/90-forward.conf
[root@server.aamishina.net services]# sysctl -p /etc/sysctl.d/90-forward.conf
net.ipv4.ip_forward = 1
[root@server.aamishina.net services]# firewall-cmd --zone=public --add-masquerade --permanent
success
[root@server.aamishina.net services]# firewall-cmd --reload
success
[root@server.aamishina.net services]#
```

Рис. 2.7: Включение перенаправления пакетов и включение маскардинга

## 2.4 Внесение изменений в настройки внутреннего окружения виртуальной машины

На VM server переходим в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/` и копируем в соответствующие каталоги конфигурационные файлы. Создаем скрипт `firewall.sh` (рис. 2.8).

```
[root@server.aamishina.net services]# cd /vagrant/provision/server
[root@server.aamishina.net server]# mkdir -p /vagrant/provision/server/firewall/etc/firewalld/services
[root@server.aamishina.net server]# mkdir -p /vagrant/provision/server/firewall/etc/sysctl.d
[root@server.aamishina.net server]# cp -r /etc/firewalld/services/ssh-custom.xml /vagrant/provision/server/firewall/etc/firewalld/services/
[root@server.aamishina.net server]# cp -r /etc/sysctl.d/90-forward.conf /vagrant/provision/server/firewall/etc/sysctl.d/
[root@server.aamishina.net server]# cd /vagrant/provision/server
[root@server.aamishina.net server]# touch firewall.sh
[root@server.aamishina.net server]# chmod +x firewall.sh
[root@server.aamishina.net server]# vim firewall.sh
```

Рис. 2.8: Создание каталогов и копирование конфигурационных файлов, создание скрипта `firewall.sh`

Редактируем скрипт (рис. 2.9).

```
#!/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. 2.9: Редактирование `firewall.sh`

Для отработки созданного скрипта во время загрузки виртуальной машины `server` в конфигурационном файле `Vagrantfile` добавляем в разделе конфигурации для сервера следующую запись:

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

## 3 Выводы

В результате выполнения работы получены навыки настройки меж-  
сетевого экрана в Linux в части переадресации портов и настройки  
Masquerading.

## 4 Ответы на контрольные вопросы

1. Где хранятся пользовательские файлы firewalld?

- В firewalld пользовательские файлы хранятся в директории /etc/firewalld/.

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

- Для указания порта TCP 2022 в пользовательском файле службы, вы можете добавить строку в секцию port следующим образом:

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

- firewall-cmd --get-services

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

- Разница между трансляцией сетевых адресов (NAT) и маскарadingом (masquerading) заключается в том, что в случае NAT исходный IP-адрес пакета заменяется на IP-адрес маршрутизатора, а в случае маскарadingа используется маршрутизатора.

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

```
firewall-cmd --zone=public --add-port=4404/tcp --permanent
firewall-cmd --zone=public --add-forward-port=port=4404
:proto=tcp:toport=22:toaddr=10.0.0.10 --permanent
firewall-cmd --reload
```

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

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