

Отчёт по лабораторной работе №13

Фильтр пакетов

Амина Аджигалиева

Содержание

1	Цель работы	5
2	Ход выполнения работы	6
2.1	Управление брандмауэром с помощью firewall-cmd	6
2.1.1	Добавление службы VNC	8
2.1.2	Добавление VNC в постоянную конфигурацию	10
2.1.3	Добавление пользовательского порта в конфигурацию бранд- мауэра	11
2.2	Управление брандмауэром с помощью firewall-config	12
2.3	Самостоятельная работа	15
3	Контрольные вопросы	17
4	Заключение	19

Список иллюстраций

2.1	Отображение зон, служб и текущей конфигурации	7
2.2	Отображение текущей конфигурации	8
2.3	Добавление vnc-server во временную конфигурацию	9
2.4	Отображение текущей конфигурации	9
2.5	Добавление vnc-server в постоянную конфигурацию	10
2.6	применение настроек	11
2.7	Добавление порта 2022/tcp и проверка конфигурации	12
2.8	Включение служб http, https и ftp	13
2.9	Добавление порта 2022/udp	13
2.10	Применение изменений после перезагрузки	14
2.11	Добавление telnet через CLI	15
2.12	Разрешение imap, pop3, smtp через GUI	16

Список таблиц

1 Цель работы

Получить навыки настройки пакетного фильтра в Linux.

2 Ход выполнения работы

2.1 Управление брандмауэром с помощью firewall-cmd

Сначала были получены административные привилегии с использованием команды `su -`.

После перехода под пользователя `root` была определена зона брандмауэра по умолчанию:

```
firewall-cmd --get-default-zone
```

Значением оказалась зона **public**. Далее был выведен список доступных зон:

```
firewall-cmd --get-zones
```

Затем просмотрен перечень поддерживаемых служб, которые могут быть добавлены к правилам брандмауэра:

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

```
aradzhigalieva@aradzhigalieva:/home/aradzhigalieva
su: Authentication failure
aradzhigalieva@aradzhigalieva:~$ su
Password:
root@aradzhigalieva:/home/aradzhigalieva#
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --get-default-zone
public
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --get-zones
block dmz drop external home internal nm-shared public trusted work
root@aradzhigalieva:/home/aradzhigalieva# 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 a
seqnet audit ausweisapp2 bacula bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc b
itcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent civilization-iv civili
zation-v cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp dhcpv6 dhcpv6-client distcc dns
dns-over-qtcp dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server factorio fing
er foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client g
anglia-master git gssd grafana gre high-availability http http3 https ident imap imaps iperf2 iperf3 ipfs ipp ipp-client ip
sec irc ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiseve
r kube-control-plane kube-control-plane-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-service
s kube-scheduler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls l
ightning-network llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minecraft minidlna mndp mongodb mo
sh mountd mpd mqtt mqtt-tls ms-mbt mssql murmur mysql nbd nebula need-for-speed-most-wanted netbios-ns netdata-dashboar
nf nfs nfs3 nmap nmap-1813 ntp nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole plex pmcd pmproxy pm
webapi pmwebapis pop3 pop3s postgresql proxyx prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netdrv ptp pulseau
dio puppetmaster quassel radius radsec rdp redis redis-sentinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba sa
mba-client samba-dc sane settlers-history-collection sip sips slimevr slp smtp smtp-submission smtps snmp snmptls snmptls-t
rap snmptrap spideroak-lansync spotify-sync squid ssh statsrv steam-lan-transfer steam-streaming stellaris stronghold-
crusader stun stuns submission supertuxkart svdrp svn syncthing syncthing-gui syncthing-relay synergy syscomlan syslog sysl
og-tls telnet tentacle terraria tftp tile38 tinc tor-socks transmission-client turn turns upnp-client vdsms vnc-server vrrp
warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-host ws-discovery-tcp ws-discovery-
udp wsd wsd-http wsmn wsmns xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix-
server zabbix-trapper zabbix-web-service zero-k zerotier
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-services
cockpit dhcpv6-client ssh
root@aradzhigalieva:/home/aradzhigalieva#
```

Рис. 2.1: Отображение зон, служб и текущей конфигурации

Чтобы определить, какие службы разрешены в текущей зоне, была выполнена команда:

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

Далее были сопоставлены результаты вывода между:

```
firewall-cmd --list-all
```

и

```
firewall-cmd --list-all --zone=public
```

Обе команды вывели одинаковую информацию, что подтвердило использование зоны **public** по умолчанию.

```

root@aradzhigalieva: /home/aradzhigalieva
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all --zone=public
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:

```

Рис. 2.2: Отображение текущей конфигурации

2.1.1 Добавление службы VNC

Служба VNC была добавлена в конфигурацию временного времени выполнения:

```
firewall-cmd --add-service=vnc-server
```

После добавления была проверена активная конфигурация:

```
firewall-cmd --list-all
```

Сервис появился в списке разрешённых.


```

root@aradzhigalieva:/home/aradzhigalieva#
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --add-service=vnc-server
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh vnc-server
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva# █

```

Рис. 2.3: Добавление vnc-server во временную конфигурацию

Затем была перезапущена служба firewalld:

```
systemctl restart firewalld
```

После повторного выполнения команды для просмотра конфигурации выяснилось, что служба VNC исчезла:

```
firewall-cmd --list-all
```

Причина: изменение было внесено только во **временную конфигурацию (runtime)**. После перезапуска службы runtime-настройки сбрасываются.

```

root@aradzhigalieva:/home/aradzhigalieva#
root@aradzhigalieva:/home/aradzhigalieva# systemctl restart firewalld.service
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva# █

```

Рис. 2.4: Отображение текущей конфигурации

2.1.2 Добавление VNC в постоянную конфигурацию

Для сохранения изменений на диске была выполнена команда:

```
firewall-cmd --add-service=vnc-server --permanent
```

Проверка конфигурации показала, что VNC ещё не отображается, так как permanent-изменения не загружаются автоматически в runtime:

```
firewall-cmd --list-all
```

Чтобы применить сохранённые изменения, был выполнен перезапуск конфигурации:

```
firewall-cmd --reload
```

После повторного просмотра активной конфигурации:

```
firewall-cmd --list-all
```

Служба VNC появилась в списке разрешённых.

```
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --add-service=vnc-server --permanent
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva#
```

Рис. 2.5: Добавление vnc-server в постоянную конфигурацию

```
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --reload
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh vnc-server
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva#
```

Рис. 2.6: применение настроек

2.1.3 Добавление пользовательского порта в конфигурацию брандмауэра

Был добавлен пользовательский порт **2022/TCP** в постоянную конфигурацию межсетевого экрана:

```
firewall-cmd --add-port=2022/tcp --permanent
```

После добавления конфигурация была перезагружена:

```
firewall-cmd --reload
```

Затем выполнена проверка активной конфигурации:

```
firewall-cmd --list-all
```

В выводе появился новый открытый порт **2022/tcp**, что подтверждает успешное добавление.

```

root@aradzhigalieva:/home/aradzhigalieva#
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --add-port=2022/tcp --permanent
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --reload
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh vnc-server
  ports: 2022/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva# █

```

Рис. 2.7: Добавление порта 2022/tcp и проверка конфигурации

2.2 Управление брандмауэром с помощью firewall-config

Для работы с графической утилитой настройки брандмауэра был запущен интерфейс:

```
firewall-config
```

После запуска система запросила пароль пользователя, имеющего права управления службой.

В выпадающем меню **Configuration** был выбран режим **Permanent**, чтобы все внесённые изменения сохранялись в постоянной конфигурации.

Далее была выбрана зона **public**. На вкладке **Services** были отмечены службы **http**, **https** и **ftp**, что разрешило доступ к ним.

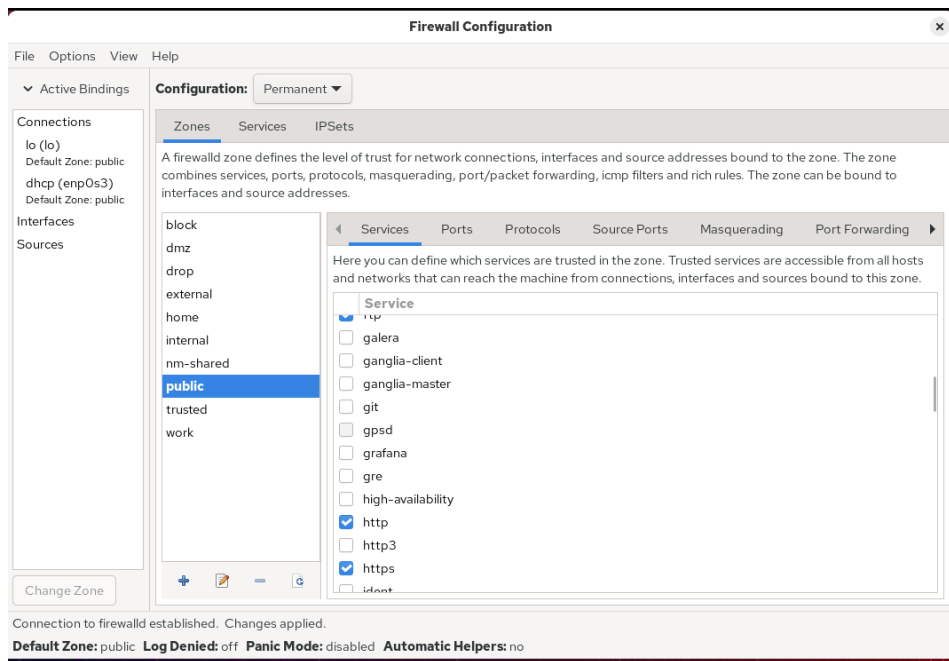


Рис. 2.8: Включение служб http, https и ftp

Затем на вкладке **Ports** был добавлен пользовательский порт:
порт **2022**, протокол **udp**.

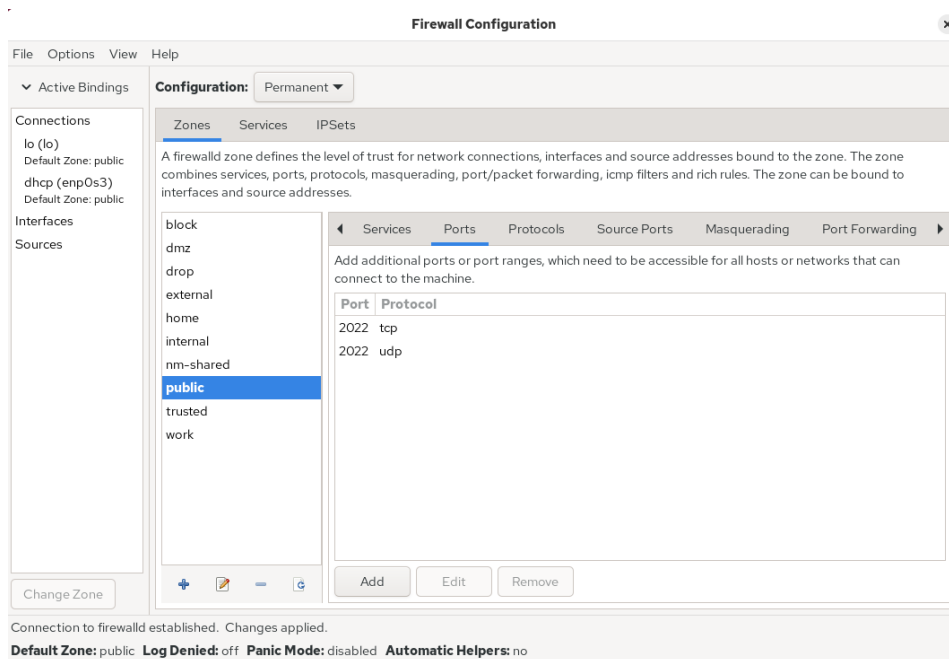


Рис. 2.9: Добавление порта 2022/udp

После закрытия утилиты была выполнена проверка текущей конфигурации:

```
firewall-cmd --list-all
```

На этом этапе изменения не отобразились, так как режим Permanent сохраняет настройки на диске, но не активирует их в конфигурации времени выполнения.

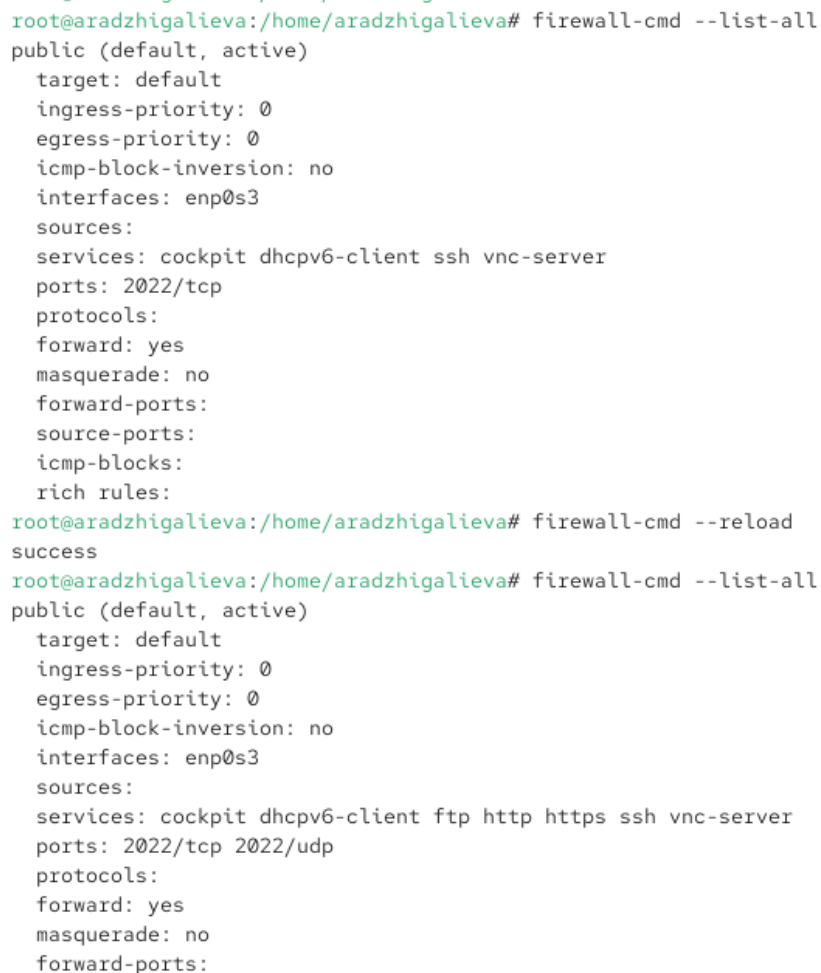
Для применения настроек была загружена конфигурация:

```
firewall-cmd --reload
```

Затем вывод команды:

```
firewall-cmd --list-all
```

показал, что открытые порты и разрешённые службы успешно добавлены.



```
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ssh vnc-server
  ports: 2022/tcp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --reload
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ftp http https ssh vnc-server
  ports: 2022/tcp 2022/udp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
```

Рис. 2.10: Применение изменений после перезагрузки

2.3 Самостоятельная работа

Необходимо создать конфигурацию брандмауэра, которая разрешает доступ к службам **telnet**, **imap**, **pop3** и **smtp**.

Служба telnet была добавлена через командную строку как постоянный параметр:

```
firewall-cmd --add-service=telnet --permanent
```

Затем были применены изменения:

```
firewall-cmd --reload
```

После перезагрузки служба появилась в списке разрешённых.

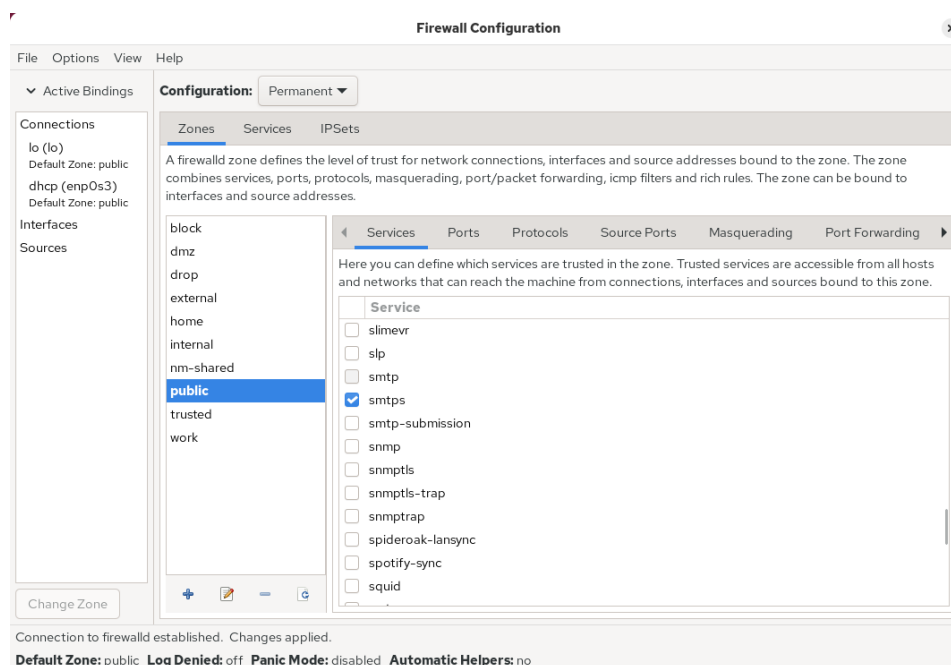


Рис. 2.11: Добавление telnet через CLI

Оставшиеся службы (**imap**, **pop3**, **smtp**) были добавлены через графическую утилиту **firewall-config**.

В режиме Permanent, на вкладке **Services**, были включены соответствующие службы.

```
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --add-service=telnet --permanent
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --reload
success
root@aradzhigalieva:/home/aradzhigalieva# firewall-cmd --list-all
public (default, active)
  target: default
  ingress-priority: 0
  egress-priority: 0
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ftp http https imap pop3 smtps ssh telnet vnc-server
  ports: 2022/tcp 2022/udp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
root@aradzhigalieva:/home/aradzhigalieva#
```

Рис. 2.12: Разрешение imap, pop3, smtp через GUI

Итоговая проверка:

```
firewall-cmd --list-all
```

показала, что **telnet**, **imap**, **pop3** и **smtp** добавлены и находятся в конфигурации активной зоны.

Настройки являются постоянными и будут применяться после перезагрузки системы.

3 Контрольные вопросы

1. **Какая служба должна быть запущена перед началом работы с firewall-config?**

Требуется запущенная служба firewalld. Без неё утилита не сможет управлять конфигурацией брандмауэра.

2. **Команда для добавления UDP-порта 2355 в зону по умолчанию?**

```
firewall-cmd --add-port=2355/udp --permanent
```

3. **Команда для отображения всей конфигурации брандмауэра во всех зонах?**

```
firewall-cmd --list-all-zones
```

4. **Команда для удаления службы vnc-server из текущей конфигурации?**

```
firewall-cmd --remove-service=vnc-server
```

5. **Команда для активации конфигурации, добавленной с опцией --permanent?**

```
firewall-cmd --reload
```

6. **Параметр для проверки, что конфигурация активна в текущей зоне?**

```
firewall-cmd --list-all
```

7. **Команда для добавления интерфейса eno1 в зону public?**

```
firewall-cmd --zone=public --add-interface=eno1 --permanent
```

8. **Если зона не указана при добавлении интерфейса, куда он будет добавлен?**

В зону, установленную по умолчанию (`firewall-cmd --get-default-zone`).

4 Заключение

В ходе выполнения лабораторной работы я разобралась с управлением межсетевым экраном в Linux с помощью утилит `firewall-cmd` и `firewall-config`.

Я изучила принципы работы с зонами, службами и портами, научилась добавлять и удалять сервисы и порты как во временную конфигурацию, так и в постоянную. Используя графический интерфейс и командную строку, я настроила доступ к различным службам, добавила пользовательский порт и убедилась, что изменения сохраняются после перезапуска.