

# Лабораторная работа №13

Основы администрирования операционных систем

---

Верниковская Е. А., НПИбд-01-23

29 ноября 2024

Российский университет дружбы народов, Москва, Россия

# Вводная часть

---

Получить навыки настройки пакетного фильтра в Linux.

### 1. Используя `firewall-cmd`:

- определить текущую зону по умолчанию
- определить доступные для настройки зоны
- определить службы, включённые в текущую зону
- добавить сервер VNC в конфигурацию брандмауэра

### 2. Используя `firewall-config`:

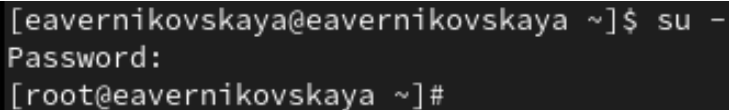
- добавить службы `http` и `ssh` в зону `public`
- добавить порт 2022 протокола UDP в зону `public`
- добавить службу `ftp`

### 3. Выполнить задание для самостоятельной работы

# Выполнение лабораторной работы

---

Запускаем терминала и получаем полномочия суперпользователя, используя *su* - (рис. 1)



```
[eavernikovskaya@eavernikovskaya ~]$ su -  
Password:  
[root@eavernikovskaya ~]#
```

**Рис. 1:** Режим суперпользователя

Определим текущую зону по умолчанию, введя: *firewall-cmd --get-default-zone* (рис. 2)

```
[root@eavernikovskaya ~]# firewall-cmd --get-default-zone  
public  
[root@eavernikovskaya ~]#
```

**Рис. 2:** Определение текущей зоны по умолчанию

Определим доступные зоны, введя: *firewall-cmd --get-zones* (рис. 3)

```
[root@eavernikovskaya ~]# firewall-cmd --get-zones
block dmz drop external home internal nm-shared public trusted work
[root@eavernikovskaya ~]#
```

**Рис. 3:** Определение доступных зон



# Управление брандмауэром с помощью firewall-cmd

Посмотрим службы, доступные на нашем компьютере, используя *firewall-cmd --get-services* (рис. 4)

```
[root@eavernikovskaya ~]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 bacul
a bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin
-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent cockpit collectd condor-collector cr
atedb ctdb dds dds-multicast dds-unicast dhcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-s
warm dropbox-lansync elasticsearch etcd-client etcd-server finger foreman foreman-proxy freeipa-4 freeipa-ldap free
ipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpsd grafana gre high-ava
lability http http3 https ident imap imaps ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdec
onnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane-se
cure kube-controller-manager kube-controller-manager-secure kube-nodeport-services kube-scheduler kube-scheduler-se
cure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llm
nr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minidlna mongodb mosh moudnt mqtt mqtt-tls ms-wbt ms
sql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovir
t-storageconsole ovirt-vmconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus prom
etheus-node-exporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio puppetmaster quassel radius rdp redis redis-sentin
el rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane sip sips slp smtp smtp-submission
smtps snmp snmptls snmptls-trap snmptrap spideroak-lansync spotify-sync squid ssdp ssh steam-streaming svdrp svn sy
ncthing syncthing-gui syncthing-relay synergy syslog syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmi
sion-client upnp-client vdsu vnc-server warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client
ws-discovery-tcp ws-discovery-udp wsman wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabb
ix-server zerotier
[root@eavernikovskaya ~]#
```

Рис. 4: Службы доступные на нашем компьютере

Определим доступные службы в текущей зоне: *firewall-cmd --list-services* (рис. 5)

```
[root@eavernikovskaya ~]# firewall-cmd --list-services
cockpit dhcpv6-client ssh
[root@eavernikovskaya ~]#
```

**Рис. 5:** Определение доступных служб в текущей зоне

## Управление брандмауэром с помощью firewall-cmd

Сравним результаты вывода информации при использовании команд *firewall-cmd --list-all* и *firewall-cmd --list-all --zone=public*. Результат одинаковый, так как в настоящее время зона *public* является активной зоной по умолчанию (рис. 6), (рис. 7)

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

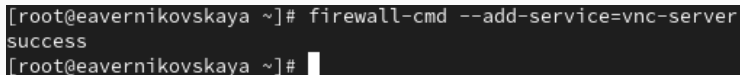
**Рис. 6:** Вывод команды *firewall-cmd --list-all*

## Управление брандмауэром с помощью firewall-cmd

```
[root@eavernikovskaya ~]# firewall-cmd --list-all --zone=public
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 7:** Вывод команды `firewall-cmd --list-all --zone=public`

Добавим сервер VNC в конфигурацию брандмауэра: *firewall-cmd --add-service=vnc-server* (рис. 8)



```
[root@eavernikovskaya ~]# firewall-cmd --add-service=vnc-server  
success  
[root@eavernikovskaya ~]#
```

**Рис. 8:** Добавление сервера VNC в конфигурацию брандмауэра

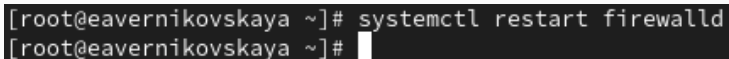
## Управление брандмауэром с помощью firewall-cmd

Проверим, добавился ли vnc-server в конфигурацию: *firewall-cmd --list-all* (рис. 9)

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 9:** Проверка добавления сервера VNC в конфигурацию

Перезапустим службу firewalld: *systemctl restart firewalld* (рис. 10)

A terminal window with a dark background. The prompt is [root@eavernikovskaya ~]#. The command systemctl restart firewalld has been entered. A white cursor is visible at the end of the second line, which also shows the prompt [root@eavernikovskaya ~]#.

```
[root@eavernikovskaya ~]# systemctl restart firewalld  
[root@eavernikovskaya ~]#
```

**Рис. 10:** Перезапуск службы firewalld

## Управление брандмауэром с помощью firewall-cmd

Проверим, есть ли vnc-server в конфигурации: *firewall-cmd --list-all*. Его нет, так как служба vnc-server не постоянная (рис. 11)

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 11:** Проверка наличия сервера VNC в конфигурации после перезапуска



Добавим службу vnc-server ещё раз, но на этот раз сделаем её постоянной, используя команду *firewall-cmd --add-service=vnc-server --permanent* (рис. 12)

```
[root@eavernikovskaya ~]# firewall-cmd --add-service=vnc-server --permanent  
success  
[root@eavernikovskaya ~]#
```

**Рис. 12:** Добавление постоянного сервера VNC в конфигурацию брандмауэра

## Управление брандмауэром с помощью firewall-cmd

Проверим наличие vnc-server в конфигурации: *firewall-cmd --list-all*. Мы увидим, что VNC-сервер не указан. Службы, которые были добавлены в конфигурацию на диске, автоматически не добавляются в конфигурацию времени выполнения (рис. 13)

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 13:** Проверка наличия постоянного сервера VNC в конфигурации

Перезагрузим конфигурацию firewalld и посмотрим конфигурацию времени выполнения: *firewall-cmd --reload* и *firewall-cmd --list-all* (рис. 14), (рис. 15)

```
[root@eavernikovskaya ~]# firewall-cmd --reload  
success  
[root@eavernikovskaya ~]#
```

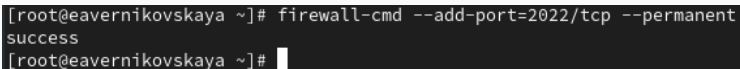
**Рис. 14:** Перезагрузка конфигурации firewalld (1)

## Управление брандмауэром с помощью firewall-cmd

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 15:** Просмотр конфигурации времени выполнения

Добавим в конфигурацию межсетевого экрана порт 2022 протокола TCP:  
*firewall-cmd --add-port=2022/tcp --permanent* (рис. 16)



```
[root@eavernikovskaya ~]# firewall-cmd --add-port=2022/tcp --permanent  
success  
[root@eavernikovskaya ~]#
```

**Рис. 16:** Добавление порта в конфигурацию

Затем снова перезагрузим конфигурацию firewalld: *firewall-cmd --reload* (рис. 17)

```
[root@eavernikovskaya ~]# firewall-cmd --reload  
success  
[root@eavernikovskaya ~]#
```

**Рис. 17:** Перезагрузка конфигурации firewalld (2)

## Управление брандмауэром с помощью `firewall-cmd`

И проверим, что порт добавлен в конфигурацию: `firewall-cmd --list-all` (рис. 18)

```
[root@eavernikovskaya ~]# firewall-cmd --list-all
public (active)
  target: default
  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@eavernikovskaya ~]#
```

**Рис. 18:** Проверка добавления порта в конфигурацию

Открываем терминал и под учётной записью нашего пользователя запускаем интерфейс GUI firewall-config: *firewall-config*. Служба отсутствует, и система предлагает нам её установить. Также при запуске вводим пароль пользователя с полномочиями управления этой службой (рис. 19), (рис. 20), (рис. 21)

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-config  
bash: firewall-config: command not found...  
Install package 'firewall-config' to provide command 'firewall-config'? [N/y] y
```

**Рис. 19:** Установка firewall-config



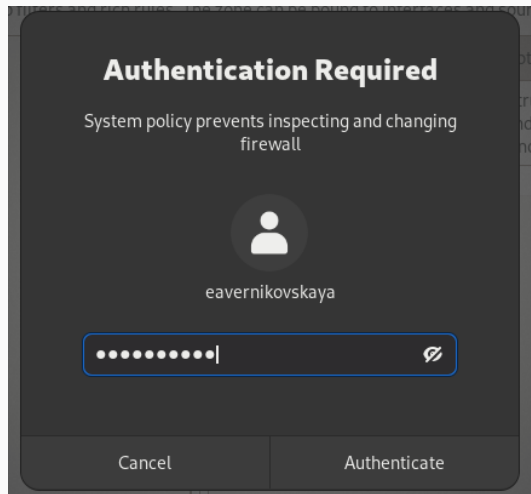


Рис. 20: Ввод пароля

# Управление брандмауэром с помощью firewall-config

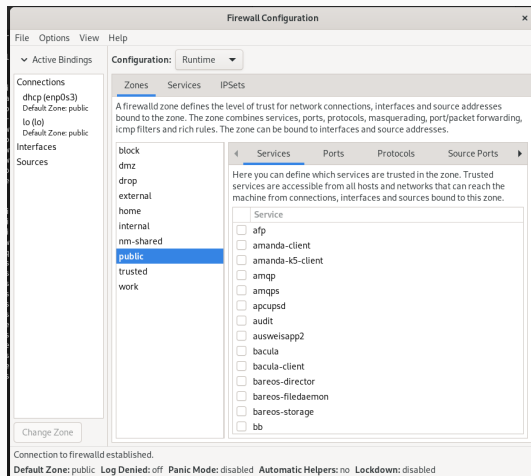
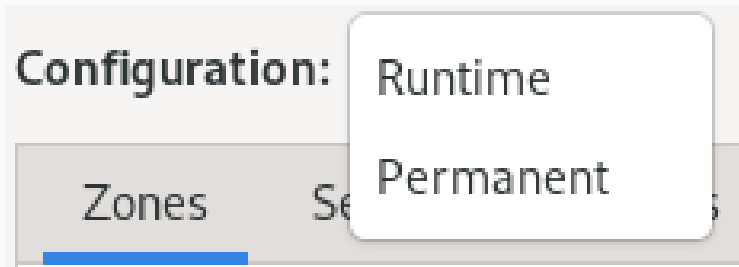


Рис. 21: Интерфейс GUI firewall-config

## Управление брандмауэром с помощью firewall-config

Нажимаем выпадающее меню рядом с параметром Configuration. Открываем раскрывающийся список и выбираем Permanent. Это позволит сделать постоянными все изменения, которые мы вносим при конфигурировании (рис. 22)



**Рис. 22:** Выбор Permanent

# Управление брандмауэром с помощью firewall-config

Выбираем зону public и отмечаем службы http, https и ftp, чтобы включить их (рис. 23)

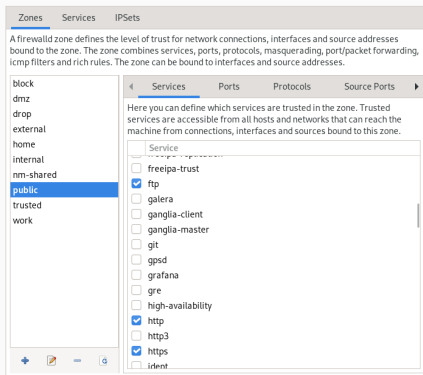
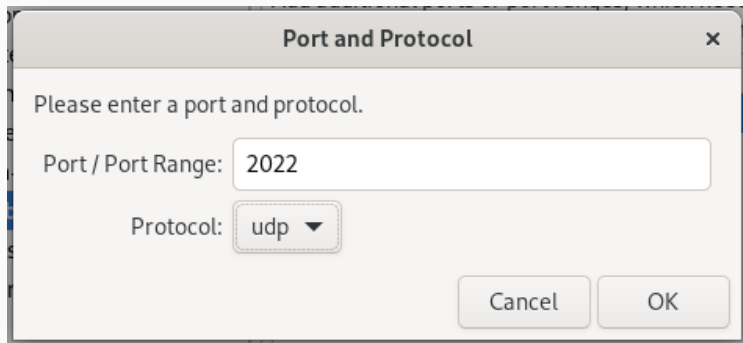


Рис. 23: Включение служб http, https и ftp

## Управление брандмауэром с помощью firewall-config

Выбираем вкладку Ports и на этой вкладке нажимаем Add. Вводим порт 2022 и протокол udp, нажимаем ОК, чтобы добавить их в список (рис. 24)



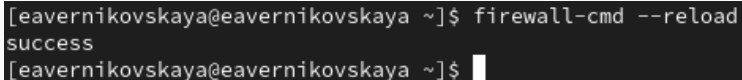
**Рис. 24:** Добавление порта

Закрываем утилиту firewall-config. В окне терминала вводим *firewall-cmd --list-all*. Изменения, которые мы только что внесли, ещё не вступили в силу. Это связано с тем, что мы настроили их как постоянные изменения, а не как изменения времени выполнения (рис. 25)

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --list-all
public (active)
  target: default
  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:
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 25: Проверка внесённых изменений (1)

Перезагрузим конфигурацию firewall-cmd: *firewall-cmd --reload* (рис. 26)



```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --reload  
success  
[eavernikovskaya@eavernikovskaya ~]$
```

**Рис. 26:** Перезагрузка конфигурации firewall-cmd (1)

Снова проверяем список доступных сервисов. Мы видим, что изменения вступили в силу (рис. 27)

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --list-all
public (active)
  target: default
  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:
  source-ports:
  icmp-blocks:
  rich rules:
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 27: Проверка внесённых изменений (2)



1. Надо создать конфигурацию межсетевого экрана, которая позволяет получить доступ к следующим службам:
  - telnet
  - imap
  - pop3
  - smtp
2. Сделать это как в командной строке (для службы telnet), так и в графическом интерфейсе (для служб imap, pop3, smtp)
3. Убедиться, что конфигурация является постоянной и будет активирована после перезагрузки компьютера

Сделаем службу telnet постоянной в командной строке: *firewall-cmd --add-service=telnet --permanent* (рис. 28)

```
[root@eavernikovskaya ~]# firewall-cmd --add-service=telnet --permanent  
success  
[root@eavernikovskaya ~]#
```

**Рис. 28:** Добавление постоянного telnet

Открываем интерфейс GUI firewall-config: *firewall-config* (рис. 29)

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-config
```

**Рис. 29:** Открытие интерфейса GUI firewall-config

Далее нажимаем выпадающее меню рядом с параметром Configuration. Открываем раскрывающийся список и выбираем Permanent. Выбираем зону public и отмечаем службы imap, pop3 и smtp, чтобы включить их (рис. 30)

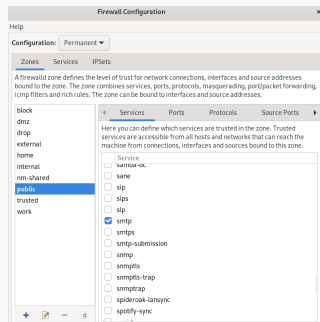


Рис. 30: Включение служб imap, pop3 и smtp

Перезагружаем конфигурацию firewall-cmd и проверяем, что изменения были применены (рис. 31), (рис. 32)

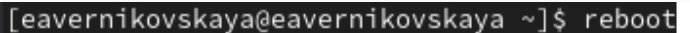
```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --reload  
success  
[eavernikovskaya@eavernikovskaya ~]$
```

**Рис. 31:** Перезагрузка конфигурации firewall-cmd (2)

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ftp http https imap pop3 smtp ssh telnet vnc-server
  ports: 2022/tcp 2022/udp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[eavernikovskaya@eavernikovskaya ~]$
```

**Рис. 32:** Проверка внесённых изменений (3)

Далее убедимся, что конфигурация является постоянной и будет активирована после перезагрузки компьютера (рис. 33), (рис. 34)



```
[eavernikovskaya@eavernikovskaya ~]$ reboot
```

**Рис. 33:** Перезагрузка ОС

```
[eavernikovskaya@eavernikovskaya ~]$ firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client ftp http https imap pop3 smtp ssh telnet vnc-server
  ports: 2022/tcp 2022/udp
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[eavernikovskaya@eavernikovskaya ~]$
```

**Рис. 34:** Проверка внесённых изменений после перезагрузки ОС



## Подведение итогов

---

В ходе выполнения лабораторной работы мы получили навыки настройки пакетного фильтра в Linux

1. Лабораторная работа №13 [Электронный ресурс] URL:  
[https://esystem.rudn.ru/pluginfile.php/2400747/mod\\_resource/content/4/014-firewall.pdf](https://esystem.rudn.ru/pluginfile.php/2400747/mod_resource/content/4/014-firewall.pdf)