

Отчёт по лабораторной работе 7

Расширенные настройки межсетевого экрана

Сейдалиев Тагиетдин Ровшенович

Содержание

1 Цель работы	5
2 Выполнение	6
2.1 Создание пользовательской службы firewalld	6
2.2 Перенаправление портов	7
2.3 Настройка Port Forwarding и Masquerading	8
2.4 Внесение изменений в настройки внутреннего окружения виртуальной машины	10
3 Заключение	12
4 Контрольные вопросы	13
4.1 1. Где хранятся пользовательские файлы firewalld?	13
4.2 2. Какую строку надо включить в пользовательский файл службы, чтобы указать порт TCP 2022?	13
4.3 3. Какая команда позволяет перечислить все службы, доступные в настоящее время на сервере?	13
4.4 4. В чём разница между NAT и masquerading?	14
4.5 5. Какая команда разрешает входящий трафик на порт 4404 и направляет его в службу SSH по адресу 10.0.0.10?	14
4.6 6. Какая команда используется для включения маскарадинга для всех пакетов, выходящих в зону public?	14

Список иллюстраций

2.1	Просмотр оригинального файла службы ssh-custom.xml	6
2.2	Редактирование файла службы с портом 2022	7
2.3	Получение списка служб и добавление новой службы	7
2.4	Добавление перенаправления порта через FirewallD	8
2.5	Подключение по SSH через порт 2022	8
2.6	Проверка параметров ядра на сервере	9
2.7	Проверка интернет-доступа с клиента	10
2.8	Содержимое скрипта firewall.sh	11

Список таблиц

1 Цель работы

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

2 Выполнение

2.1 Создание пользовательской службы firewalld

На сервере был создан модифицированный файл описания службы на основе стандартного ssh.xml.

После копирования файл был открыт и просмотрен.

```
[root@server.trseidaliev.net server]#  
[root@server.trseidaliev.net server]#  
[root@server.trseidaliev.net server]# cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/ssh-custom.xml  
[root@server.trseidaliev.net server]# cd /etc/firewalld/services/  
[root@server.trseidaliev.net services]# cat 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.trseidaliev.net services]#
```

Рис. 2.1: Просмотр оригинального файла службы ssh-custom.xml

В файле использован синтаксис XML:

- строка объявления указывает версию и кодировку документа;
- корневой элемент `<service>` описывает параметры службы FirewallD;
- тег `<short>` содержит краткое имя службы;
- элемент `<description>` даёт развёрнутое описание назначения службы;
- элемент `<port>` определяет номер порта и протокол.

В открытом файле изменён порт на значение 2022.

Описание службы скорректировано для отражения факта модификации.



```
ssh-custom.xml
/etc/firewalld/services

1 <?xml version="1.0" encoding="utf-8"?>
2 <service>
3   <short>SSH</short>
4   <description>Secure Shell port 2022</description>
5   <port protocol="tcp" port="2022"/>
6 </service>
```

Рис. 2.2: Редактирование файла службы с портом 2022

Просмотр списка стандартных служб показал отсутствие ssh-custom.

После перезагрузки правил FirewallD новая служба стала отображаться среди доступных, но не была активирована.

```
[root@server.trseidaliev.net services]# firewall-cmd --list-services
cockpit dhcp dhcpcv6-client dns http https ssh
[root@server.trseidaliev.net services]# firewall-cmd --add-service=ssh-custom
success
[root@server.trseidaliev.net services]# firewall-cmd --list-services
cockpit dhcp dhcpcv6-client dns http https ssh ssh-custom
[root@server.trseidaliev.net services]# firewall-cmd --add-service=ssh-custom --permanent
success
[root@server.trseidaliev.net services]# firewall-cmd --reload
success
[root@server.trseidaliev.net services]#
```

Рис. 2.3: Получение списка служб и добавление новой службы

Новая служба была добавлена в активные службы FirewallD.

После сохранения изменений и перезагрузки правил ssh-custom стала доступна постоянно.

2.2 Перенаправление портов

На сервере настроено перенаправление трафика с порта 2022 на стандартный порт 22.

Результат применения правила показан на скриншоте.

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

Рис. 2.4: Добавление перенаправления порта через FirewallD

На клиентской машине выполнено подключение по SSH через порт 2022.

Вход прошёл успешно.

```
[trseidaliev@client.trseidaliev.net ~]$ ssh -p 2022 trseidaliev@server.trseidaliev.net  
The authenticity of host '[server.trseidaliev.net]:2022 ([192.168.1.1]:2022)' can't be established.  
ED25519 key fingerprint is SHA256:n0w0vphoobfpKiXoFhmCpcTAKAVt01RYFK5Kxnw0XTU.  
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.trseidaliev.net]:2022' (ED25519) to the list  
of known hosts.  
trseidaliev@server.trseidaliev.net's password:  
Activate the web console with: systemctl enable --now cockpit.socket  
  
Last login: Wed Nov 26 16:46:11 2025  
[trseidaliev@server.trseidaliev.net ~]$  
[trseidaliev@server.trseidaliev.net ~]$  
logout  
Connection to server.trseidaliev.net closed.  
[trseidaliev@client.trseidaliev.net ~]$
```

Рис. 2.5: Подключение по SSH через порт 2022

2.3 Настройка Port Forwarding и Masquerading

На сервере был выполнен просмотр параметров, отвечающих за перенаправление пакетов.

По умолчанию значение net.ipv4.ip_forward оказалось равным нулю.

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

Рис. 2.6: Проверка параметров ядра на сервере

Создан файл в каталоге sysctl.d, содержащий параметр включения пересылки IPv4-пакетов.

После применения конфигурации forwarding был активирован.

В зоне public активирован masquerading.

После перезагрузки правил изменения вступили в силу.

На клиенте выполнена проверка — веб-страницы успешно открываются.

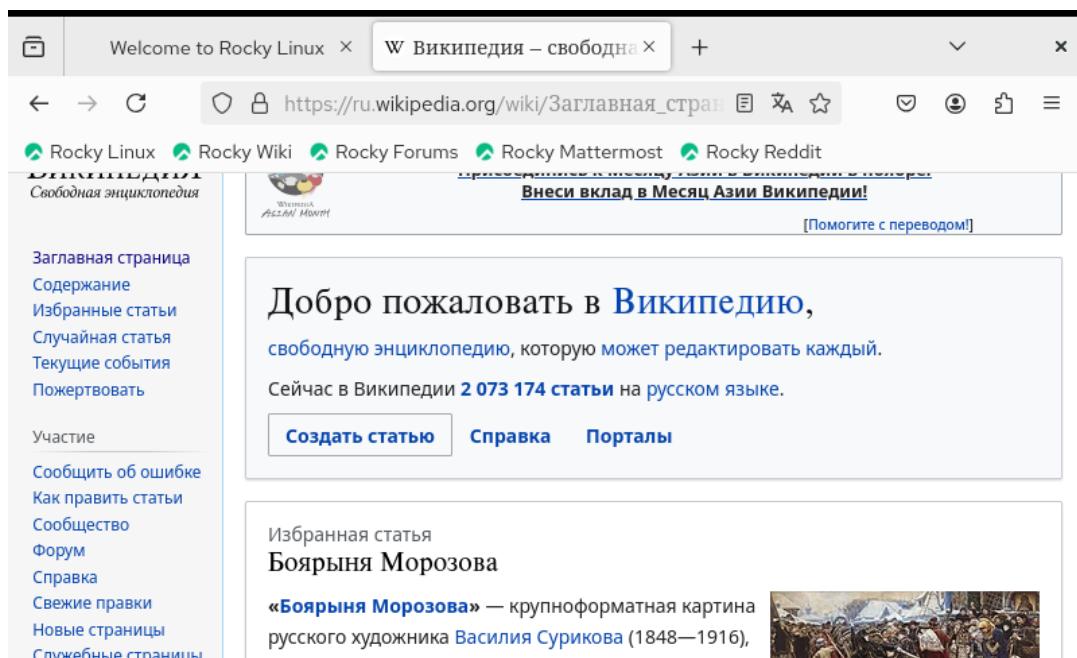


Рис. 2.7: Проверка интернет-доступа с клиента

2.4 Внесение изменений в настройки внутреннего окружения виртуальной машины

В каталоге /vagrant/provision/server создана структура подкаталогов для хранения конфигурационных файлов FirewallD и параметров sysctl.

В соответствующие директории были помещены подготовленные файлы ssh-custom.xml и 90-forward.conf.

В каталоге /vagrant/provision/server создан исполняемый сценарий firewall.sh, предназначенный для развёртывания сетевых настроек при provisioning.

Содержимое файла приведено на скриншоте.

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

Рис. 2.8: Содержимое скрипта firewall.sh

3 Заключение

В ходе работы:

- создана пользовательская служба firewalld на основе ssh.xml;
- изменён стандартный порт SSH на 2022 и добавлено перенаправление трафика на порт 22;
- активирована служба ssh-custom и перезагружены правила FirewallD;
- включено перенаправление IPv4-пакетов и настроен masquerading;
- проверено подключение по SSH через новый порт и доступ в Интернет с клиентской машины;
- подготовлены конфигурационные файлы и скрипт `firewall.sh` для автоматического провижининга.

4 Контрольные вопросы

4.1 1. Где хранятся пользовательские файлы firewalld?

Пользовательские файлы служб и зон firewalld располагаются в каталоге: /etc/firewalld/. Здесь находятся пользовательские версии конфигураций, перекрывающие системные файлы из /usr/lib/firewalld/.

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

Нужно добавить элемент порта: <port protocol="tcp" port="2022"/>

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

Для вывода списка всех доступных служб используется команда: `firewall-cmd --get-services`

4.4 4. В чём разница между NAT и masquerading?

NAT — общая технология трансляции сетевых адресов: подменяет IP-адреса и/или порты в пакетах при прохождении через маршрутизатор.

Masquerading — разновидность NAT, при которой исходящий трафик получает внешний IP-адрес интерфейса автоматически. Используется, когда внешний IP может изменяться и не задаётся вручную.

4.5 5. Какая команда разрешает входящий трафик на порт 4404 и перенаправляет его в службу SSH по адресу 10.0.0.10?

Команда перенаправления с разрешением порта: `firewall-cmd --add-forward-port=port=4404:proto=tcp:toaddr=10.0.0.10:toport=22`

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

Для активации masquerading применяется: `firewall-cmd --zone=public --add-masquerade`