

Лабораторная работа №7

Расширенные настройки межсетевого экрана

Сахно Никита НФИбд-02-23

Содержание

1	Цель работы	1
2	Задание	1
3	Выполнение лабораторной работы.....	2
3.1	Создание пользовательской службы firewalld.....	2
3.2	Настройка Port Forwarding и Masquerading.....	3
3.3	Внесение изменений в настройки внутреннего окружения виртуальной машины	5
4	Выводы.....	5
5	Контрольные вопросы.....	5

1 Цель работы

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

2 Задание

1. Настроить межсетевой экран виртуальной машины server для доступа к серверу по протоколу SSH не через 22-й порт, а через порт 2022.
2. Настроить Port Forwarding на виртуальной машине server.
3. Настроить маскарадинг на виртуальной машине server для организации доступа клиента к сети Интернет.
4. Написать скрипт для Vagrant, фиксирующий действия по расширенной настройке межсетевого экрана. Соответствующим образом внести изменения в Vagrantfile.

3 Выполнение лабораторной работы

3.1 Создание пользовательской службы firewalld

На основе существующего файла описания службы ssh создадим файл с собственным описанием и посмотрим содержимое файла службы.

```
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>
```

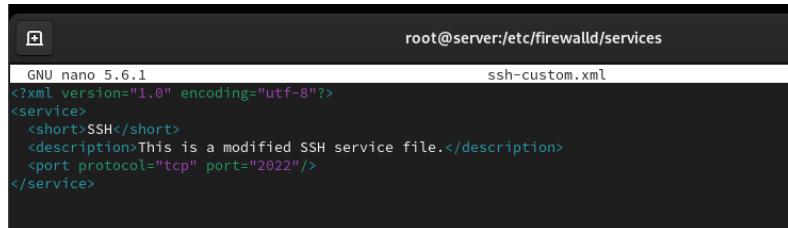
Создание файла с собственным описанием

В первой строчке указана версия xml и используемая кодировка - utf8. На второй строчке указан тег service, далее его тег-потомок short, внутри которого указан SSH. Затем указан тег description, внутри которого прописано описание протокола ssh, и указан протокол передачи порта tcp и номер порта.

Откроем файл описания службы на редактирование и заменим порт 22 на новый порт (2022):

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

В этом же файле скорректируем описание службы для демонстрации, что это модифицированный файл службы.



```
root@server:/etc/firewalld/services
GNU nano 5.6.1                                     ssh-custom.xml
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>SSH</short>
  <description>This is a modified SSH service file.</description>
  <port protocol="tcp" port="2022"/>
</service>
```

Отредактированный файл описания службы

Посмотрим список доступных FirewallD служб:

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

Новая служба ещё не отображается в списке.

```
RH-satellite-6 RH-Satellite-6-capsule aip amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 bacula bacula-client bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-mon cfengine checkmk-agent cockpit condor-collector cratedb ctdb dhcp dhcpcv6 dhcpcv6-client distcc dns dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-client finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ssl-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpm grafana gpm-high-availability http https ident imap imaps jnfs jpp ipp-client ipsec irc ircs iiscsi-target isns jellyfin jenkins kadmin kdeconnect kerberos kibana klogin 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-scheduler-secure kube-worker kubelet kubelet-readyonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-tcp llmnr-udp manage sieve matrix mdns memcache minidlna mongodb mqtt mqtt-tls ms-wbt mssql murmur mysql nbd netbios-nr netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole plex pmcd pmproxy pmweb bapi pmwebapis pop3 pop3s postgresql privoxy prometheus prometheus-node-exporter proxy-dhcp ps3netssrv ptp pulseaudio pupp etmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane sftp sips slp smtp smtp-submission snmp snmpd snmpd-trap snmptrap spiderOak-lansync spotify-sync squid ssdp ssh steam-streaming svdrp svn syncthing syncthing-gui synergy syslog syslog-tls telnet tentacle tftpd tile38 tinc tor-socks transmission-client upnp-client vdsm vnc-server wbem-https wbguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans xmpp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server zerotier
```

Список доступных FirewallD служб

Перегрузим правила межсетевого экрана с сохранением информации о состоянии и вновь выведем на экран список служб, а также список активных служб. Созданная служба отображается в списке доступных для FirewallD служб, но не активирована. Добавим новую службу в FirewallD и выведем на экран список активных служб:

Список FirewallD служб и добавление новой службы в FirewallD

Организуем на сервере переадресацию с порта 2022 на порт 22:

```
firewall-cmd --add-forward-port=port=2022:proto=tcp:toport=22
```

На клиенте попробуем получить доступ по SSH к серверу через порт 2022:

```
ssh -p 2022 nvsakhno@server.nvsakhno.net
```

К сожалению, в доступе мне было отказано.

3.2 Настройка Port Forwarding и Masquerading

На сервере посмотрим, активирована ли в ядре системы возможность перенаправления IPv4-пакетов пакетов:

```
sysctl -a | grep forward
```

```
net.ipv4.conf.all.bc_forwarding = 0
net.ipv4.conf.all.forwarding = 0
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
```

Проверка активации перенаправления IPv4-пакетов

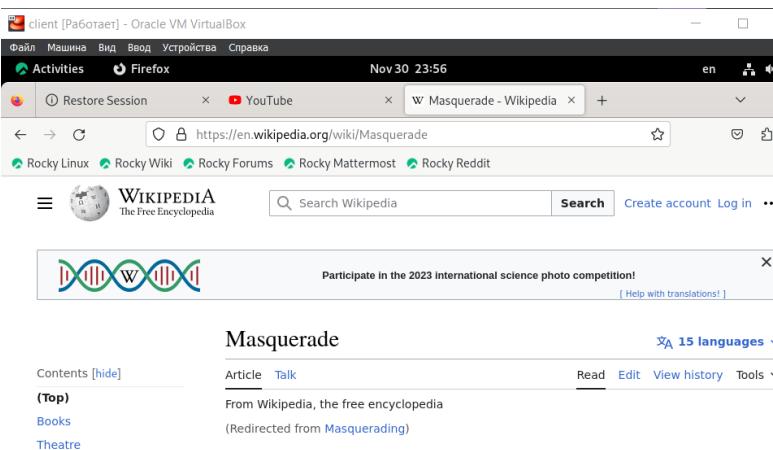
Включим перенаправление IPv4-пакетов на сервере. Включим маскарадинг на сервере и перезапустим систему:

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

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

Включение перенаправление IPv4-пакетов и маскарадинга на сервере

На клиенте проверим доступность выхода в Интернет.



Проверка доступности выхода в Интернет

Выход в Интернет на клиенте доступен.

3.3 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. В каталоге `/vagrant/provision/server` создадим файл `firewall.sh`.

```
[1] 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/services/
cp -r /etc/sysctl.d/90-forward.conf /vagrant/provision/server/firewall/etc/sysctl.d/
cd /vagrant/provision/server
touch firewall.sh
chmod +x firewall.sh
nano firewall.sh
```

Внесения изменений в настройки внутреннего окружения

Открыв его на редактирование, пропишите в нём следующий скрипт:

```
root@server:/vagrant/provision/server
GNU nano 5.6.1
#!/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
```

Редактирование файла

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

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

4 Выводы

В процессе выполнения данной лабораторной работы я получил навыки настройки межсетевого экрана в Linux в части переадресации портов и настройки Masquerading.

5 Контрольные вопросы

- Где хранятся пользовательские файлы firewalld?

```
/usr/lib/firewalld/services
```

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 --get-services
```

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

При маскарадинге вместо адреса отправителя(как делается это в NAT) динамически подставляется адрес назначенного интерфейса (сетевой адрес + порт).

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

```
sudo 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 --permanent
```