Настройка файервола (firewalld, iptables)

Введение

Файерволы являются важным компонентом безопасности в операционных системах, позволяя контролировать входящий и исходящий сетевой трафик на основе заданных правил. В НАЙС ОС используются два основных инструмента для настройки файервола: `firewalld` и `iptables`. В данной документации мы рассмотрим настройку и использование обоих инструментов, предоставив примеры команд и конфигураций.

Оба инструмента имеют свои особенности и случаи применения. `firewalld` предоставляет более высокоуровневый интерфейс и динамическое управление правилами, в то время как `iptables` предлагает более низкоуровневый и детализированный контроль. Мы начнем с рассмотрения `firewalld`, а затем перейдем к `iptables`.

Настройка firewalld

`firewalld` — это динамическая служба управления файерволом, которая поддерживает профили зон сети для определения уровня доверия к сетям и интерфейсам.

Установка firewalld

Для установки `firewalld` используйте `dnf`:

```
sudo dnf install firewalld
```

Запуск и управление службой firewalld

Запустите и настройте автозапуск службы `firewalld`:

```
sudo systemctl start firewalld
sudo systemctl enable firewalld
```

Основные команды firewalld

Для работы с `firewalld` используется утилита `firewall-cmd`. Вот некоторые основные команды:

Проверка состояния firewalld

```
sudo firewall-cmd --state
```

Просмотр активных зон

```
sudo firewall-cmd --get-active-zones
```

Просмотр правил для зоны

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

Добавление правила для разрешения сервиса

```
sudo firewall-cmd --zone=public --add-service=http --permanent
sudo firewall-cmd --reload
```

Удаление правила для сервиса

```
sudo firewall-cmd --zone=public --remove-service=http --permanent
sudo firewall-cmd --reload
```

Настройка зон firewalld

Зоны в `firewalld` определяют уровень доверия к сетям и интерфейсам. Каждая зона имеет свои правила и службы.

Переключение зоны интерфейса

```
sudo firewall-cmd --zone=home --change-interface=eth0 --permanent
sudo firewall-cmd --reload
```

Создание новой зоны

```
sudo firewall-cmd --permanent --new-zone=myzone
sudo firewall-cmd --reload
```

Добавление правил в новую зону

```
sudo firewall-cmd --zone=myzone --add-port=8080/tcp --permanent
sudo firewall-cmd --reload
```

Использование rich rules в firewalld

Rich rules предоставляют более детализированный контроль над правилами файервола.

Пример rich rule для разрешения доступа с конкретного IP-адреса

```
sudo firewall-cmd --zone=public --add-rich-rule='rule family="ipv4" source
address="192.168.1.100" service name="ssh" accept' --permanent
sudo firewall-cmd --reload
```

Удаление rich rule

```
sudo firewall-cmd --zone=public --remove-rich-rule='rule family="ipv4" source
address="192.168.1.100" service name="ssh" accept' --permanent
sudo firewall-cmd --reload
```

Настройка iptables

`iptables` предоставляет более низкоуровневый доступ к настройке правил файервола. Он позволяет контролировать сетевой трафик с использованием цепочек и таблиц.

Установка iptables

Для установки `iptables` используйте `dnf`:

```
sudo dnf install iptables iptables-services
```

Запуск и управление службой iptables

Запустите и настройте автозапуск службы `iptables`:

```
sudo systemctl start iptables
sudo systemctl enable iptables
```

Основные команды iptables

Команды `iptables` позволяют добавлять, изменять и удалять правила файервола.

Просмотр текущих правил

```
sudo iptables -L -v
```

Добавление правила для разрешения входящего трафика на порт 80

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

Удаление правила

```
sudo iptables -D INPUT -p tcp --dport 80 -j ACCEPT
```

Сохранение правил iptables

После добавления или изменения правил сохраните их, чтобы они применялись при перезапуске:

```
sudo service iptables save
```

Создание и управление цепочками iptables

Цепочки в `iptables` позволяют группировать правила для упрощения управления.

Создание пользовательской цепочки

```
sudo iptables -N mychain
```

Добавление правил в пользовательскую цепочку

```
sudo iptables -A mychain -p tcp --dport 8080 -j ACCEPT
```

Перенаправление трафика в пользовательскую цепочку

```
sudo iptables -A INPUT -p tcp -j mychain
```

Удаление пользовательской цепочки

Сначала удалите все правила из цепочки, затем удалите саму цепочку:

```
sudo iptables -F mychain
sudo iptables -X mychain
```

Использование iptables с NAT

Для настройки трансляции сетевых адресов (NAT) используйте таблицу `nat` в `iptables`.

Пример настройки маскарадинга

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Удаление правила NAT

```
sudo iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
```

Использование iptables для ограничения доступа

Для ограничения доступа к определенным портам или IP-адресам используйте соответствующие правила в `iptables`.

Блокировка доступа с конкретного IP-адреса

```
sudo iptables -A INPUT -s 192.168.1.100 -j DROP
```

Разрешение доступа только с конкретного IP-адреса

```
sudo iptables -A INPUT -s 192.168.1.100 -p tcp --dport 22 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

Автоматизация и скрипты

Для упрощения управления правилами файервола можно использовать скрипты и автоматизировать задачи.

Пример скрипта для настройки iptables

Создайте файл `iptables.rules` и добавьте в него следующие строки:

```
#!/bin/bash

# Очистка текущих правил
sudo iptables -F
sudo iptables -t nat -F

# Установка политики по умолчанию
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT ACCEPT

# Разрешение трафика на локальном интерфейсе
sudo iptables -A INPUT -i lo -j ACCEPT

# Разрешение установленного соединения
sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Разрешение SSH
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Сохранение правил
sudo service iptables save
```

Сделайте скрипт исполняемым и выполните его:

```
chmod +x iptables.rules
./iptables.rules
```

Автоматизация с помощью systemd

Создайте systemd службу для автоматического применения правил при загрузке системы.

Создание файла службы

Создайте файл `/etc/systemd/system/iptables.service` и добавьте в него следующие строки:

```
[Unit]
Description=Apply iptables rules
After=network.target

[Service]
Type=oneshot
ExecStart=/path/to/iptables.rules
```

```
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

Включение и запуск службы

```
sudo systemctl enable iptables.service
sudo systemctl start iptables.service
```

Журналирование и аудит

Журналирование действий файервола помогает отслеживать и анализировать сетевой трафик и выявлять потенциальные угрозы.

Включение журналирования в iptables

Для включения журналирования добавьте правило с целью `LOG`:

```
sudo iptables -A INPUT -p tcp --dport 80 -j LOG --log-prefix "HTTP traffic: "
```

Просмотр журналов

Журналы можно просмотреть с помощью команды `tail`:

```
sudo tail -f /var/log/messages
```

Заключение

Настройка файервола в НАЙС ОС с использованием `firewalld` и `iptables` предоставляет мощные инструменты для управления сетевым трафиком и обеспечения безопасности системы. `firewalld` предлагает более высокоуровневый и динамический подход к управлению правилами, в то время как `iptables` предоставляет детализированный и низкоуровневый контроль. Следуя приведенным инструкциям и примерам, вы сможете эффективно настроить и управлять файерволом в НАЙС ОС, обеспечивая защиту вашей системы от несанкционированного доступа и других угроз.