Добрый вечер!

Давайте заблокируем доступ к вашим ресурсам со стороны РКН, а заодно и начнем вести лог, хранящий информацию о попытках обращения к вашему ip со стороны этих ресурсов вот такого вида:

```
Mar 25 17:54:05 «ИМЯ_СЕРВЕРА» kernel: [3827476.520241] Blocked IP attempt: IN=eth0 OUT= MAC=«MAC_АДРЕС_УСТРОЙСТВА» SRC=«IP_из_blacklist» DST=«IP_вашего_сервера» LEN=60 TOS=0x08 PREC=0x40 TTL=51 ID=24312 DF PROTO=TCP SPT=47146 DPT=443 WINDOW=64240 RES=0x00 SYN URGP=0
```

Источником адресов для черного списка является **репозиторий** https://github.com/C24Be/AS_Network_List/ На текущий момент черный список содержит около 900 сетей, но регулярно обновляется.

Выполнив приведенные ниже команды, вы автоматизируете процесс внесения в правила брандмауэра адресов из черного списка - если в репозитории появился новый адрес, он добавится в список заблокированных и в лог, при следующем ручном запуске скрипта. Так же вы создадите расписание выполнения скрипта, чтобы вручную этого делать не приходилось.

Тестирование проводилось на Ubuntu 22.04, но все должно работать на большинстве версий Linux. Потенциально могут возникнуть проблемы с iptables, если ваш сервер использует nftables. В этом или иных случаях скормите текст скрипта нейросети и попросите его переделать под ваши условия. Либо поправьте руками. Цель этой инструкции популярно и детально объяснить как просто и быстро можно настроить черный список на вашем сервере, обходясь исключительно стандартными функциями.

Подготовка

Сконфигурируем под наши цели системный журнал:

```
sudo nano /etc/rsyslog.d/50-default.conf
```

Спустимся в самый низ и добавим

```
:programname, isequal, "sudo" ~
:msg, contains, "Blocked IP attempt: " /var/log/blacklist/blacklist.log
& ~
```

Закроем с сохранением (Ctrl+X затем Y и Enter)

Разрешим запись в каталог, в котором будут расположены файлы скрипта и логи

```
sudo chown syslog:adm /var/log/blacklist
sudo chmod 0755 /var/log/blacklist
```

Перезапустим службу rsyslog

```
sudo service rsyslog restart
```

Создание и выполнение скрипта

Создадим пустой файл скрипта

```
sudo nano /var/log/blacklist/blacklist_updater.sh
```

Вставим в него текст скрипта

```
#!/bin/bash

# Создаем каталог, если он еще не существует
sudo mkdir -p /var/log/blacklist/

# Переименовываем существующий файл blacklist.txt в old_blacklist.txt
sudo mv /var/log/blacklist/blacklist.txt /var/log/blacklist/old_blacklist.txt

# Копируем файл blacklist.txt из источника по ссылке
if ! sudo wget -O /var/log/blacklist/blacklist.txt https://github.com/C24Be/AS_Network_List/raw/main/
blacklists/blacklist.txt; then
    echo "Не удалось загрузить новый черный список. Оставляем старый список без изменений."
     echo "$(date +"%Y-%m-%d %H:%M:%S") - Не удалось загрузить новый черный список. Оставляем старый список без
изменений." >> /var/log/blacklist/blacklist_updater.log
    exit 1
fi

# Пути к файлам с IP-адресами
OLD_IP_FILE="/var/log/blacklist/old_blacklist.txt"
NEW_IP_FILE="/var/log/blacklist/blacklist.txt"

# Считываем IP-адреса из старого файла
old_addresses=()
while IFS= read -r ip || [[ -n "$ip" ]]; do
    old_addresses+=("$ip")
done < "$OLD_IP_FILE"

# Считываем IP-адреса из нового файла
new_addresses=()
while IFS= read -r ip || [[ -n "$ip" ]]; do
    new_addresses+=("$ip")
done < "$NEW_IP_FILE"

# Добавляем новые адреса и удаляем старые из правил
added=0
removed=0
for addr in "${new_addresses[@]}"; do
    if ! sudo iptables -t raw -C PREROUTING -s "$addr" -j DROP &>/dev/null; then
        sudo iptables -t raw -A PREROUTING -s "$addr" -j LOG --log-prefix "Blocked IP attempt: "
        sudo iptables -t raw -A PREROUTING -s "$addr" -j DROP
        ((added++))
    fi
done

for addr in "${old_addresses[@]}"; do
    if ! grep -q "$addr" "$NEW_IP_FILE"; then
        sudo iptables -t raw -D PREROUTING -s "$addr" -j LOG --log-prefix "Blocked IP attempt: "
        sudo iptables -t raw -D PREROUTING -s "$addr" -j DROP
        ((removed++))
    fi
done

# Сохраняем правила брандмауэра в файл
sudo sh -c "iptables-save > /etc/iptables/rules.v4"

# Выводим информацию о добавленных и удаленных адресах
echo "Добавлено адресов в черный список: $added"
echo "Удалено адресов из черного списка: $removed"

# Добавляем запись в лог-файл
echo "$(date +"%Y-%m-%d %H:%M:%S") - Добавлено адресов в черный список: $added, Удалено адресов из черного
списка: $removed" >> /var/log/blacklist/blacklist_updater.log
```

Закроем с сохранением (Ctrl+X затем Y и Enter)

Разрешим его запуск

```
sudo chmod +x /var/log/blacklist/blacklist_updater.sh
```

Запустим скрипт в первый раз, исполнение займет некоторое время:

```
sudo /var/log/blacklist/blacklist_updater.sh
```

Вы должны получить ответ вида:

```
Добавлено адресов в черный список: 898
Удалено адресов из черного списка: 0
```

Все работает. Разберем подробнее, что мы имеем.

`/var/log/blacklist/` теперь содержит файлы:

`blacklist_updater.sh` - сам скрипт, на случай если понадобится внести изменения

`blacklist_updater.log` - хранит лог результатов выполнения скрипта и фиксирует время

`blacklist.log` - тот самый лог, который хранит информацию о попытках обращения с заблокированных адресов к вашему серверу. Содержит порт, мак-адрес, свойства пакета и другую полезную информацию

`blacklist.txt` и `old_blacklist.txt` - копия файла из репозитория, а так же предыдущая версия этого файла, для отслеживания изменений в списке адресов

Проверить, что все адреса теперь в черном списке можно командой:

```
sudo nano /etc/iptables/rules.v4
```

Если ранее вы вносили свои правила, они так же должны остаться.

Если по какой-то причине вам надо очистить всю цепочку PREROUTING в таблице raw, выполните следующую команду, которая **удалит все** добавленные адреса:

```
sudo iptables -t raw -F PREROUTING
```

Создаем расписание выполнения

Можно запускать скрипт вручную, но лучше установить расписание, по которому он будет выполняться, например каждый день в определенное время.
Для начала узнаем время вашего сервера, оно может отличаться от времени вашего часового пояса:

```
date
```

Создаем расписание, время используем то, что у сервера, для этого выполняем:

```
sudo crontab -e
```

При первом запуске вероятно спросит каким редактором открыть, выбираем привычный nano.

В конце файла добавляем текст

```
0 9 * * * /bin/bash /var/log/blacklist/blacklist_updater.sh
```

Это создаст расписание на выполнение данного скрипта каждый день в 9 утра по времени сервера. Укажите условия, которые удобнее вам.

Вот что означает каждое поле:

- 0 - минуты (от 0 до 59)
- 9 - часы (от 0 до 23)
- * - день месяца (от 1 до 31)
- * - месяц (от 1 до 12)
- * - день недели (от 0 до 7, где 0 и 7 - воскресенье)

Закроем с сохранением (Ctrl+X затем Y и Enter)

Теперь у вас есть ежедневно обновляемый черный список и лог обращений от адресов из этого списка.