SkyDNS Zapret ISP

Сервис для фильтрации интернет ресурсов по требованиям РКН.

Описание SkyDNS Zapret ISP

Введение

SkyDNS Zapret ISP - система фильтрации трафика, разрабатываемая компанией SkyDNS. Сервис может ограничивать доступ к конкретным URL'ам, доменным именам или IP-адресам. Присутствует возможность создавать свои правила фильтрации, используя инструменты командной строки (см. zi-ctl filter) или web-интерфейс (см. Создание нового правила).

Система полностью соответствует всем требованиям Роскомнадзора по блокировке запрещенных ресурсов, входящих в единый реестр. SkyDNS Zapret ISP входит в список фильтров, рекомендованных Роскомнадзором к использованию.

В состав системы входит утилита SkyDNS Zapret Check, которая позволяет осуществлять проверку качества фильтрации самостоятельно.

Если у Вас возникнут какие-либо замечания по документации, просьба сообщить об этом специалистам технической поддержки **support@skydns.ru**. В кратчайшие сроки, Вы получите ответ, а документация будет оперативно обновлена.

Компоненты

Система фильтрации состоит из следующих компонентов:

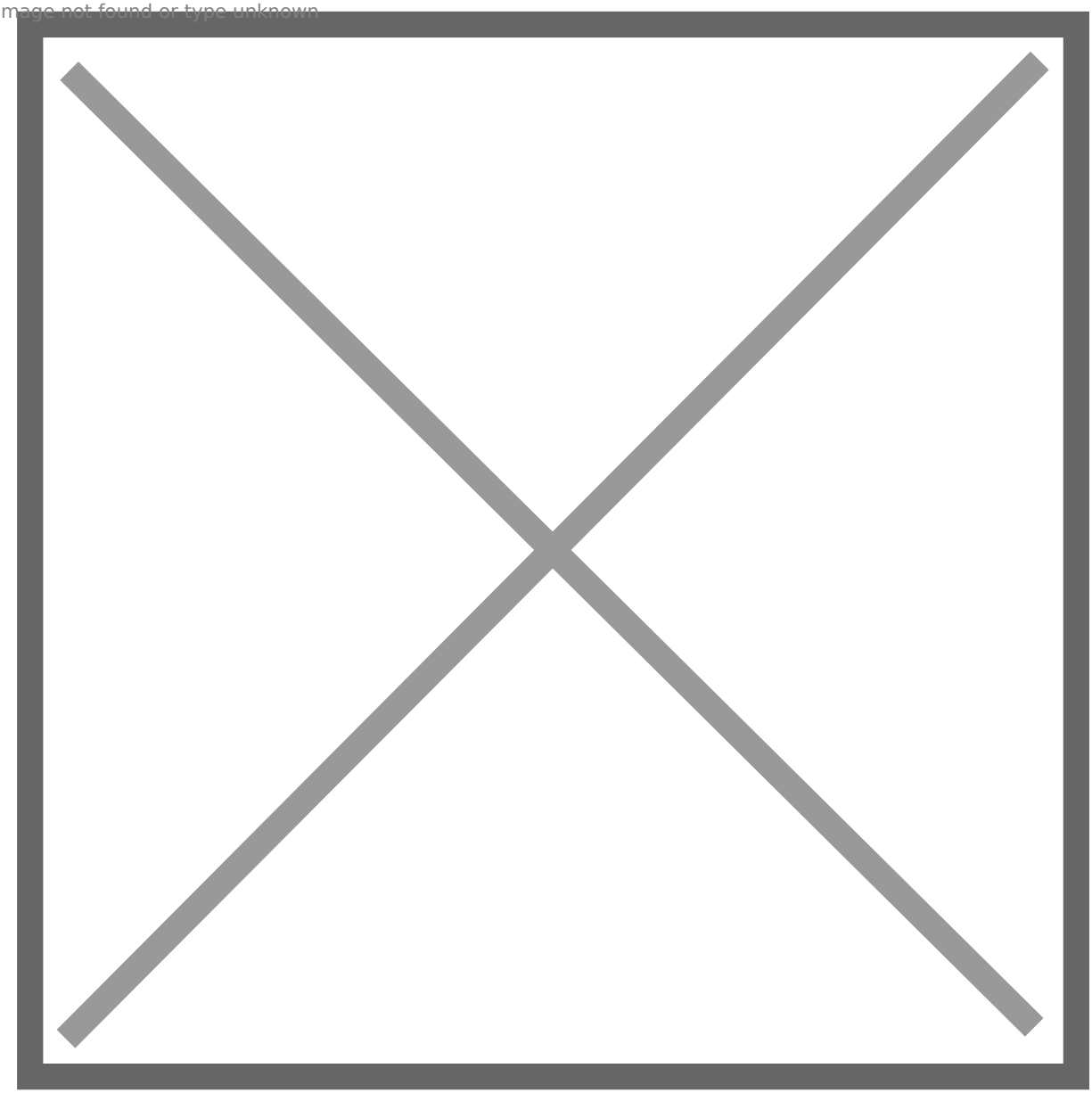
- Модуль загрузки реестра запрещенных ресурсов с eais.rkn.gov.ru.
- Модуль загрузки списка URL, подготовленного SkyDNS, на основе списка Министерства юстиции РФ.
- Модуль преобразования данных для системы фильтрации.
- Рекурсивный DNS-сервер Unbound. Выполняет функции фильтрующего и обычного DNS-серверов.
- Модуль асинхронного разрешения доменов в IP-адреса.
- Модуль динамической маршрутизации. Выполняет анонсирование маршрутов на маршрутизатор (необходимость использования данного модуля зависит от схемы внедрения решения в существующую сеть).
- Система фильтрации по URL, которая состоит из прокси-сервера Squid и external acl для него.
- Набор правил iptables и списки ipset выполняют блокировку в том случае, когда доступ ограничивается по IP-адресу.
- Страница блокировки, соответствующая требованиям Роскомнадзора.

- Web-интерфейс администратора системы (см. Web-интерфейс).
- База данных.
- zi-ctl - консольное меню, позволяющее производить управление системой (см. Команды zi-ctl).

Алгоритмическая схема работы

Диаграмма ниже демонстрирует, как будет выглядеть процесс http(s) запросов при внедрении SkyDNS Zapret ISP.

Image not found or type unknown

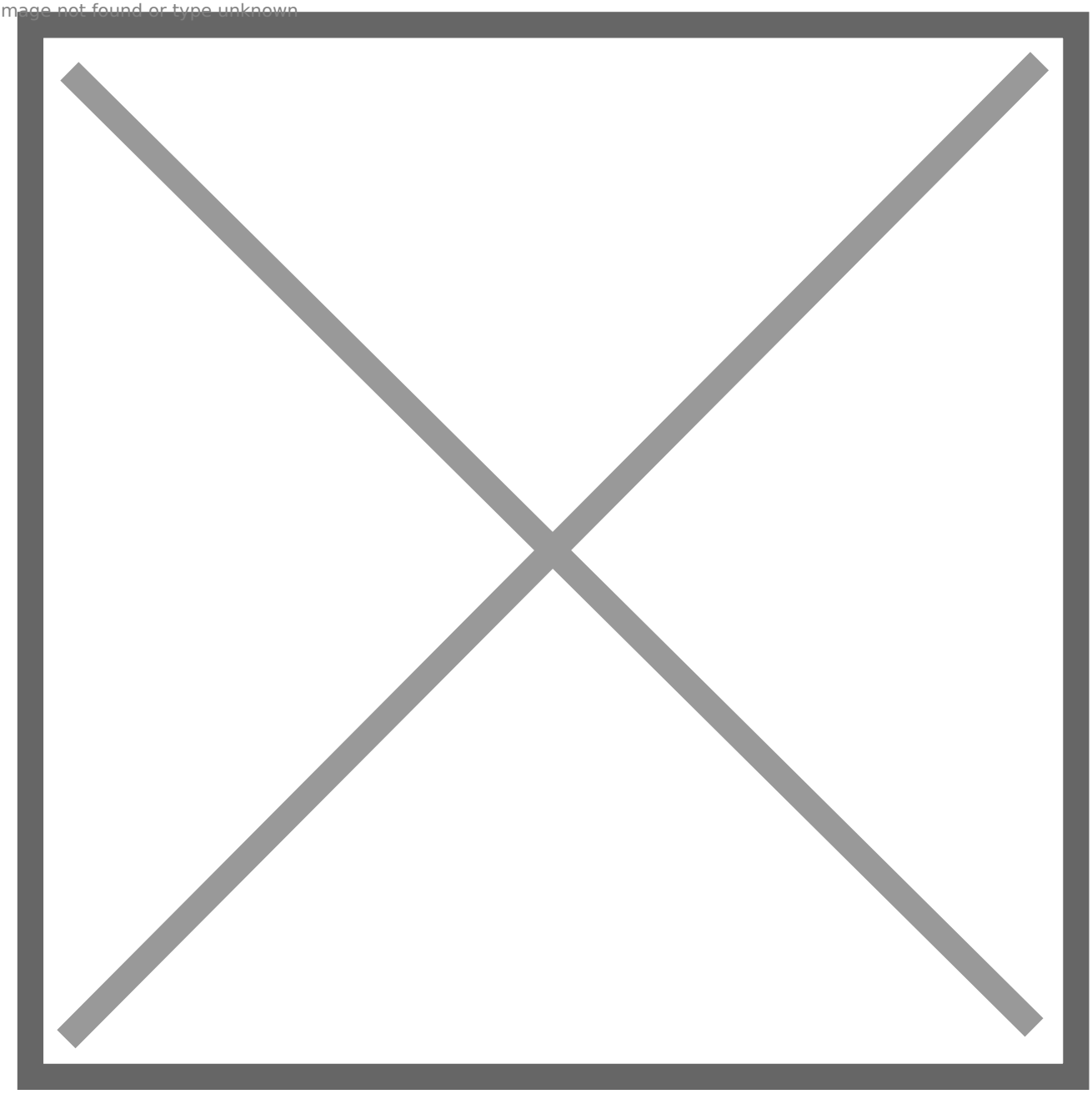


Путь трафика при использовании SkyDNS Zapret ISP

Диаграмма ниже демонстрирует, как пойдёт трафик при использовании системы фильтрации.

Схема подключения - с внутренним и пограничным маршрутизаторами (см. Схемы подключения SkyDNS Zapret ISP в сеть).

Image not found or type unknown



Для получения информации по работе ACL, читайте Проверки в ACL.

Системные требования SkyDNS Zapret ISP

Минимальные требования к серверу

- ОС Debian 8 64-bit
- 8 Гб RAM
- 1,6 ГГц 4 ядра
- 20 Гб свободного места на диске

Рекомендованные системные требования

Предлагаем Вам ознакомиться с рекомендованными системными требованиями в зависимости от объёмов трафика:

Процессор

Общая ширина канала	Количество ядер CPU	Пример процессора
до 20 Гб/с	8-12 ядер	Intel(R) Xeon(R) CPU X3430 2.40 ГГц
до 30 Гб/с	12-16 ядер	Intel(R) Xeon(R) CPU E5-2603 v4 1.70 ГГц
до 40 Гб/с	16-24 ядра	Intel(R) Xeon(R) CPU E5450 3.00 ГГц
до 50 Гб/с	24-32 ядра	Intel(R) Xeon(R) CPU X5560 2.80 ГГц

Объём оперативной памяти

Минимальный объём оперативной памяти, требуемой SkyDNS Zapret ISP составляет ~2.2 Гб - столько потребляет система без нагрузки.

Общая ширина канала	Объём оперативной памяти
до 20 Гб/с	12-18 Гб
до 30 Гб/с	18-24 Гб
до 40 Гб/с	24-48 Гб
до 50 Гб/с	48-72 Гб

Жёсткий диск

Объём дискового пространства, потребляемый системой, напрямую зависит от объёма логов. Рекомендуется выделить для SkyDNS Zapret ISP 50 Гб свободного места.

> Результат был получен в лабораторных условиях на последовательно-равномерной нагрузке. В интернете результат может отличаться.

Тест производительности

Для замеров показателей производительности в сеть одного из провайдеров был установлен сервер фильтрации.

Характеристики сервера:

- Процессор - Intel(R) Xeon(R) CPU E5450 3.00GHz.
- Объём оперативной памяти - 16 Gb.
- На сервер был установлен только SkyDNS Zapret ISP вместе с необходимыми пакетами.
- Использовалась стандартная конфигурация Squid.

Ниже приводится таблица с показателями.

Соотношение между объёмом трафика и загрузкой процессора.

Объём трафика, поступающей на сервер	Показатель Load average
400 Мб/с	2; 2.1; 2
1 Гб/с	3.4; 3.8; 3.5
1.5 Гб/с	5; 5.4; 5.2

Потребление оперативной памяти ~ 6 Gb.

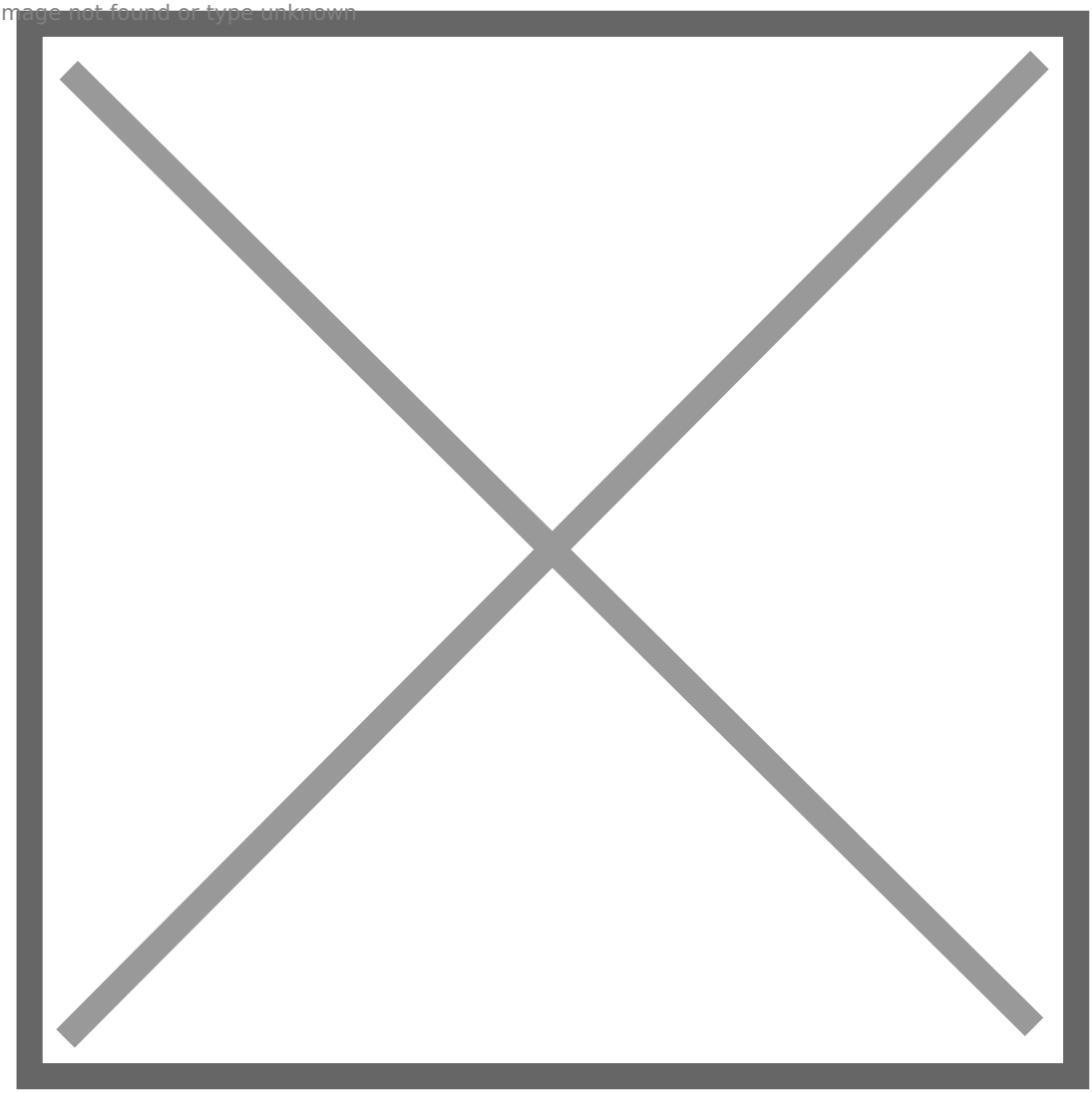
Виртуализация

Использование виртуализации не рекомендуется. Если вы все же используете виртуализацию, то убедитесь, что виртуальная машина получает необходимое для нормального функционирования количество ресурсов согласно системным требованиям, необходимым для обработки Вашего трафика. Так как менеджер виртуальных машин является программным обеспечением, параметры сборки, а так же качество написанных алгоритмов напрямую влияет на производительность. Компания SkyDNS не берётся предсказывать производительность для отдельно взятых менеджеров виртуальных машин.

Схемы подключения SkyDNS Zapret ISP в сеть

Схема при использовании динамической маршрутизации с внутренним и пограничным маршрутизаторами

Image not found or type unknown



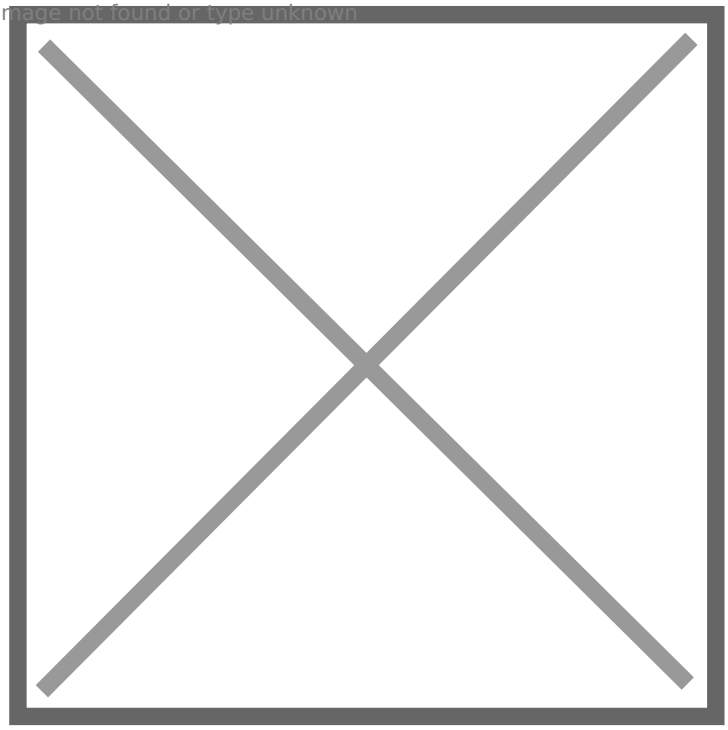
SkyDNS Zapret ISP использует exabgp для анонсирования и удаления маршрутов при использовании схемы с динамической маршрутизацией. Демон bgpd, входящий в Quagga, получает маршруты от демона exabgp по BGP. Оба запущены на loopback интерфейсе. Демон ospfd или bgpd (в зависимости от того, какой протокол Вы решите использовать для взаимодействия с маршрутизатором), входящий в Quagga, устанавливает полученные маршруты во внутренний маршрутизатор.

Анонсируемые SkyDNS Zapret ISP во внутренний маршрутизатор маршруты не должны попадать на пограничный маршрутизатор, иначе фильтруемый трафик зациклится.

Если пограничный маршрутизатор выполняет функции NAT, то убедитесь, что для сервера фильтрации SkyDNS Zapret ISP не применяются какие-либо NAT ограничения, которые используются в администрируемой сети, например, ограничение на количество сессий. Через систему фильтрации будет проходить трафик множества клиентов, поэтому к ней нельзя применять ограничения, которые применяются к одному клиенту.

Схема при использовании динамической маршрутизации с одним маршрутизатором

Image not found or type unknown



При такой схеме необходимо настроить два VLAN - на vlan100 настроить OSPF или BGP: в зависимости от того, какой из протоколов Вы планируете использовать для анонсирования маршрутов (Динамическая маршрутизация); а через vlan200 настроить выход в интернет. На маршрутизаторе настроить PBR (Policy Based Routing), чтобы запросы из vlan200 не маршрутизировались во vlan100, а уходили в интернет.

Рассмотрим пример настройки PBR на маршрутизаторах Cisco с использованием route-map:

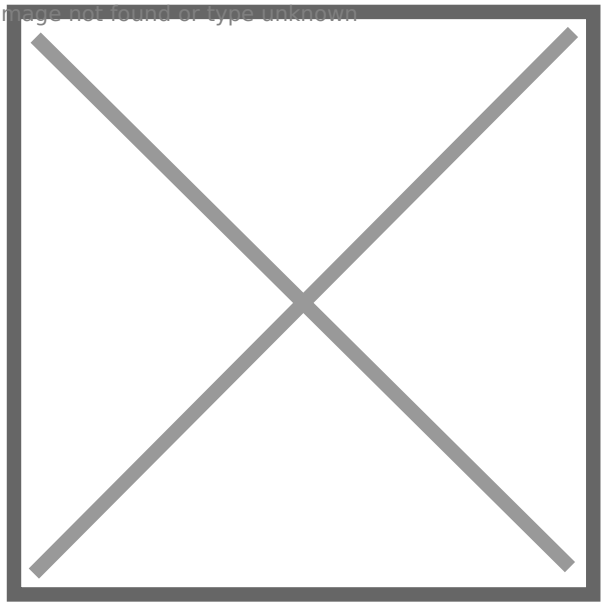
```
route-map PBR permit 10
 match ip address vlan200 # Какой трафик маршрутизировать
 set ip next-hop 10.0.1.1 # Куда отправлять маршрутизируемый трафик
```

После создания route-map необходимо применить её на интерфейсе, принимающем трафик:

```
router(config-if)# ip policy route-map PBR
```

Схема при использовании статической маршрутизации

Image not found or type unknown



В SkyDNS Zapret ISP есть возможность выгрузить список IP-адресов, в которые разрешаются домены запрещенных ресурсов. Для этого используется команда zi-ctl routes.

Вы можете написать свой скрипт загрузки маршрутов на роутер и добавить правило в cron. Примеры скриптов находятся в директории /usr/share/skydns-zi/examples/ (Статическая маршрутизация).

Правило следует исполнять раз в шесть минут. Время выбрано исходя из периодичности вызова команды создания stub-зон (zi-ctl create-zones) и удаления неактуальных маршрутов (с истёкшим ttl zi-ctl delete-expired).

> Если Вы используете такой тип подключения, то в конфигурационном файле, Вам необходимо установить значение: `resolver:`
> `dynamic-routing: false`

Использование какой-либо другой схемы

Вы можете использовать какую-либо другую схему включения SkyDNS Zapret ISP в свою сеть. В это случае Вам необходимо будет обеспечить маршрутизацию трафика к IP-адресам запрещенных ресурсов на сервер SkyDNS Zapret ISP.

Доступные средства:

1. Quagga, поддерживающая следующие протоколы: OSPF, BGP, IS-IS, RIP. Вы можете настроить маршрутизацию между Quagga и Вашим маршрутизатором по одному из перечисленных протоколов.

2. Используя команду zi-ctl routes, Вы можете получить список актуальных IP-адресов, после чего передавать их роутеру любым удобным для Вас способом.

Google Global Cache (GGC)

Если у вас установлена одна или несколько нод GGC, убедитесь, что IP-адрес, с которого SkyDNS Zapret ISP осуществляет запросы к серверам в интернет, входит в адресное пространство, для которого настроено использование GGC.

Установка

Установка Debian

1. Установите Debian согласно официальному руководству
http://www.debian.org/releases/stable/amd64/.

SkyDNS Zapret ISP может быть установлен как на Debian 8 Jessie, так и на Debian 9 Stretch.

<з>Установочный образ Debian 8 Jessie находится по адресу
https://www.debian.org/releases/jessie/debian-installer/.

Установочный образ Debian 9 Stretch находится по адресу
https://www.debian.org/releases/stretch/debian-installer/.

Для работоспособности сервера достаточно установить **SSH server** и **Стандартные системные утилиты**. Остальное ПО устанавливать не нужно.

> Обратите внимание, что версия Debian Jessie должна быть 8.5 или выше; Debian Stretch 9.8 или выше.

2. Убедитесь что в `/etc/apt/sources.list` присутствуют сетевые репозитории Debian:

```
Для Debian 8:
deb http://httpredir.debian.org/debian jessie main
deb http://httpredir.debian.org/debian jessie-updates main

Для Debian 9:
deb http://httpredir.debian.org/debian stretch main
deb http://httpredir.debian.org/debian stretch-updates main
```

3. Обновите систему

```
sudo apt-get update
sudo apt-get dist-upgrade
```

После этого перезагрузите сервер.

Установка Репозитория SkyDNS

1. Отправьте на электронный адрес технической поддержки SkyDNS **support@skydns.ru** письмо, с запросом на получение логина и пароля для доступа к репозиторию.

2. Установите пакет apt-transport-https:

```
sudo apt-get install apt-transport-https
```

3. Добавьте GPG ключ:

```
Для Debian8:
wget -qO - https://<YOUR_REPO_LOGIN>:<YOUR_REPO_KEY>@mirror.skydns.ru/zapret-info/zapret3/keyrin

Для Debian9:
wget -qO - https://<YOUR_REPO_LOGIN>:<YOUR_REPO_KEY>@mirror.skydns.ru/zapret-info/zapret3-deb9/k
```

Если вы используете **тестовую версию**, то ключ добавляется следующей командой:

```
wget -qO - https://<YOUR_REPO_LOGIN>:<YOUR_REPO_KEY>@mirror.skydns.ru/zapret-info-test/zapret3/k
```

4. В `/etc/apt/sources.list` добавьте следующие репозитории:

```
Для Debian 8:
deb http://httpredir.debian.org/debian jessie-backports main
deb https://<YOUR_REPO_LOGIN>:<YOUR_REPO_KEY>@mirror.skydns.ru/zapret-info/zapret3/ ./

Для Debian 9:
deb http://httpredir.debian.org/debian stretch-backports main
deb https://<YOUR_REPO_LOGIN>:<YOUR_REPO_KEY>@mirror.skydns.ru/zapret-info/zapret3-deb9/ ./
```

Если вы используете **тестовую версию**, то вам нужно добавить репозитории:

```
Для Debian 8:
deb http://httpredir.debian.org/debian jessie-backports main
deb https://:@mirror.skydns.ru/zapret-info-test/zapret3/ ./

Для Debian 9:
deb http://httpredir.debian.org/debian stretch-backports main
deb https://:@mirror.skydns.ru/zapret-info-test/zapret3-deb9/ ./
```

5. Убедитесь, что ваша консольная кодировка по умолчанию **ru_RU.UTF-8** или **en_US.UTF-8**.

Установка и обновление SkyDNS Zapret ISP

Установка и обновление осуществляется командами:

```
sudo apt-get update
sudo apt-get install skydns-zi
```

> Если Вы используете **тестовую версию**, то вместо пакета skydns-zi нужно устанавливать skydns-zi-test:

```
sudo apt-get install skydns-zi-test
```

Далее следуйте инструкции Настройка.

Настройка

Добавления алгоритма GOST

Для формирования ключей Роскомнадзор использует алгоритм **GOST**. Данный алгоритм не является используемым по умолчанию. Чтобы проверить, включен ли он на Вашем сервере выполните:

```
openssl ciphers | tr ':' '\n' | grep GOST
```

Если Вы видите следующий вывод:

```
GOST2012-GOST8912-GOST8912
GOST2001-GOST89-GOST89
```

То можете перейти к следующему разделу.

В противном случае Вам необходимо добавить **GOST** в **openssl**. Готовый пакет Вы можете скачать из репозитория SkyDNS:

```
apt install libengine-gost-openssl1.1
```

> Если Вы используете Debian 8 Jessie, тогда версию **openssl** из репозитория SkyDNS с помощью команды:
> `apt install openssl_1.0.2l-1~bpo8+1_amd64.deb`

Если Вы не изменяли конфигурационный файл **openssl**, тогда выполните следующие команды:

```
sed -i "1s/^/openssl_conf = openssl_def\n/" /usr/lib/ssl/openssl.cnf
echo "
[openssl_def]
engines = engine_section

[engine_section]
gost = gost_section

[gost_section]
engine_id = gost
default_algorithms = ALL
CRYPT_PARAMS = id-Gost28147-89-CryptoPro-A-ParamSet
" >> /usr/lib/ssl/openssl.cnf
```

Иначе добавьте эти разделы самостоятельно.

Проверьте, что **GOST** алгоритмы используются библиотекой.

Получение реестра запрещенных ресурсов

> Пропустить это шаг, если вы используете **тестовую версию** (Тестовая версия).

Чтобы осуществлять выгрузку списка заблокированных ресурсов, провайдер должен иметь ЭЦП. Требования к ЭЦП описаны на сайте Роскомнадзора http://eais.rkn.gov.ru/tooperators/.

Выберите режим подписи запроса на получение реестра:

- подписывать запросы прямо на сервере автоматически;
- подписать запрос вручную один раз.

Для автоматической подписи нужно иметь в виде файлов сертификат ЭЦП и незашифрованный закрытый ключ от него, оба файла должны быть в формате PEM. Они должны начинаться, соответственно, со строк `----- BEGIN CERTIFICATE -----` и `----- BEGIN PRIVATE KEY -----` и не содержать слово **Encrypted в заголовке.**

Автоматический режим подписания запросов на получение реестра

Скопируйте ключ и сертификат на сервер в каталог `/var/lib/skydns-zi/keys/`. Убедитесь в правильности заголовков файлов. В файле `/etc/skydns-zi/config.yml` пропишите данные, необходимые для автоматического построения запроса:

```
downloader:
  rkn:
    operator_info:
      operatorname: ООО "ОПЕРАТОР"
      inn: '6670123456'
      ogrn: '116670002345'
      email: 'root@loclhost.com'
      certificate: /var/lib/skydns-zi/keys/private.pem
      privatekey: /var/lib/skydns-zi/keys/id_rsa.key

    # Этот блок необходимо закомментировать или удалить
    # signed_files:
    #   xmlreqfile: /var/lib/skydns-zi/req/request.xml
    #   signaturefile: /var/lib/skydns-zi/req/request.xml.sig
```

Ручной режим подписания запроса на получение реестра (один раз)

- Создайте XML-файл request.xml по образцу:

```
<?xml version="1.0" encoding="windows-1251"?>
<request>
<requestTime>2014-11-26T18:04:31.000+00:00</requestTime>
<operatorName>ООО Оператор</operatorName>
<inn>6670123456</inn>
<ogrn>1116670002345</ogrn>
<email>root@operator.ru</email>
</request>
```

> Кодировка windows-1251, двойные кавычки, переводы строк в стиле Windows, никаких лишних пробелов. Сайт eais.rkn.gov.ru разбирает XML не по стандарту.

- Для этого файла создайте электронную подпись в формате PKCS7 (base64) в отдельном файле request.xml.sig

Если вы все сделано правильно, то файл подписи будет начинаться со строки `----- BEGIN PKCS7 -----`.

- Оригинальный файл и файл подписи скопируйте на сервер в каталог `/var/lib/skydns-zi/req/`

Правильность подписи и копирования можно проверить на сервере при условии наличия там OpenSSL 1.0.1 командой:

```
sudo openssl smime -binary -noverify -engine gost -verify -inform PEM -in /var/lib/skydns-zi/req
```

Эта команда должна успешно завершаться в том виде, как указано выше, и (это важно) выдавать ошибку **no content**, если убрать опцию `-content /var/lib/skydns-zi/req/request.xml`.

> При копировании XML-файла могут испортиться окончания строк, что сделает подпись недействительной. Файлы надо передавать в бинарном режиме.
> Некоторые программы обрамляют подпись в `----- BEGIN PKCS7 SIGNED -----`. Это нарушение стандарта (должно быть `----- BEGIN PKCS7 -----`, без лишних пробелов). К счастью, это легко поддается исправлению в текстовом редакторе (END тоже надо исправить).
> Некоторые программы под Windows перед подписью игнорируют последний символ новой строки в подписываемом файле. Если ничего не помогает (то есть подпись никак не подходит), этот символ надо попробовать убрать.

В файле `/etc/skydns-zi/config.yml` укажите пути к созданным файлам:

```
downloader:
  rkn:
    # Этот блок необходимо закомментировать или удалить
    # operator_info:
    #   operatorname: ООО "ОПЕРАТОР"
```

```
#   inn: '6670123456'
#   ogrn: '116670002345'
#   email: 'root@loclhost.com'
#   certificate: /var/lib/skydns-zi/keys/private.pem
#   privatekey: /var/lib/skydns-zi/keys/id_rsa.key

signed_files:
  xmlreqfile: /var/lib/skydns-zi/req/request.xml
  signaturefile: /var/lib/skydns-zi/req/request.xml.sig
```

Динамическая маршрутизация

Необходимость настройки протокола маршрутизации на этом этапе зависит от выбранной схемы подключения SkyDNS Zapret ISP в сеть (см. Схемы подключения SkyDNS Zapret ISP в сеть).

Общая схема, используемая при динамической маршрутизации: демон **Exabgp** отправляет демону **Quagga bgpd** маршруты по **bgp**, который заносит маршруты в локальную таблицу маршрутизации. После чего демон **Quagga ospfd** (**bgpd**, если Вы решите создать **bgp-сессию** между системой фильтрации и внутренним маршрутизатором) анонсирует маршруты на маршрутизатор.

Выполните команду, которая запросит необходимые для настройки данные и выполнит настройку OSPFv2 в случае, когда SkyDNS Zapret ISP и внутренний маршрутизатор находятся в одной сети и никаких **ospf-сессий** больше не существует.

zi-ctl configure

```
sudo zi-ctl configure
```

В случае если необходимо произвести более сложную настройку динамической маршрутизации, то это нужно сделать самостоятельно.

Конфигурационный файлы для Quagga и Exabgp располагаются в `/etc/quagga/` и `/etc/exabgp/` соответственно.

Статическая маршрутизация

Для настройки статической маршрутизации необходимо написать скрипт загрузки маршрутов на маршрутизатор и добавить его исполнение в `cron`.

Вы можете использовать примеры скриптов на Shell Script для роутеров Cisco и Huawei, либо использовать любой другой удобный для вас язык.

Cisco

```
#! /bin/sh -e

acl_output=' /srv/tftp/routes. acl'
echo "no ip access-list extended webtraffic" > ${acl_output}. tmp
echo "ip access-list extended webtraffic" >> ${acl_output}. tmp
for route in $(zi-ctl routes && zi-ctl routes --ipv6)
do
    echo "permit ip any host ${route}" >> ${acl_output}. tmp
done
echo "end" >> ${acl_output}. tmp
mv ${acl_output}. tmp ${acl_output}

# Здесь нужно написать команду, которая выполнит загрузку правил на маршрутизатор
```

Huawei

```
#! /bin/sh -e


acl_output=' /srv/tftp/routes. acl'
echo "acl number 5000" > ${acl_output}. tmp
i=0
for route in $(zi-ctl routes && zi-ctl routes --ipv6)
do
    i=$((i+=5))
    echo "rule ${i} deny destination ${route}" >> ${acl_output}. tmp
done
mv ${acl_output}. tmp ${acl_output}


# Здесь нужно написать команду, которая выполнит загрузку правил на маршрутизатор
```

Вы также можете скопировать данные примеры из директории `/usr/share/skydns-zi/examples/` .

Пример

> #.#.#.# - IP-адрес TFTP-сервера.

Рассмотрим пример внедрения статической маршрутизации для роутера Cisco ASR1002.

- В конец скрипта добавьте строчку:

```
/usr/bin/snmpset -v1 -ccommunity ${router_ip} .1.3.6.1.4.1.9.2.1.53.#.#.#.# s ${acl_output}
```

- Установите пакет SNMP:

```
sudo apt-get install snmp
```

- Установите и настройте сервер TFTP:

```
sudo apt-get install atftpd
```

- Создайте каталог `/srv/tftp/`:

```
mkdir –p /srv/tftp
```

- Создайте маршрутную карту на роутере:

```
route-map webcache-redirect permit 10
match ip address webtraffic
set ip next-hop <zapret-isp-ip>
```

- Примените маршрутную карту на интерфейс, на который приходит трафик от пользователей. Например, `vlan 1`:

```
int vlan 1
ip policy route-map webcache-redirect
```

- Добавьте правило в `cron` на исполнение скрипта каждые шесть минут.

Настройка фильтрующего DNS-сервера

Вместе с пакетом поставляется DNS-сервер Unbound. Он установится на сервер во время установки пакета.

Фильтрующий DNS-сервер запущен на `127.0.0.2:53` для протокола IPv4 и на `::1@12346` для протокола IPv6. Запросы к нему перенаправляются посредством правил `iptables`.

Распространять адрес фильтрующего DNS-сервера можно, используя протокол DHCP, или указав его вышестоящим для Ваших DNS-серверов.

Запустите команды:

```
# добавить DNS-сервер в автозапуск
systemctl enable skydns-unbound
# перезапустить сервис
service skydns-unbound restart
```

Прием трафика

Для работы продукта необходимо задать список сетей, трафик из которых будет фильтроваться. Сделать это можно, используя консольную команду zi-ctl nets:

```
sudo zi-ctl nets add NET_1 NET_2 NET_N
```

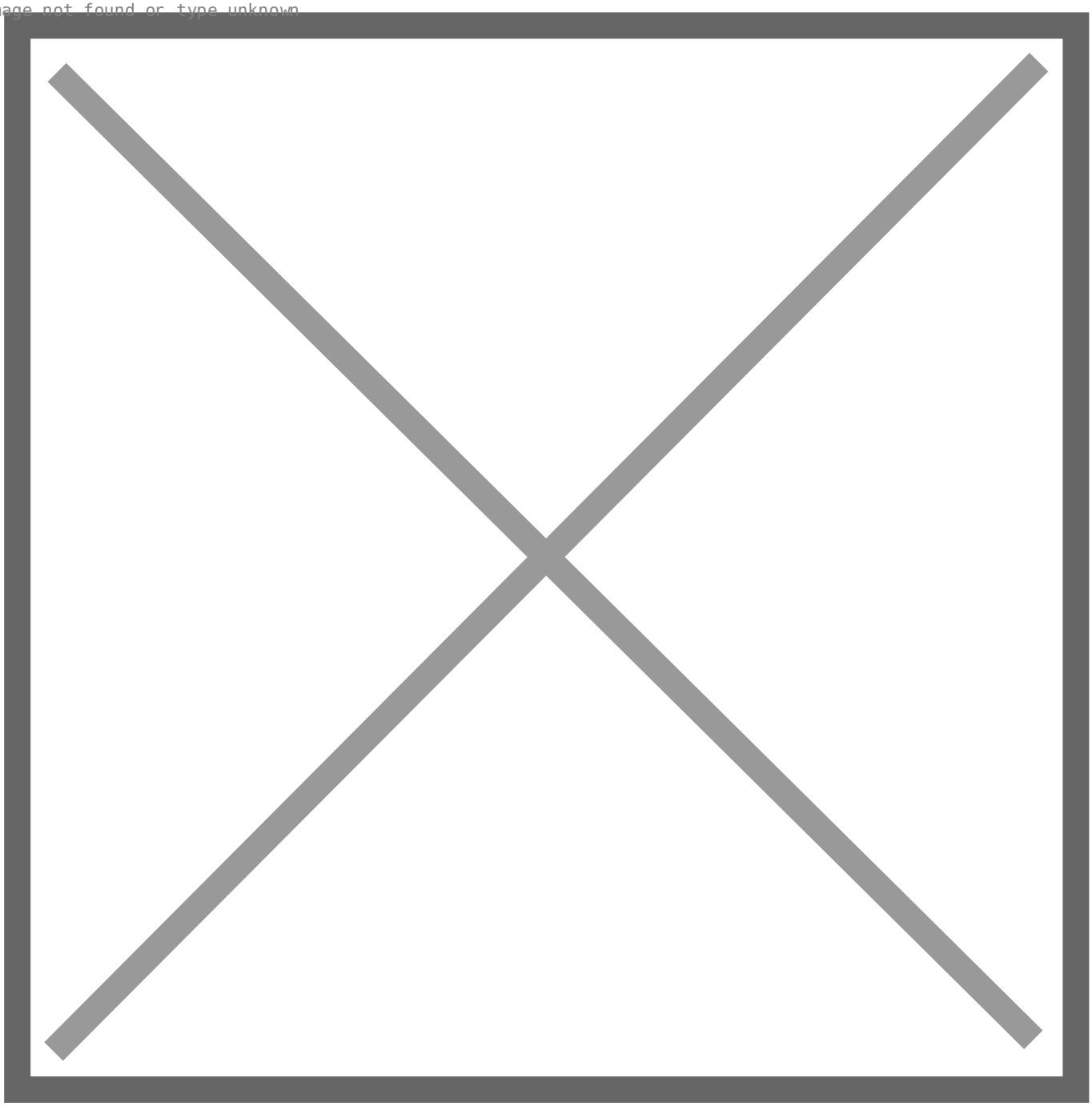
Создание административного пользователя

Чтобы начать использовать web-интерфейс, необходимо создать пользователя. Сделать это можно, используя консольную команду zi-ctl user:

```
sudo zi-ctl user edit
```

По умолчанию WEB интерфейс размещён на `0.0.0.0:80.`

Image not found or type unknown



В окне введите почтовый адрес и пароль только что созданного пользователя.

Настройка почтовых отправлений

> Нерегулярное обновление реестра может вызвать претензии Роскомнадзора. Следует читать почту, которую получает root. Все уведомления об ошибках скачивания реестра будут направляться туда.

В файле `/etc/cron.d/skydns-zi`, вместо `MAILTO=root` пропишите `MAILTO=<YOUR_EMAIL>` и настройте MTA для отправки почты. Для отправки уведомлений на несколько адресов e-mail, необходимо в MTA прописать алиасы.

При успешном завершении скриптов загрузки и преобразования, сообщение будет следующего вида:

```
Downloading module MJ has finished.
Downloading module RKN has finished.
Parsing module MJ has finished.
Parsing module RKN has finished.
```

где MJ - для списка Министерства юстиции РФ, RKN - для списка Роскомнадзора.

Пример завершения с ошибкой:

```
Downloading module RKN has finished.
Downloading module MJ has finished with exception.  # Модуль загрузки выполнился с ошибкой.
Traceback (most recent call last):
  File "/usr/share/python/skydns-zi/local/lib/python3.4/site-packages/isp_filter/downloader/__in
    module_.download(info)
  File "/usr/share/python/skydns-zi/local/lib/python3.4/site-packages/isp_filter/downloader/mj.p
    raise Exception()
Exception:
Parsing module RKN has finished.
Parsing module MJ has finished.
```

Завершение настройки и проверка

Ограничение доступа к web-интерфейсу по IP-адресу

Из соображений безопасности рекомендуем Вам ограничить доступ к web-интерфйсу по IP-адресу:

```
iptables -I INPUT -p tcp ! -s <YOUR_IP_ADDRESS> --dport 80 -j DROP
```

`<YOUR_IP_ADDRESS>` - IP-адрес компьютера, с которого Вы будете использовать web-интерфейс.

Первоначальная проверка

- Убедитесь, что сервер SkyDNS Zapret ISP имеет выход в интернет.

С сервера выполните команды `ping 8.8.8.8` и `ping ya.ru` - эти ресурсы должны быть доступны.

- Проверьте работоспособность локального DNS-сервера:

```
# На сервере
dig @127.0.0.1 yandex.ru
```

- Проверьте, что работает фильтрующий DNS-сервер:

```
# На сервере
dig @127.0.0.2 <SOME_BLOCKED_DOMAIN>

# На клиенте
dig @%s.%s.%s.%s <SOME_BLOCKED_DOMAIN>
```

В ответе будут IP-адреса, с неизменяющимся ttl равным 3600.

- Загрузите источники, используя консольную команду zi-ctl download:

```
sudo zi-ctl download
```

Она выполняется несколько минут. На выходе будут сообщения вида:

```
2018-06-28 12:20:56,697 Downloading module <SRC_NAME> has finished. # Загрузка завершилась успеш
2018-06-28 12:20:57,405 Parsing module <SRC_NAME> has finished. # Загруженные данные были успешн
```

`<SRC_NAME>` - название источника (MJ - для списка Министерства юстиции РФ, RKN - для списка Роскомнадзора)

- Перезапустите сервис zi-update:

```
sudo supervisorctl restart zi-update
```

Проверьте, что он работает:

```
tail -f /var/log/skydns-zi/updater.log
```

На выходе будет лог, в котором указана текущая скорость разрешения доменных имен и количество успешно/неуспешно разрешенных доменов на текущий момент. Также там должна присутствовать информация о соответствии доменом с IP-адресами:

```
2018-10-05 11:06:04,812 Host <domain>: ip <ipv4>
2018-10-05 11:06:04,812 Host <domain>: ip <ipv6>
2018-10-05 11:06:04,812 Could not resolve domain <domain> into IPv4/IPv6
```

- Проверьте настройку маршрутизации:

Если используются протоколы динамической маршрутизации, проверьте их настройки, наличие запущенных демонов, а также установление отношения соседства следующими способами:

Выполните команду `service quagga status` и проверьте наличие следующих процессов:

```
CGroup: /system.slice/quagga.service
        ├─30266 /usr/lib/quagga/zebra --daemon -A 127.0.0.1 # Основной демон Quagga.
        ├─30270 /usr/lib/quagga/bgpd --daemon -A 127.0.0.1 # Демон BGP.
        ├─30274 /usr/lib/quagga/ospfd --daemon -A 127.0.0.1 # Демон OSPF.
        └─30279 /usr/lib/quagga/watchquagga --daemon zebra bgpd ospfd # Демон мониторинга.
```

Проверьте, что запущен Exabgp `service exabgp status`.

Для проверки состояния BGP-сессии между Exabgp и Quagga выполните команду:

```
vtysh -E -c "sh ip bgp neighbors"
```

Для проверки состояния соседей по OSPF выполните команду:

```
vtysh -E -c "sh ip ospf neighbor"
```

Проверьте, что маршрутизатор направляет трафик, идущий на IP-адреса запрещенных ресурсов, на систему фильтрации. Это можно сделать, проверив командой `tracert` или

`traceroute` маршрут до какого-либо заблокированного ресурса с компьютера, выступающего в роли компьютера клиента. После выполнения команды, один из промежуточных IP-адресов должен быть адресом SkyDNS Zapret ISP.

- Проверьте правила iptables и списки ipset.

Выполните команды `iptables-save` и `ipset list`. Убедитесь в правильности правил iptables (предполагаемые правила располагаются в `/usr/share/skydns-zi/firewall.conf`) и наличии списков ipset.

- Проверьте, что запущен сервис Squid:

```
service squid status
```

Если все шаги пройдены, то система работает корректно.

Дополнительная проверка

После выполнения первоначальной проверки, рекомендуется выполнить проверку утилитой SkyDNS Zapret Check.

Это позволит Вам выявить наличие каких-либо неявных проблем.

Логирование

По умолчанию после установки включен уровень записи в логи DEBUG - подробное отслеживание всего, что происходит в системе. После завершения настройки, в случае, если объем дискового пространства, которым Вы располагаете, меньше 20 Gb, установите уровень записи в логи INFO (логируются основные события - загрузка списков и т.п) или WARNING (только предупреждения и ошибки). Сделать это можно в файле `/etc/skydns-zi/config.yml`:

```
---
loglevel: DEBUG # (DEBUG, INFO, WARNING)
```

После изменения уровня логирования нужно перезапустить сервисы:

```
sudo service squid restart
sudo supervisorctl restart zi-update
```

Файлы логов системы фильтрации находятся в папке `/var/log/skydns-zi/`. Предназначение файлов в папке `/var/log/skydns-zi/`:

1. Логи acl пишутся в `acl.log`.
2. Работу zi-update можно отслеживать в `updater.log`.
3. Проверять анонс маршрутов можно через `routing.log`.
4. Всё остальное пишется в `isp-filter.log`.

Файлы логов `squid` находятся в папке `/var/log/squid/`.

Файлы логов `supervisor`, который управляет `zi-update`, находятся в папке `/var/log/supervisor/`.

Файлы логов `uwsgi` (web-интерфейс) находятся в `/var/log/uwsgi/`.

Web-интерфейс

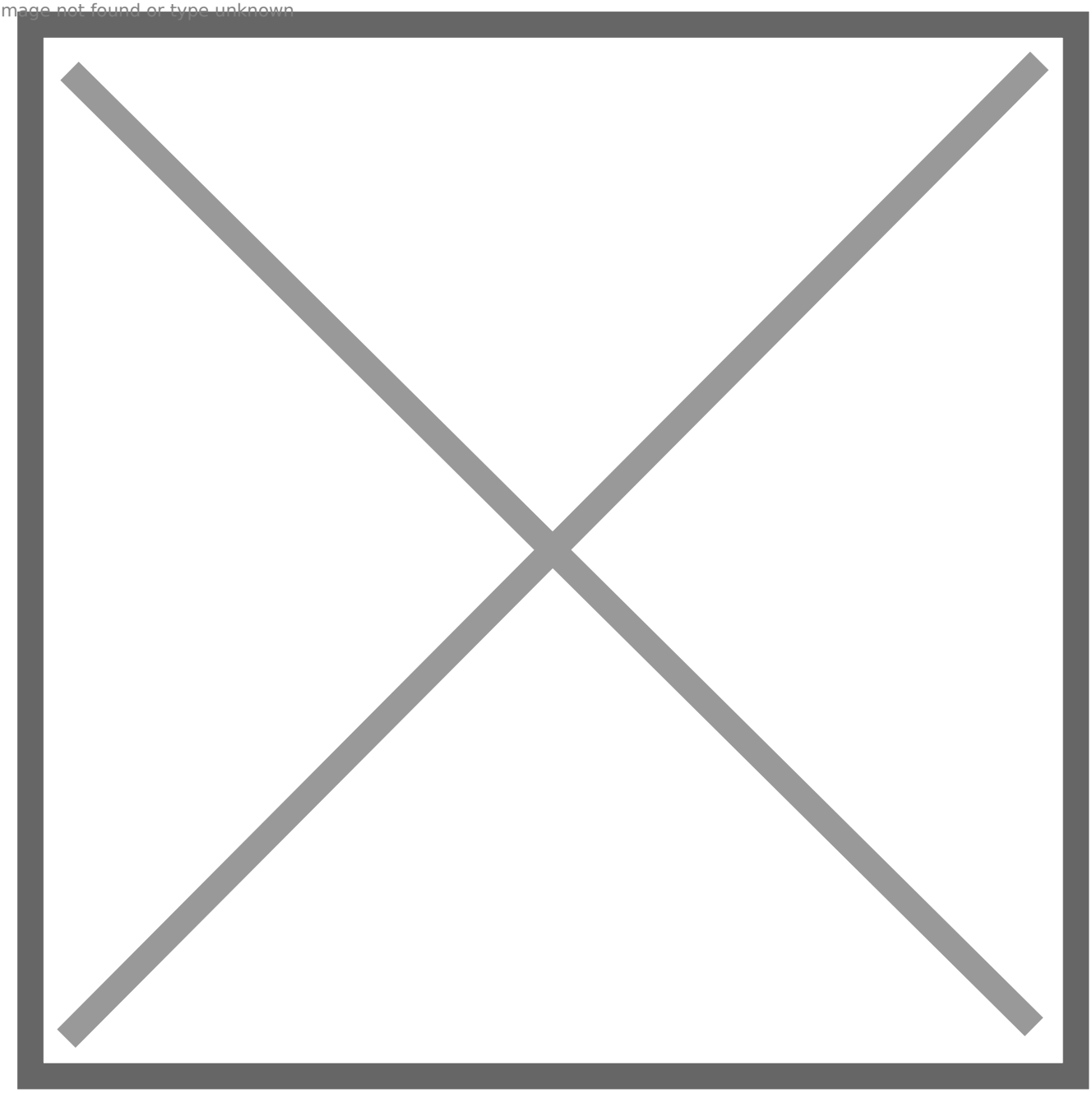
Интерфейс администратора предоставляет возможности удобного мониторинга состояния системы, поиска конкретных заблокированных ресурсов и многое другое.

> Чтобы начать использовать web-интерфейс, необходимо создать пользователя. Если Вы ещё этого не сделали, следуете инструкции (см. Создание административного пользователя).

Просмотр сводной информации о системе

На этой вкладке Вы можете посмотреть общее состояние системы - дату последней успешной загрузки определённого источника, количество свободного места, объёмы потребляемой памяти и тому подобное.

Image not found or type unknown

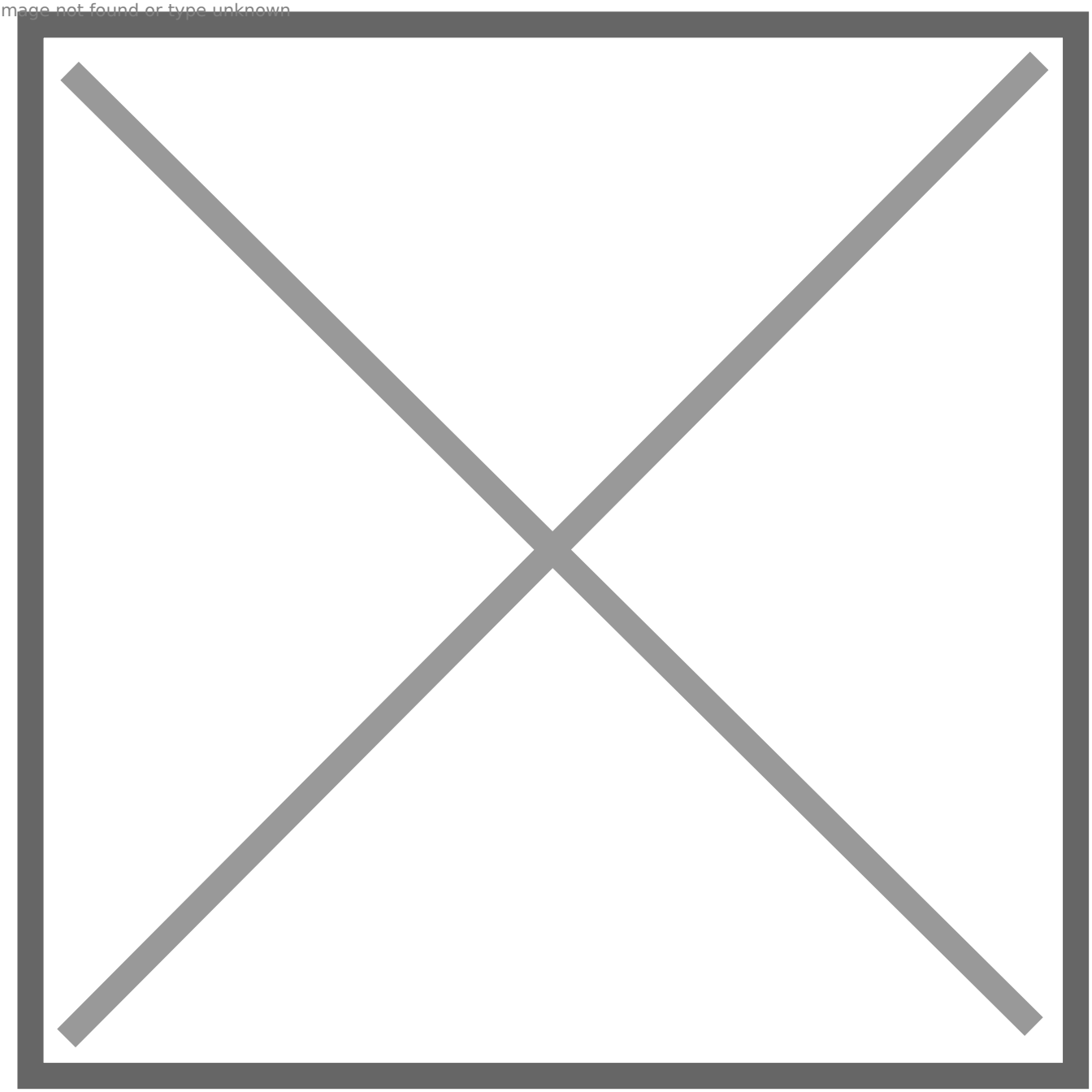


Извещение о наличии проблем

Если во время использования SkyDNS Zapret ISP возникнут какие-то проблемы с системой, то на главном экране, в соответствующем разделе появится оповещение.

> Оповещения появляются при определенном проценте потребления ресурсов. Если Вы хотите это поменять, в Конфигурационный файл поменяйте значения в разделе `admin`.

Image not found or type unknown

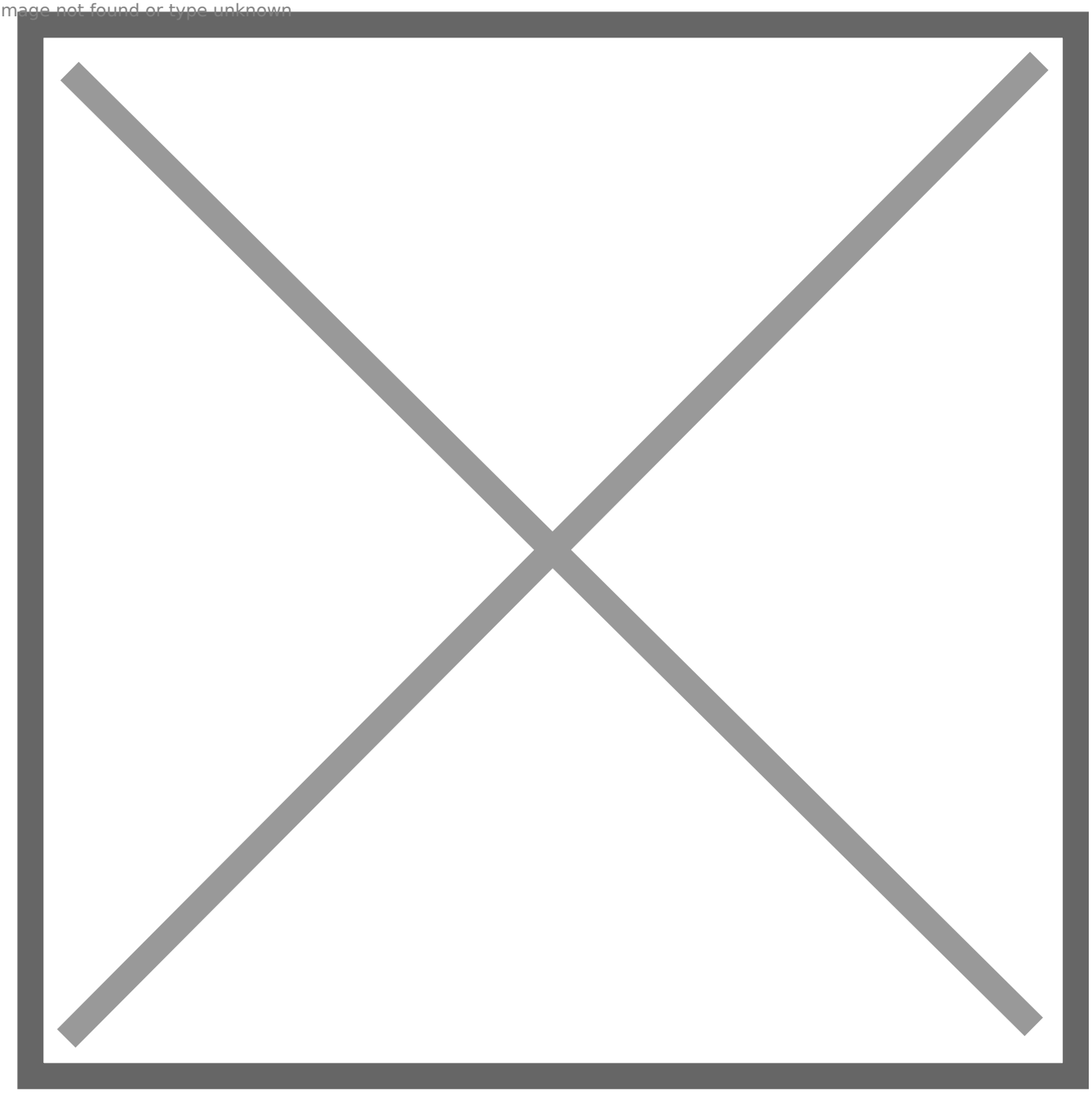


Пример оповещения о проблеме наличия свободного места на диске.

Просмотр и редактирование текущей конфигурации сервиса

В этом разделе Вы можете посмотреть текущий конфигурационный файл, а также изменить его.

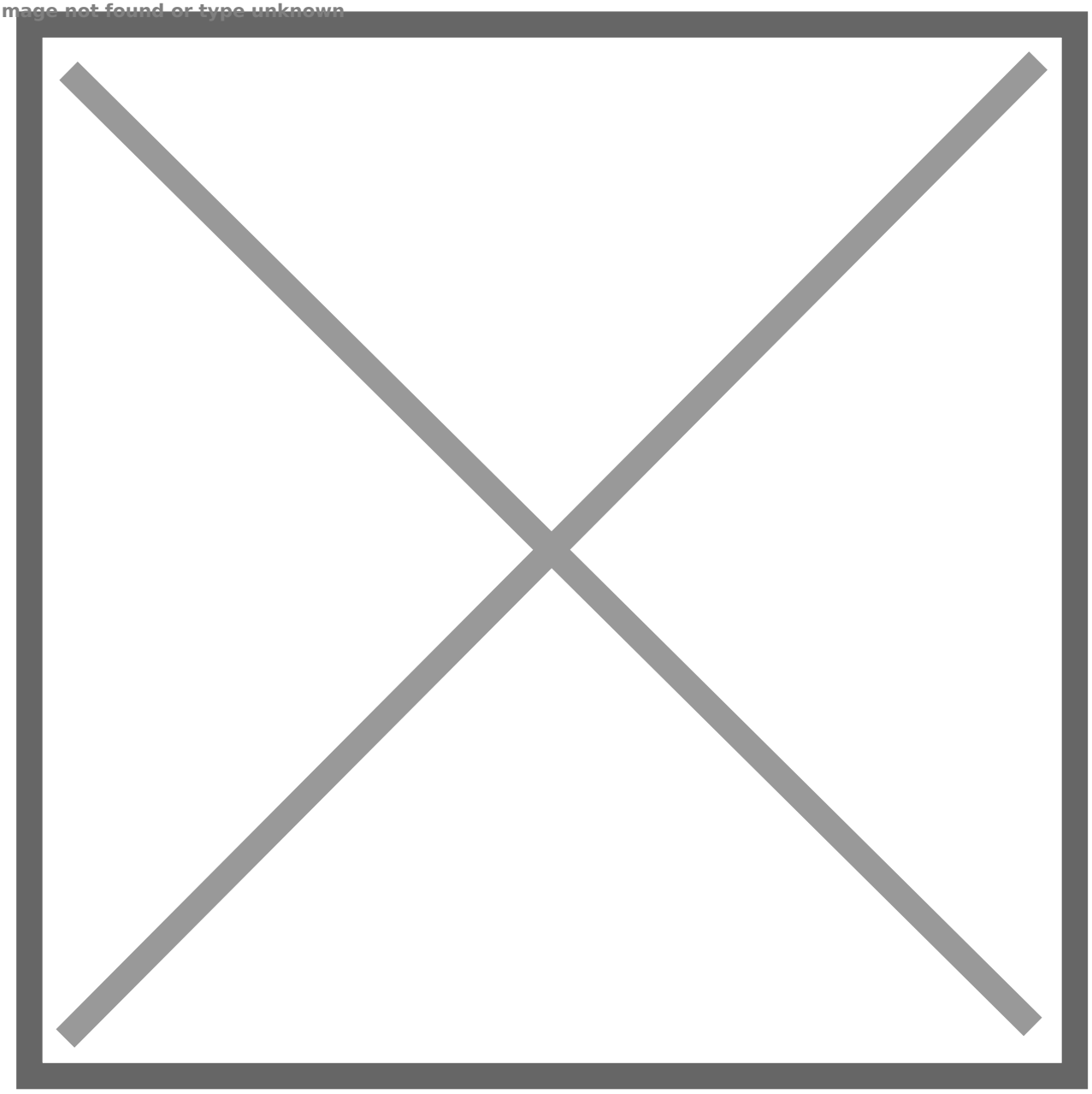
Image not found or type unknown



Редактирование страницы блокировки

В этом разделе Вы можете посмотреть страницу блокировки, а также изменить её.

Image not found or type unknown

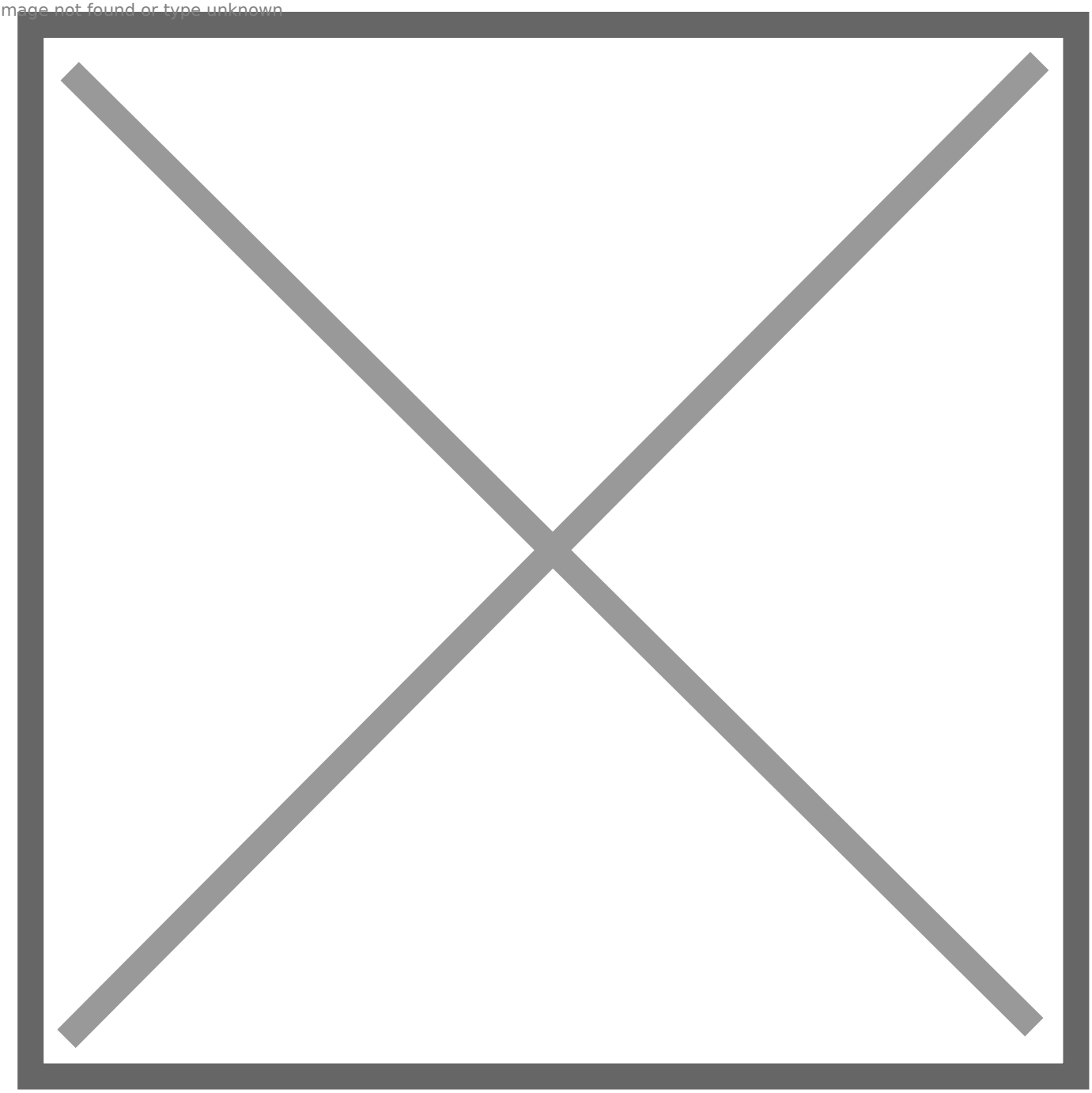


Управление пользователями

В этом разделе Вы может создать новых пользователей или изменить существующих. Все созданные пользователи являются административными, то есть имеют доступ к просмотру всех страниц, могут менять страницу блокировка, а также конфигурационный файл.

Общий вид

Image not found or type unknown

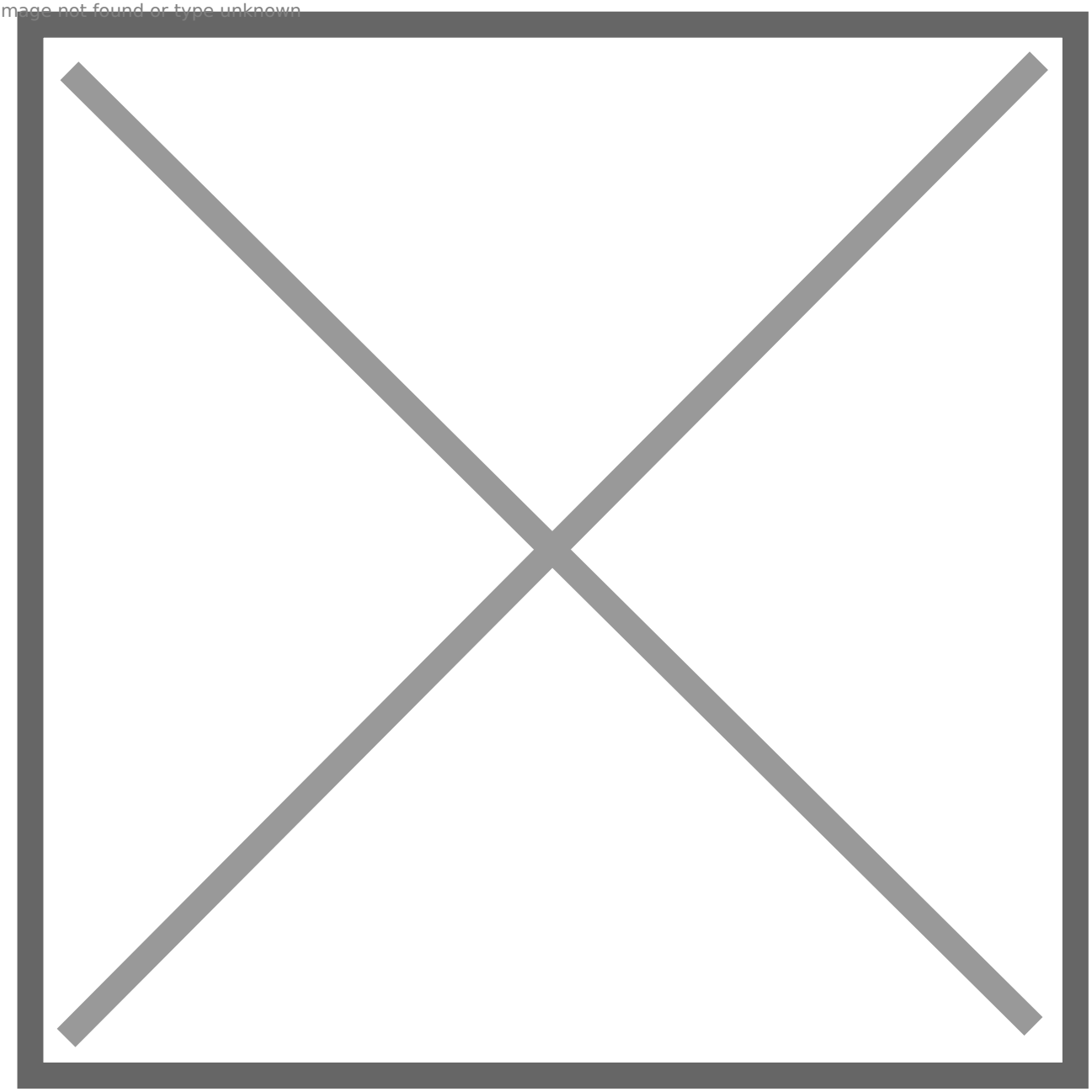


Пример страницы управления пользователями.

Создание нового пользователя

Галочка **Активный** определяет, сможет ли пользователь использовать web-интерфейс.

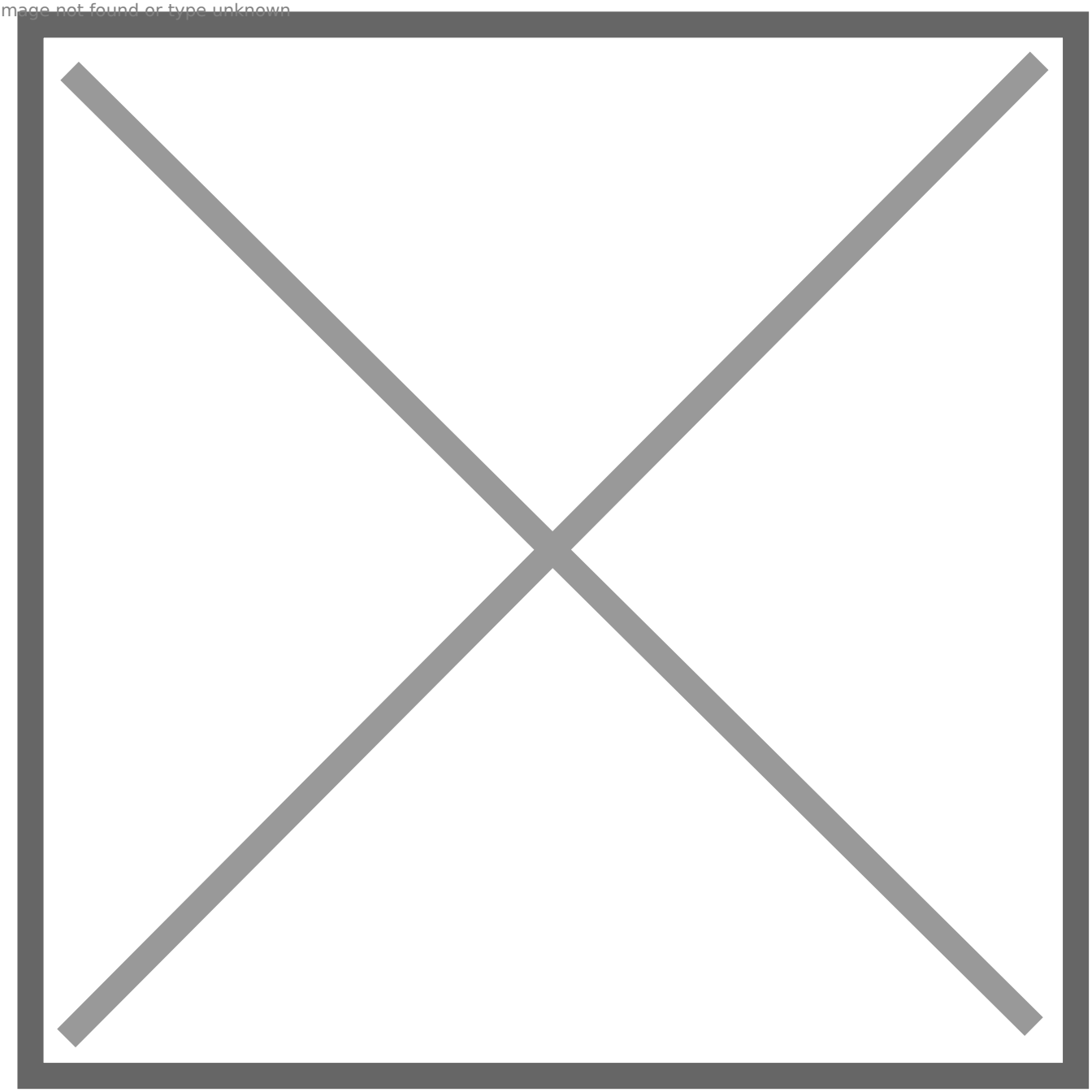
Image not found or type unknown



Списки исключений

В разделе **Списки** вы можете посмотреть сводную информацию по правилам фильтрации для конкретного списка.

Image not found or type unknown



Пояснение к колонкам:

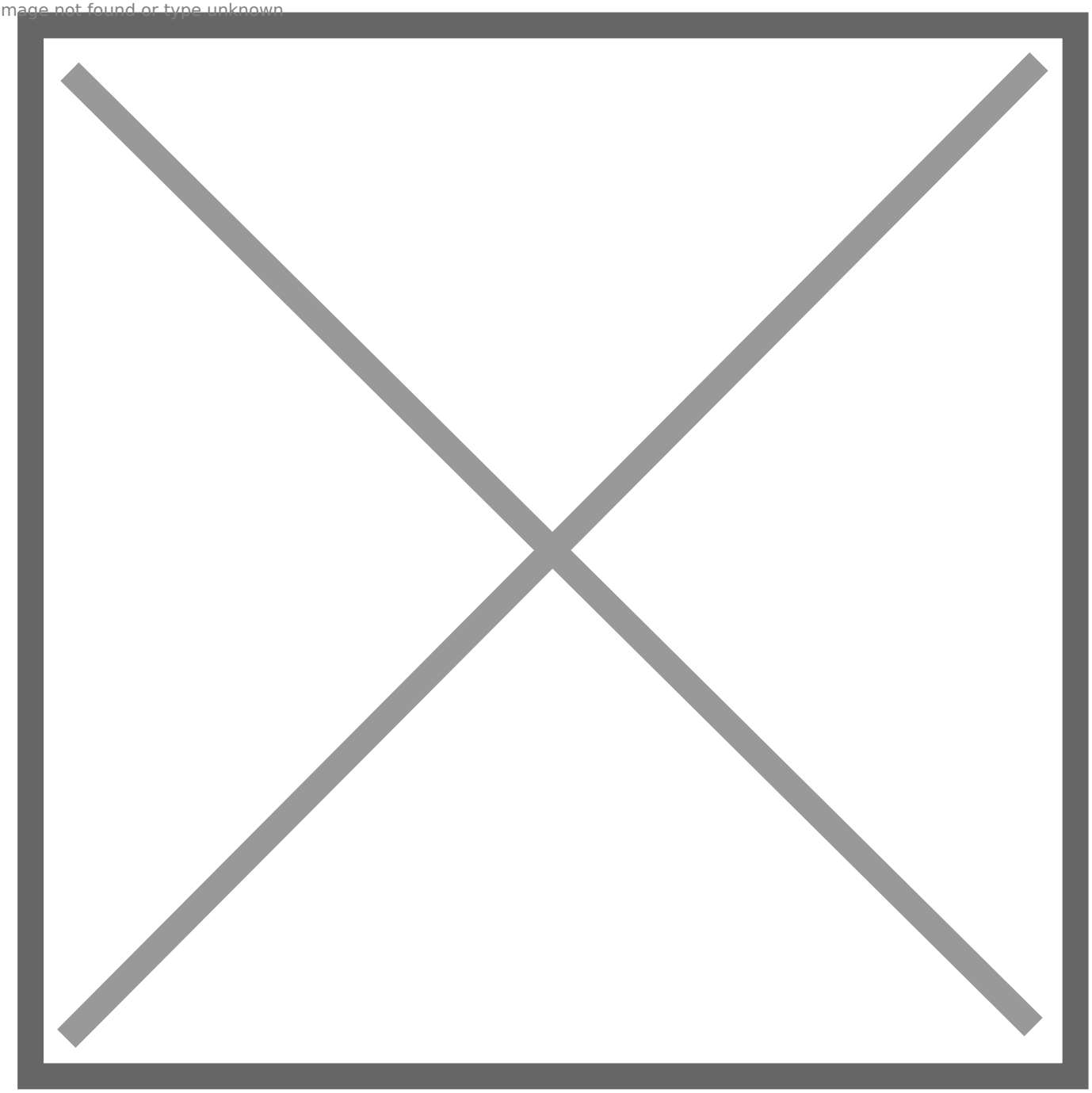
1. Тип блокировки - определяет способ по которому ограничивается доступ к ресурсу.
2. Значение - правило фильтрации.
3. Хост - доменное имя, полученное из правила фильтрации.
4. Фильтруемые протоколы - определяет протокол, по которому будет заблокирован доступ к ресурсу.
5. Маршруты - список маршрутов, которые соответствуют доменному имени.
6. Получить IP-адрес - принудительно разрешает доменное имя.

Создавать или изменять правила можно только для пользовательских списков (см. Списки исключений).

Создание нового правила

Вы можете добавлять собственные правила для более точной настройки фильтрации.

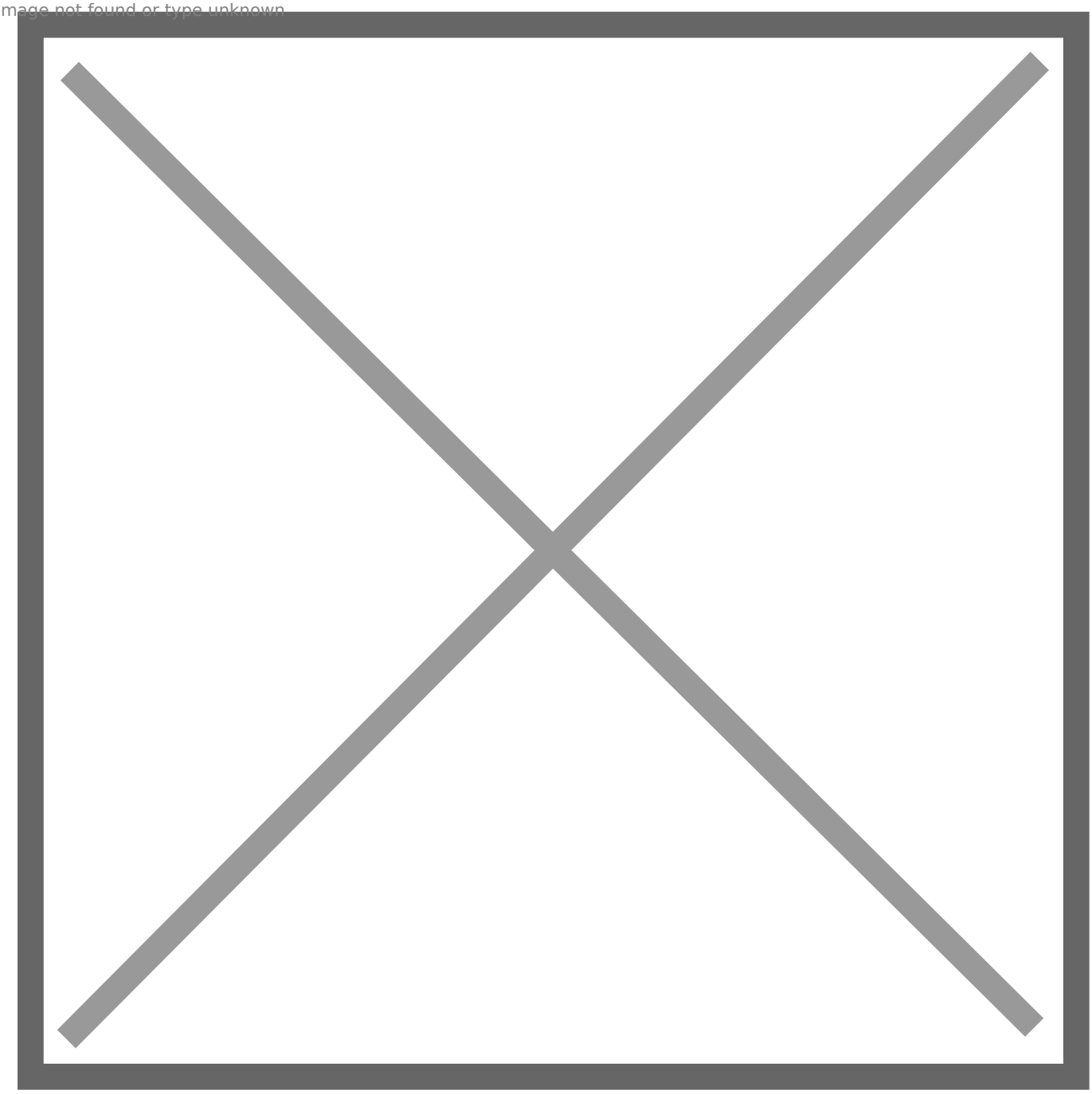
Image not found or type unknown



Изменение существующего правила

В этом разделе вы можете изменять существующие правила. Чтобы изменить правило, нажмите на карандаш в крайней левой колонке.

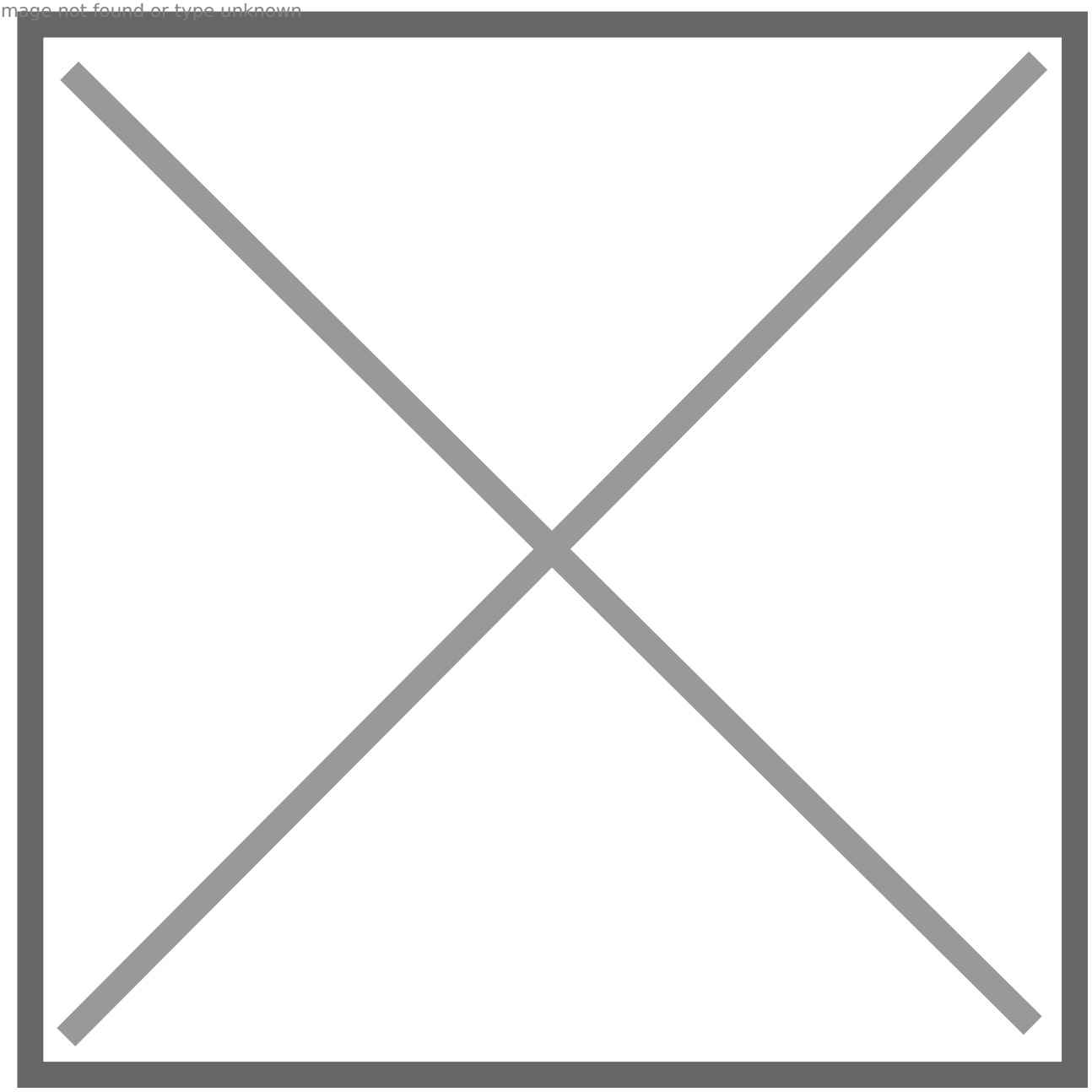
Image not found or type unknown



Фильтруемые подсети

Через этот раздел осуществляется работа со списком сетей, трафик из которых будет проходить через SkyDNS Zapret ISP.

Image not found or type unknown

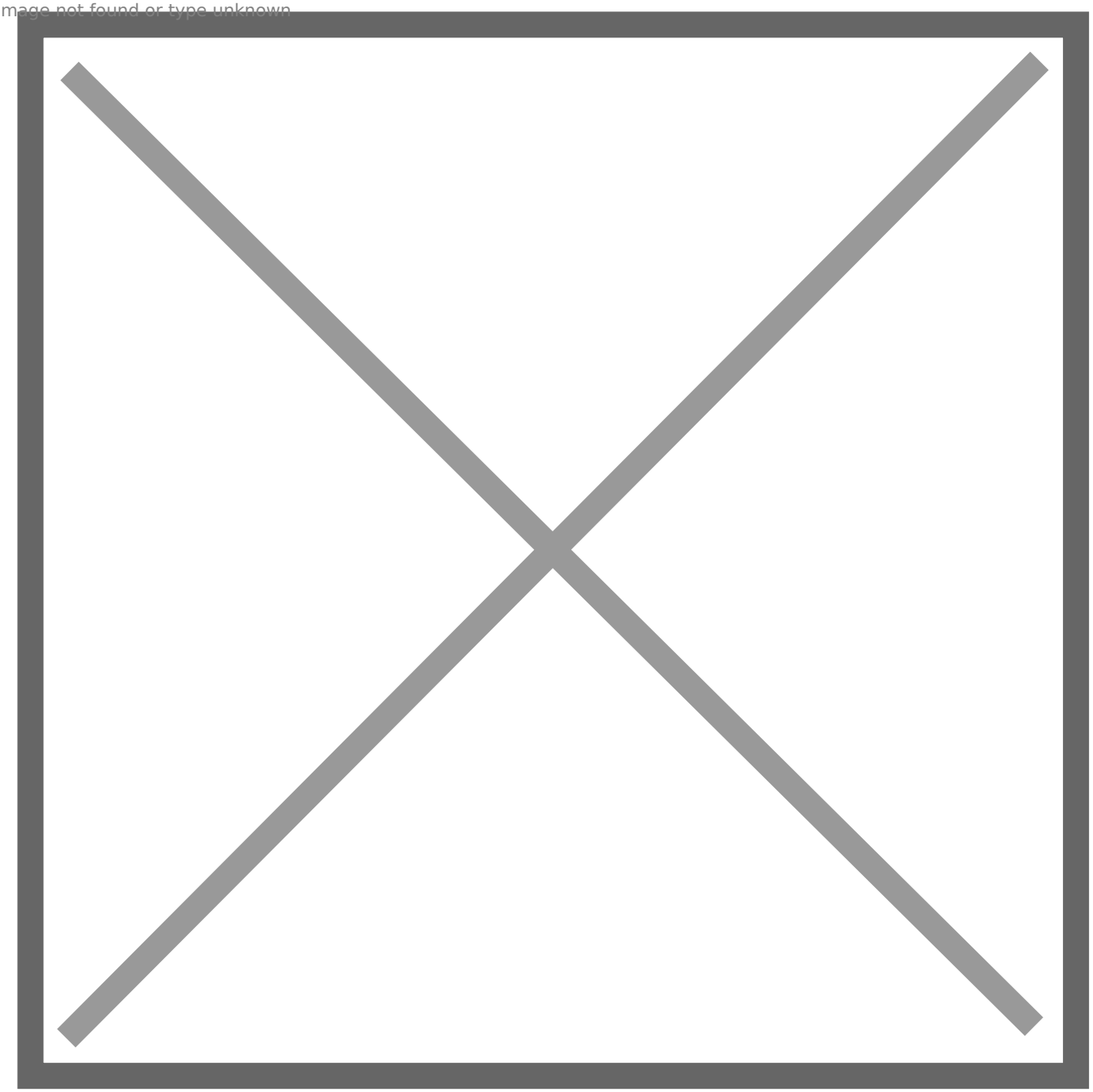


На приведенном скриншоте трафик из сети `192.168.30.0/24` подвергается фильтрации, а из сети `1.1.0.0/18` нет.

Список всех существующих маршрутов

В этом разделе вы можете посмотреть все существующие маршруты.

Image not found or type unknown



Пояснение к колонкам:

1. Сеть - значение сети.
2. Протокол - определяет версию протокола, к которой относится Сеть.
3. Хост - доменное имя, полученное из правил фильтрации.
4. Запись - список правил фильтрации. Они имеют одинаковое значение Хост.
5. Список - список, в котором находятся правила фильтрации из колонки Запись.

Поиск заблокированных ресурсов

Можно производить поиск заблокированных ресурсов среди маршрутов и в списках блокировок для установления причин блокировки того или иного ресурса. Для этого существуют фильтры в разделах Списки и Маршруты

Image not found or type unknown

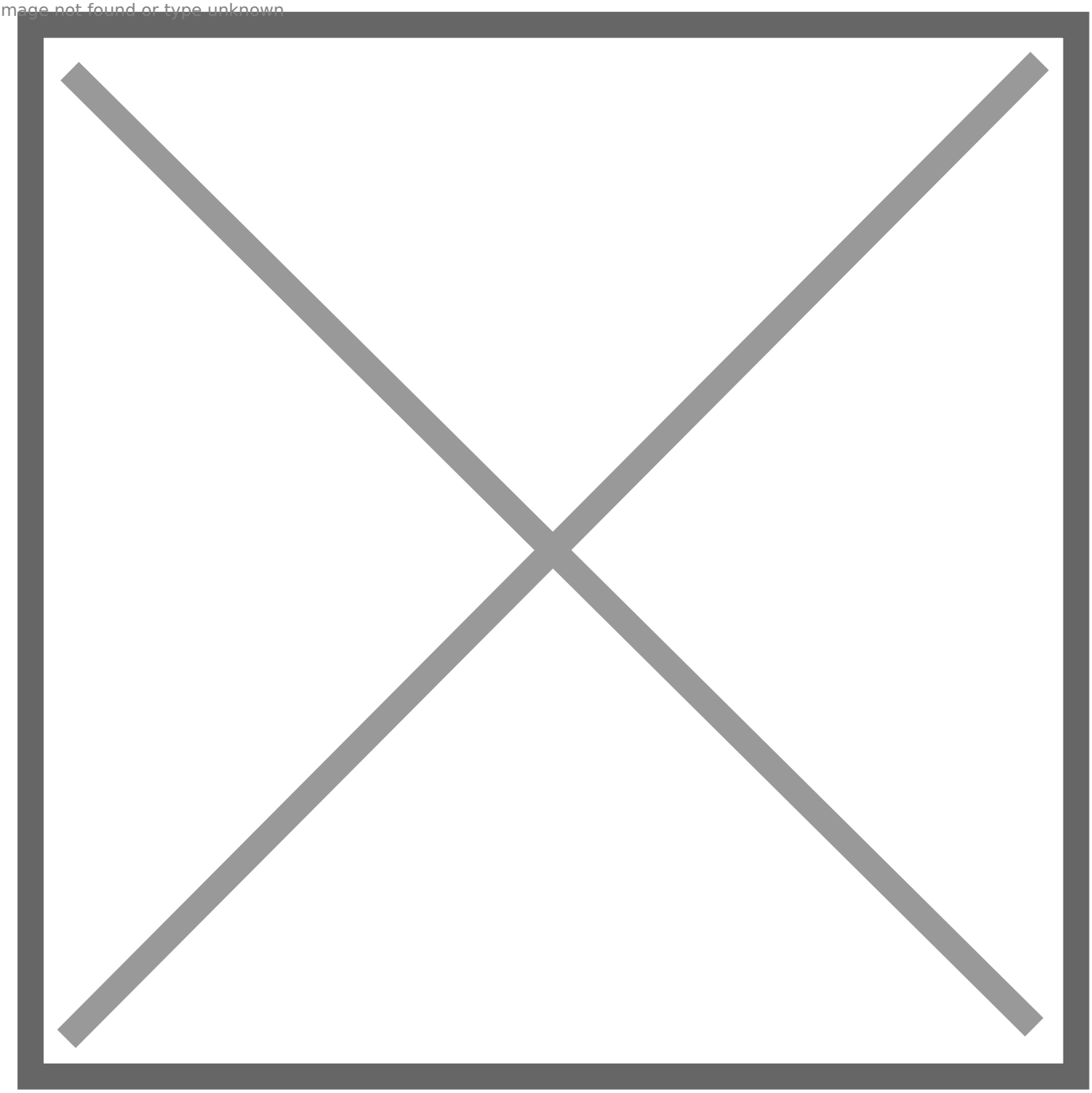
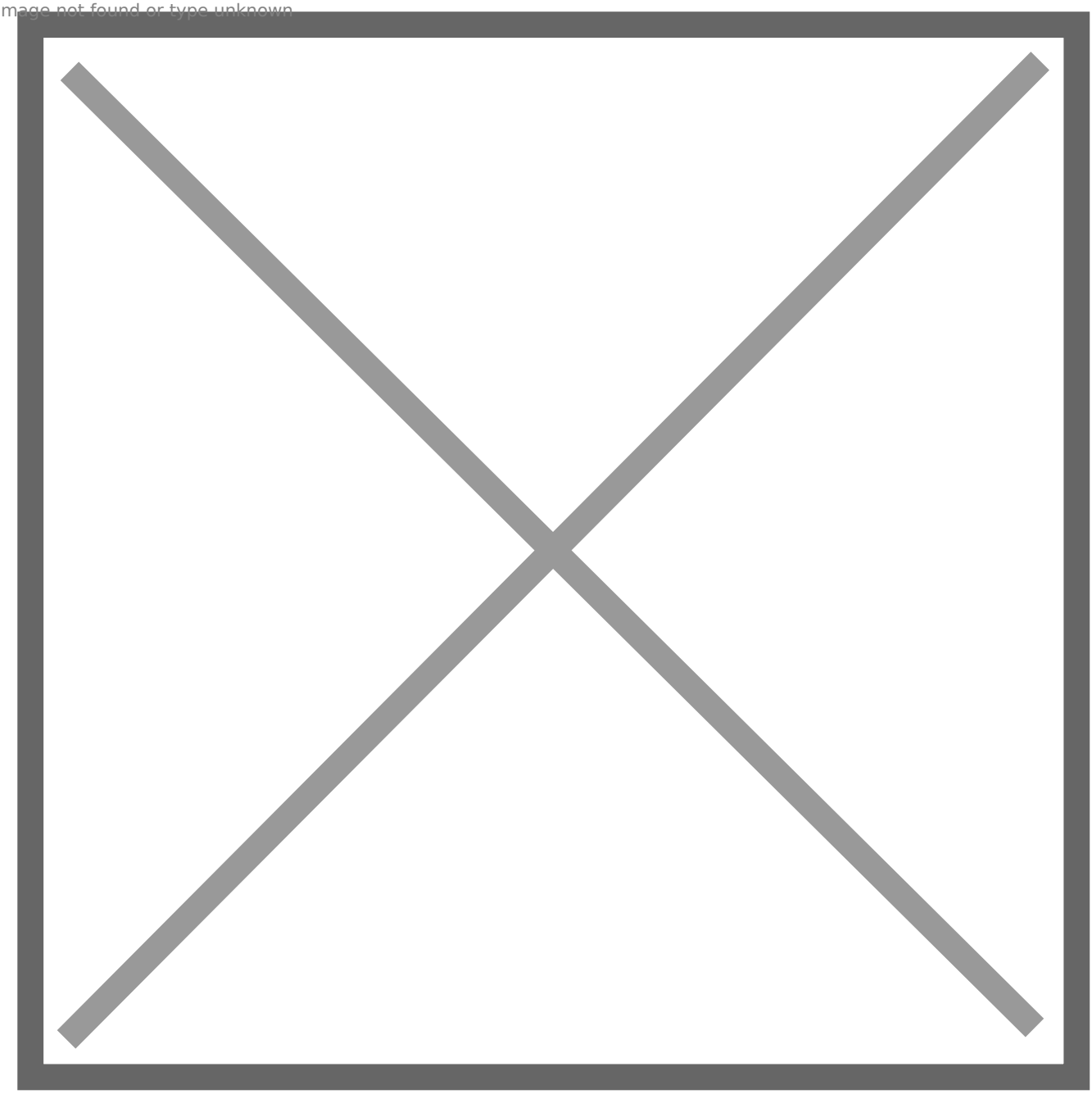


Image not found or type unknown



Команды zi-ctl

SkyDNS Zapret ISP предоставляет консольное меню `zi-ctl`, которое позволяет вручную выполнить действия по управлению системой, если такая возможность потребуется.

Список команд с кратким описанием.

Название команды	Краткое описание
check-announce	Анонсирует маршруты, отсутствующие на маршрутизаторе.
configure	Выполняет настройку OSPF.
convert	Заносит данные, полученные из источников, в базу данных.
create-zones	Создает stub-зоны на основе записей из таблиц Route и Host.
delete-expired	Удаляет маршруты с истекшим ttl.
domain	Анонсирует или удаляет связанные с доменом маршруты.
download	Производит загрузку подключенных источников.
filter	Команда для изменения белого/черного/серого списков.
ipset	Команда для работы с ipset.
nets	Команда, для манипуляции с сетями, из которых идёт трафик.
route	Анонсирует или удаляет указанный маршрут.
routes	Выводит список всех маршрутов, содержащихся в таблице routes.
update-announce	Производит выгрузку всех маршрутов в таблицу маршрутизации.
user	Создаёт нового пользователя или изменяет существующего.
verify	Выводит статус адресов (анонсирован или нет) для указанного домена.

> У каждой команды есть опция `--help`, которая выводит краткую информацию по команде вместе со списком допустимых аргументов/параметров.

zi-ctl check-announce

Команда выявляет наличие различий между маршрутами, находящимися в локальной таблице маршрутизации, и списком маршрутов, хранящемся в базе данных. В ситуации когда маршрут есть в базе, но он отсутствует в таблице, он заносится в неё в этот же момент. Запускается по крону раз в 5 минут.

```
zi-ctl check-announce
```

zi-ctl configure

Выполняет настройку протокола маршрутизации OSPF для простейшей топологии - SkyDNS Zapret ISP и один маршрутизатор. После вызова команды Вам будет предложено ввести различные параметры, необходимые для настройки OSPF.

```
zi-ctl configure
```

Помимо этого будут созданы конфигурационные файлы для Quagga.

zi-ctl convert

Производит преобразование и последующую запись правил фильтрации, полученных из подключенных источников, в базу данных. Выполняется для тех источников, для которых параметр `enable_parser: true`.

```
zi-ctl convert [OPTIONS]
```

Список опций:

- `--name` - позволяет преобразовать данные конкретного источника (необязательный). По умолчанию в базу заносятся данные со всех включенных источников.

zi-ctl create-zones

Создает stub-зоны на основе записей из таблиц Route и Host. Stub-зоны используются фильтрующим DNS-сервером (Фильтрующий DNS-сервер). Выполняется по `cron` раз в пять минут.

zi-ctl delete-expired

Удаляет маршруты с истекшим ttl. SkyDNS Zapret ISP использует двойной ttl - настоящий, полученный от dns-сервера и фиктивный (увеличенный). Маршруты удаляются в тот момент, когда истекает фиктивный, поэтому пользователь может не волноваться, что удалятся используемые маршруты.

```
zi-ctl delete-expired
```

zi-ctl domain

Команда для управления отдельным доменом и его маршрутами. Анонсирует/удаляет все маршруты, ассоциированные с доменом. При удалении домен и маршруты удаляются из базы данных.

```
zi-ctl domain COMMAND DOMAIN
```

Список параметров:

- `COMMAND` - указывает действие, которое следует применить к заданному домену. Варианты: `announce`/`withdraw` (обязательный).
- `DOMAIN` - указывает с каким доменом будут проводиться манипуляции (обязательный).

`zi-ctl domain announce yandex.ru` - передаст все маршруты ассоциированные с доменом `yandex.ru` на маршрутизатор.

> `zi-ctl domain withdraw yandex.ru` - удалит домен `yandex.ru` и все его маршруты из базы.

zi-ctl download

Производит загрузку всех включенных источников (параметр `enable_downloader: true`). После загрузки происходит запись в базу данных.

```
zi-ctl download [OPTIONS]
```

Список возможных опций:

- `--name` - позволяет загрузить данные из конкретного источника (необязательный). По умолчанию загружаются все включенные источники.

zi-ctl filter

Команда для управления правилами фильтрации.

```
zi-ctl filter COMMAND [ARGS]
```

Список параметров:

- `add` - добавляет запись в черный/белый/серый список. `[ARGS]` - последовательность аргументов из:
    1. LIST_TYPE - тип списка.
    2. BLOCK_TYPE - тип блокировки (URL, домен, IP-адрес).
    3. VALUE - значение.

    См. раздел Списки исключений для дополнительной информации.

- `restore` - восстанавливает черный/белый/серый список из файла. `[ARGS]` - последовательность аргументов из:
    1. LIST_TYPE- тип списка.
    2. FILES- пути к файлам с сохранёнными списками.

- `list` - выводит список всех записей для чёрного/белого/серого списка. Вызвав с параметром `-o`, вывод информации произойдёт в указанный файл.
- `clear` - очищает список по номеру. Чтобы получить списки с их номерами, вызовите команду с параметром `--help`.
- `clean_all` - очищает все таблицы, связанные с процессом фильтрации. Созданные пользователи и добавленные сети останутся.

> `zi-ctl filter add white domain yandex.ru` - добавляет домен `yandex.ru` в белый список.
> `zi-ctl filter list black` - выводит список всех правил Чёрного списка.

zi-ctl ipset

Команда для работы с ipset-ами.

```
zi-ctl ipset COMMAND [OPTIONS]
```

Список параметров:

- `save` - формирует ipset-ы на основе таблиц в бд. Запускается по `cron` раз в 5 минут.
- `update` - пересоздаёт ipset-ы и правила ip(6)tables.

Список опций:

- `--recreate` - команда `zi-ctl ipset update` может быть вызвана с этой опцией, чтобы выполнить полное пересоздание правил ip(6)tables и ipset-ов. Старые правила будут удалены.

zi-ctl nets

Команда для настройки фильтруемых сетей.

```
zi-ctl nets COMMAND [ARGS]
```

Список параметров:

- `list` - выводит список всех фильтруемых сетей.
- `add` - добавляет новые сети. Если вызвана с параметром `--bypass`, трафик из указанных сетей фильтроваться не будет. `[ARGS]` - последовательность сетей.
- `delete` - удаляет указанные сети. `[ARGS]` - последовательность сетей.

`zi-ctl nets add 192.168.0.0/24 192.168.1.0/24` - добавляет сети к фильтруемым.
`zi-ctl nets add 192.168.2.0/24 --bypass` - трафик из сети `192.168.2.0/24` фильтроваться не будет.

zi-ctl route

Анонсирует/удаляет переданный маршрут.

```
zi-ctl route COMMAND ROUTE
```

`zi-ctl route announce 1.1.2.0/24` - передаст маршрут к сети `1.1.2.0/24` на маршрутизатор.
`zi-ctl route withdraw 1.3.4.7/32` - удалит маршрут на IP-адрес `1.3.4.7/32` из таблицы маршрутизации.

Список параметров:

- `COMMAND` - указывает действие, которое следует применить к заданному маршруту. Варианты: `announce` / `withdraw` (обязательный).
- `ROUTE` - указывает с каким маршрутом будут проводиться манипуляции (обязательный).

zi-ctl routes

Выводит на экран список маршрутов, находящихся в базе данных. По умолчанию выводятся маршруты протокола IPv4.

```
zi-ctl routes [OPTIONS]
```

Список опций:

- `--ipv6` - выводятся маршруты протокола IPv6.

`zi-ctl routes --ipv6` - выведет список маршрутов протокола IPv6.

zi-ctl update-announce

Производит выгрузку всех маршрутов в таблицу маршрутизации.

zi-ctl user

Позволяет управлять пользователями - создание новых или изменение параметров существующих.

Список аргументов:

- `list` - выводит список всех существующих пользователей.
- `edit` - добавляет нового пользователя или изменяет реквизиты существующего. Выполняется в интерактивном режиме.

zi-ctl verify

Выполняет проверку какие из адресов домена, переданного команде в качестве аргумента, присутствуют в локальной таблице маршрутизации, а какие нет. Если включена Агрегация маршрутов, поиск будет производится и в агрегированных сетях тоже. Если домен отсутствует в базе, об этом будет сообщено пользователю.

> `zi-ctl verify ya.ru` - выведет статус для каждого из IP адресов Яндекса (отсутствие/наличие в таблице маршрутизации).

Тестовая версия

Отличия от полной версии

В тестовой версии отсутствует возможность загрузки реестра с сайта Роскомнадзора. Однако Вы всё ещё можете использовать список URL, подготовленный SkyDNS, на основе списка экстремистских материалов Министерства юстиции РФ.

При настройке тестовой версии, в разделе Настройка пропустите пункт 5.1.

Переход на полную версию

Для замены тестовой версии на полную версию необходимо:

1. Установите репозиторий SkyDNS с полной версией (см. Установка Репозитория SkyDNS).
2. Удалить пакет skydns-zi-test командой:

```
apt-get remove skydns-zi-test
```

3. Установите полную версию (см Установка и обновление SkyDNS Zapret ISP).
4. Выполнить настройку в соответствии с разделом Настройка.

SkyDNS Zapret Check

Описание

SkyDNS Zapret Check производит имитацию пользователя, который пытается получить доступ к запрещенным ресурсам. Утилита позволяет Вам самостоятельно осуществлять проверку качества фильтрации.

Установка

Для использования утилиты, установите её на отдельный компьютер с Debian 8, который будет имитировать компьютер обычного пользователя. Компьютер обязательно должен располагаться в фильтруемой сети (см. Прием трафика).

1. Установите Debian 8 (см. Установка Debian)
2. Установите репозиторий SkyDNS (см. Установка Репозитория SkyDNS)
3. Установите пакет skydns-zi-check

   ```
   sudo apt-get update
   sudo apt-get install skydns-zi-check
   ```

4. Настройте сетевой интерфейс так, чтобы весь трафик проходил через маршрутизатор, который осуществляет перенаправление трафика на систему фильтрации.

Настройка

1. Настройте SkyDNS Zapret ISP
   Для того, чтобы SkyDNS Zapret Check мог получать актуальные списки фильтрации, нужно добавить авторизационный токен в SkyDNS Zapret ISP. Токен может включать в себя символы английского языка и цифры.
   Добавьте следующую настройку в `/etc/skydns-zi/config.yml`

   ```
   ---

   admin:
     check-api-token: <YOUR_TOKEN>
   ```

   И перезапустите uwsgi

   ```
   sudo service uwsgi restart
   ```

2. Настройте SkyDNS Zapret Check
   Конфигурационный файл `/etc/skydns-zi-check/config.yml`.

Добавьте Ваш токен и IP-адрес интерфейса системы фильтрации, на который маршрутизируется трафик к запрещенным ресурсам:

```
---

admin-url: http://<ZAPRET_ISP_IP_OR_DOMAIN>
admin-token: <YOUR_TOKEN>
```

Если Вы хотите получать сообщения о результатах проверок, заполните соответствующие блоки в конфигурационном файле:

```
# Параметры, необходимое для создания SMTP-сессии.
smtp:
  host: <EMAIL HOSTING>:<PORT(OPTIONAL)>
  username: <USERNAME>
#  password: # OPTIONAL
#  ssl: yes # ENABLE SSL/TLS (OPTIONAL)

# Адреса, на которые будут приходить сообщения
recipients:
#  - <EMAIL_1>
#  - <EMAIL_2>
```

Результаты будут приходить после каждой проверки.

3. Укажите в `/etc/resolv.conf` IP-адрес SkyDNS Zapret ISP.

Запуск

Запуск осуществляется командой:

```
zi-check -t <src> -a <number>
```

`src` - источник, который вы хотите проверить (`rkn`/`mjust`).

`number` - количество случайных записей, которые будут проверены. По умолчанию все.

Получение результатов

Результаты пишутся в лог `/var/log/skydns-zi-check/check-result-<RUN_DATETIME>.log`.

После выполнения проверки в командной строке будет выведено сообщение о её результатах, в частности количество пропусков:

```
-------------------------------------------------------
| Checking for          | rkn                         |
| Time(UTC)             | 2018-07-10 05:00:01.941874 |
| Time(Local)           | 2018-07-10 05:00:01.941883 |
| Total count           | 0                           |
| Not blocked           | 0                           |
| Blocked               | 0                           |
| Unable to get content | 0                           |
```

```
-----------------------------------------------------
Последняя строчка - Количество ресурсов, доступ к которым получить не удалось (например, несущес
```

Если у Вас настроена почта, то на неё придёт сообщение с результатами проверки.

Если Вам потребуется узнать количество пропусков для прошлых проверок, используйте команду:

```
grep FILTERING_ERROR /var/log/skydns-zi-check/check-result-<DESIRED_DATE>.log |  wc -l
```

Продвинутая настройка

Данный раздел является необязательным. Он предназначен для тех, кто хочет максимально настроить систему фильтрации под требования Вашей сети.

Конфигурационный файл

Ниже представлен пример конфигурационного файла со всеми существующими параметрами:

```
---
# Устанавливает уровень логирования.
# Возможные варианты: DEBUG (логируется вся вспомогательная информация), INFO (логируется только
loglevel: DEBUG

# Включает логирование ответов ACL. Увеличивает время ответа ACL на ~20%
log_acl: False

# Содержит системные пути к различным командам.
system_paths:
  # Путь к команде `ipset`.
  ipset: /sbin/ipset
  # Путь к команде `unbound-control`.
  unbound_control: /usr/sbin/unbound-control
  # Путь к команде `iptables`.
  iptables: /sbin/iptables
  # Путь к команде `ip6tables`.
  ip6tables: /sbin/ip6tables
  # Путь к команде `iptables-restore`.
  iptables_restore: /sbin/iptables-restore
  # Путь к команде `ip6tables-restore`.
  ip6tables_restore: /sbin/ip6tables-restore
  # Путь к команде `service`.
  service: /usr/sbin/service

# Включает поддержку IPv6 в системе фильтрации - IPv6 адреса переносятся из источников в базу; д
ipv6_support: true

ip_aggregator:
   # Граница агрегации.
   ip-barrier: 120
   # Включение агрегирования.
   enabled: true

resolver:
  # Количество параллельных подключений к Unbound.
  concurrency: 500
```

```
  # Время, которое прибавляется в реальному `ttl`, чтобы получить фиктивный.
  ttl: 3h
  # Таймаут между циклами разрешения доменов находящихся в базе.
  interval: 5m
  # Анонсировать маршруты посредством exabgp.
  dynamic-routing: true
  # next hop для всех анонсируемых маршрутов, менять только в случае необходимости.
  bgp-next-hop: 8.8.8.8
  # Устанавливает максимальное количество IP-адресов, которые будут заноситься в базу, после раз
  ip-limit: 12
  # Адрес DNS-сервера, через который сервис `zi-update` будет производить разрешение доменных им
  dns-name: 127.0.0.1

admin:
  # Токен, используемый утилитой zi-check для авторизации и получения списка урлов.
  check-api-token: 32b24177-ad41-40ea-af74-45c9e5bdbf4d
  # Следующие значения устанавливают порог, превысив который, в web-интерфейсе на главной страни
  # Допустимое количество дней с момента последней загрузки источника (любого).
  max_time_delta: 2
  # Допустимый процент используемой RAM.
  memory_threshold: 80
  # Допустимый процент используемого своп-файла.
  max_expected_swap: 80
  # Допустимый процент используемого места на диске.
  max_expected_memory: 80

# Каждый модуль загрузки (enricher) состоит из парсера и загрузчика. По умолчанию они оба включе
# enable_parser: false - выключает парсер (занесение информации в бд).
# enable_downloader: false - выключает загрузчик. Парсер всё ещё работает.
enrichers:
   # Мин.Юст.
   mjust:
      name: mjust
      #enable_parser: false
      #enable_downloader: false

   # Белый список Роскомнадзора.
   # Стоит отметить, что он является полуофициальным. В нём находятся домены государственных слу
   wrkn:
      name: wrkn
      enable_parser: false
      enable_downloader: false
      #  Адрес загрузки.
      url: https://storage.googleapis.com/smisc/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA%20%D1%81%D0
      # Время, через которое источник будет загружаться. По умолчанию 24 часа.
      # Пример формата: 12h - каждые 12 часов, 10m - каждые 10 минут.
      period: 24h

   # Список Роскомнадзора.
   rkn:
      name: rkn
      #enable_parser: false
      #enable_downloader: false
```

```
      operator_info:
        # Настроить, если для подписи запросов к API получения выгрузки используется ключ.
        # Этот блок необходимо закомментировать или удалить.
        operatorname: ООО "ОПЕРАТОР"
        inn: '6670123456'
        ogrn: '116670002345'
        email: 'root@loclhost.com'
        # Файлы для создания квалифицированной электронной подписи.
        certificate: /var/lib/skydns-zi/keys/private.pem
        privatekey: /var/lib/skydns-zi/keys/id_rsa.key

      signed_files:
        # Настроить, если для подписи запросов к API получения выгрузки.
        # Используется заранее заготовленный и подписанный запрос.
        # Этот блок необходимо закомментировать или удалить.
        xmlreqfile: /var/lib/skydns-zi/req/request.xml
        signaturefile: /var/lib/skydns-zi/req/request.xml.sig

ipset_config:
   # Обратите внимание, что существуют аналогичные параметры для протокола IPv6 (в тех же блоках
   # Определяет размер хэша для указанных `ipset`.
  hash_size:
    v4_http_black_dst: 65536
    v4_https_black_dst: 65536
    v4_blacklist_nets_dst: 65536
    v4_isp_nets_src: 1024
    v4_isp_bypass_nets_src: 1024
    v4_whitelisted_ips: 1024

   # Устанавливает максимальное количество элементов, которое может содержать указанный `ipset`.
  max_elem:
    v4_http_black_dst: 150000
    v4_https_black_dst: 150000
    v4_blacklist_nets_dst: 150000
    v4_isp_nets_src: 1024
    v4_isp_bypass_nets_src: 1024
    v4_whitelisted_ips: 1024
```

Отключение загрузки и осуществление фильтрации по URL из списка Минюста

По умолчанию с сайта https://www.skydns.ru SkyDNS Zapret ISP скачивает список URL, подготовленный SkyDNS, на основе списка Министерства юстиции РФ. Чтобы не скачивать и не осуществлять фильтрацию по этому списку URL, установите значение параметров `enricher.mjust.downloader: false` `enricher.mjust.parser: false` в `/etc/skydns-zi/config.yml`:

```
mjust:
   downloader: false
   parser: false
```

Удалите связанные с этим списком файлы:

```
rm /var/lib/skydns-zi/src/zi-mj-urllist*
```

> Отключать не рекомендуется, так как через этот источник распространяются поддомены доменов, которые Роскомнадзор блокирует по маске.

Firewall

При установке пакета создаются правила firewall и ipset-ы, необходимые для работы skynds-zi. Если Вам необходимо добавить свои правила или ipset-ы, добавьте их в файлы `/etc/firewall.conf` и `/etc/ipset.conf`.

Содержимое данных файлов должно соответствовать формату выходных данных команд `iptables-save` и `ipset save`. Если Вы добавляете правила любым другим способом, они не будут восстановлены после перезагрузки системы.

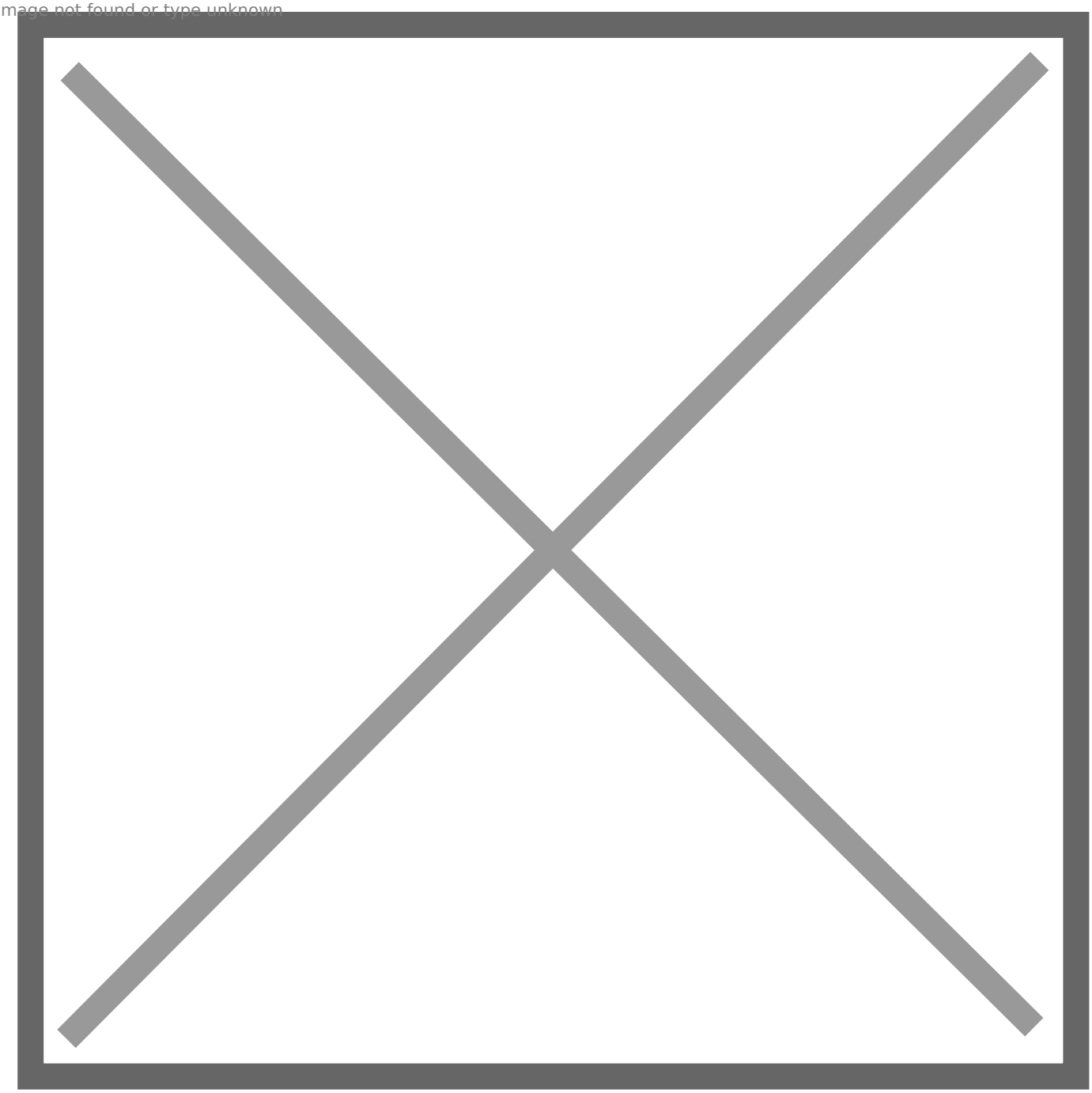
Отказоустойчивость

Рассмотрим различные схемы подключения SkyDNS Zapret ISP в сеть и способы осуществления отказоустойчивости:

1. Динамическая маршрутизация. В случае выхода из строя сервера SkyDNS Zapret ISP, перенаправление исходящего трафика на систему фильтрации осуществляться не будет, таким образом, не пострадает основной сервис сети - предоставление доступа абонентам в сеть Интернет.
2. Статическая маршрутизация. В случае выхода из строя сервера SkyDNS Zapret ISP, необходимо на уровне скрипта предусмотреть очистку таблицы маршрутизации от добавленных статических маршрутов.

Диаграмма ниже демонстрирует способ осуществления отказоустойчивости при внедрении динамической маршрутизации:

Image not found or type unknown



В данном случае было настроено две OSPF-сессии - по одной для каждой из системы фильтрации. В случае отказа одного из серверов, трафик маршрутизировался на второй.

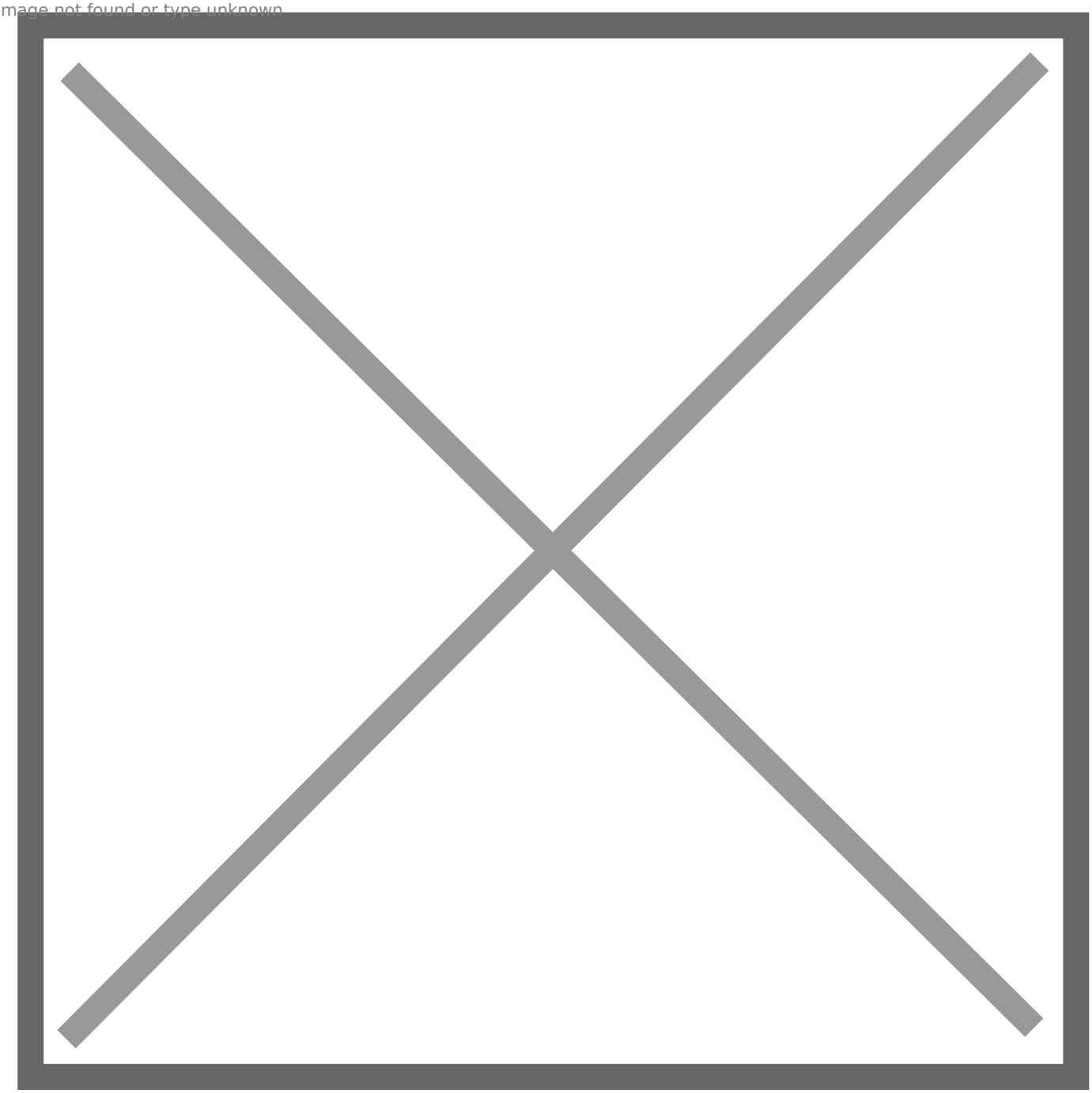
Масштабирование

Для масштабирования решения возможна установка в сеть дополнительного сервера SkyDNS Zapret ISP.

Если Вы используете схему с динамической маршрутизацией - необходимо, чтобы все маршруты, получаемые с серверов фильтрации по OSPF имели одинаковую стоимость маршрута (Equal-cost multi-path routing - ECMP), иначе никакого смысла от дополнительных серверов не будет, так как трафик будет идти через один. По умолчанию протокол OSPF поддерживает до 4 альтернативных путей.

Диаграмма ниже демонстрирует способ осуществления отказоустойчивости при внедрении динамической маршрутизации:

Image not found or type unknown



В данном случае было настроено две OSPF-сессии - по одной для каждой из системы фильтрации.

Настройка балансировки на маршрутизаторе является индивидуальной, согласно документации производителя Вашего маршрутизатора.

Если Вы используете схему со статической маршрутизацией, то для всех дополнительных серверов достаточно будет выполнить шаги, указанные здесь (см Статическая маршрутизация).

В настоящее время единый центр управления серверами отсутствует. Каждый сервер придётся настраивать и обслуживать отдельно. На каждом сервере необходимо будет установить одинаковый конфигурационный файл, продублировать Списки исключений и настроить OSPF (BGP) сессии.

Списки исключений

Черный список

Список ресурсов, которые провайдер должен индивидуально блокировать по судебным решениям. Допустимые типы блокировок: URL, domain, IP-адрес. Изменения вступают в силу с задержкой порядка пяти минут.

Белый список

Список ресурсов, которые провайдер исключает из блокировки. Исключение может быть произведено по: URL, domain, IP-адрес. Имеет более высокий приоритет, чем Черный список. Изменения вступают в силу с задержкой порядка пяти минут.

> Если Вы добавляете домен в Белый список, то все IP-адреса, ассоциированные с ним, станут **белыми**. Таким образом Вы можете противодействовать тем доменам, которые добавляют IP-адреса популярных сайтов к своим (IP-спуфинг).
> Пример: `yandex.ru` разрешился в `77.88.8.8/32`, этот IP-адрес исчезает из анонсов.

Серый список

Список сетей. Данный список позволяет Вам перенаправлять весь трафик, идущий в заданные сети, на сервер фильтрации. Отличие от добавления IP-адреса в Черный список заключается в том, что трафик на IP-адреса из Черного списка блокируется на уровне `iptables`, не достигая ACL. Его приоритет выше Черного списка, но ниже Белого. Изменения вступают в силу с задержкой порядка пяти минут.

> Если Вы хотите незамедлительного применения обновлений, воспользуйтесь командами: zi-ctl domain в случае добавление в список домена или URL, zi-ctl route в случае добавление в список IP-адреса.

Редактирование

Через консоль: zi-ctl filter

Через WEB UI:

Раздел Списки/(Черный|Белый|Серый) список (Создание нового правила)

Продвинутая настройка Squid

Конфигурационный файл `/etc/squid/squid.conf`.

В случае, если Вы наблюдаете проблему излишней блокировки, используйте следующую информационную справку.

Принцип работы Squid: Squid осуществляет балансировку запросов по 2 воркерам каждый из которых форкает по 4 процесса zi_sni_check, которые используются для фильтрации https трафика по имени домена, и zi_url_check, которые используется для фильтрации http трафика.

За это отвечают данные строки конфига:

```
workers 2
external_acl_type zi_sni_check children-max=4 %ssl::>sni sudo -u skydns-zi zi-squid-acl SNI
external_acl_type zi_url_check children-max=4 %DST %PORT %PATH sudo -u skydns-zi zi-squid-acl UR
```

Изменение параметра `workers` изменяет количество процессов Squid-а. Параметр `children-max` отвечает за количество подпроцессов, производящих фильтрацию.

Следует помнить, что каждый подпроцесс имеет свой кэш ответов. Необоснованное увеличение их количества может привести к истощению ресурсов машины, и, как результат, прекращения функционирования системы фильтрации в целом.

Параметры, помогающие решить проблему чрезмерной блокировки:

- `negative_ttl` - определяет время, в течение которого будут храниться отрицательные ответы (блокировка доступа) от подпроцесса фильтрации (значение в секундах, по умолчанию 1 час).
- `cache` - определяет размерность кэша одного подпроцесса (по умолчанию 262144).

Необходимо добавить эти параметры перед `children-max`

После любых изменений нужно перезапустить сервис:

```
service squid restart
```

и обязательно проверить корректность запуска в лог файле `/var/log/squid/cache.log`.

Squid и HTTPS

Для фильтрации HTTPS трафика Squid использует самоподписанный сертификат. По умолчанию вместе с пакетом поставляется самоподписанный сертификат.

`/etc/squid/squidCA.pem` - расположение самоподписанного сертификата.

Если Вам нужно заменить его на свой, то выполните следующие команды:

```
cp <YOUR_CERT> /etc/squid/squidCA.pem
chmod 400 /etc/squid/squidCA.pem
rm -rf /var/lib/squid_ssl_db
/usr/lib/squid/ssl_crtd -c -s /var/lib/squid_ssl_db
chown -R proxy:proxy /var/lib/squid_ssl_db
service squid restart
```

> Ваш сертификат обязательно должен быть самоподписанным.

> Чтобы создать самоподписанный сертификат, следуйте инструкции.

Агрегация маршрутов

SkyDNS Zapret ISP включает в себя функцию уменьшения количества анонсируемых маршрутов посредством сокращения длины префикса анонсируемой сети. По умолчанию данная функция включена (Конфигурационный файл `ip_aggregator.enabled: true`).

Сжатие происходит, когда количество маршрутов из сети превышает заданный барьер (Конфигурационный файл `ip_aggregator.ip-barrier`). Алгоритм учитывает адреса, находящиеся в белом списке.

> Рассмотрим следующий пример: в анонсах присутствуют сети `1.1.1.2/32` и `1.1.1.3/32`. Вы включаете функцию сжатия маршрутов, устанавливая `ip_aggregator.ip-barrier: 1`, после чего анонсируется лишь `1.1.1.0/24`.
> Предположим, Вы добавили в белый список адрес `1.1.1.135/32`. В результате начнут анонсироваться следующие подсети: `1.1.1.0/25`, `1.1.1.192/26`, `1.1.1.160/27`, `1.1.1.144/28`, `1.1.1.136/29`, `1.1.1.128/30`, `1.1.1.132/31`.

В результате агрегации на сервер фильтрации будет перенаправляться дополнительный трафик. Однако это не приведёт к его блокировке. Весь трафик, для которого нет правил фильтрации, будет пропущен дальше без изменений. Агрегация также повлечёт за собой увеличение нагрузки на сетевой интерфейс и процессор сервера, поэтому стоит внимательно подходить к выбору параметра `ip_aggregator.ip-barrier`. Значение по умолчанию - 120, было выбрано исходя из анализа маршрутов, находящихся в базе.

> Чтобы изменения вступили в силу, необходима перезагрузка `ExaBGP` и `zi-update`. Для этого выполните команды:

```
supervisorctl stop zi-update
service exabgp restart
supervisorctl start zi-update
```

Правила iptables и ipset

SkyDNS Zapret ISP поставляет с собой набор базовых правил для обеспечения заворота трафика. Правила содержаться в отдельных файлах, соответствующих входному формату команд `ip(6)tables-restore` и `ipset restore`. Если Вы хотите применить свои собственные правила:

- для `iptables` выполните `cp /usr/share/skydns-zi/firewall.conf /etc/firewall.conf`. После чего добавьте в файл `/etc/firewall.conf` ваши правила.
- для `ip6tables` выполните `cp /usr/share/skydns-zi/firewall6.conf /etc/firewall6.conf`. После чего добавьте в файл `/etc/firewall6.conf` ;ваши правила.
- для `ipset` добавьте правила в `/etc/custom_ipsets.conf`.

После этого выполните:

```
zi-ctl ipset update
```

Фильтрация по протоколу IPv6

Начиная с версии 3.2.3, SkyDNS Zapret ISP предоставляет возможность фильтрации по протоколу IPv6.

> Обратитесь к разделу Фильтрация доступа к ресурсам в реестре по протоколу IPv6 прежде чем начать настройку.

Для включения фильтрации необходимо:

- В конфигурационном файле добавить строчку: `ipv6_support: true`.
- В файле `/etc/quagga/daemons` поменять `ospf6d=no` на `ospf6d=yes`.
- Повторить Настройка маршрутизации (настроить OSPF для IPv6 в случае динамической маршрутизации, обновить скрипт выгрузки маршрутов в случае статической).

- Добавить Ваш IPv6 префикс (Прием трафика), если Вы это не сделали при первоначальной настройке.
- Выполните команду `zi-ctl ipset update`.

Настройка ipset-ов

Начиная с версии 3.2.5, SkyDNS Zapret ISP предоставляет возможность настроить параметры ipset-ов.

Настройка размера ipset-ов, а также допустимый размер хэша устанавливается с помощью блока `ipset_config`. После выставления новых значений, выполните:

```
zi-ctl ipset update --recreate
```

FAQ

При установке или обновлении не запускается супервизор

Известна проблема, когда при установке или обновлении SkyDNS Zapret ISP выдается примерно такая ошибка:

```
Leaving 'diversion of /lib/systemd/system/exabgp.service to /lib/systemd/system/exabgp.service-s
Launching supervisor ...
Job for supervisor.service failed. See 'systemctl status supervisor.service' and 'journalctl -xn
invoke-rc.d: initscript supervisor, action "restart" failed.
Launching uwsgi ...
```

Проблема связана с systemd, В этой ситуации рекомендуется перезапустить супервизор вручную

```
service supervisor stop
service supervisor start
```

Проблема экспорта маршрутов

При подключении к сети провайдера с помощью BGP встречается проблема экспорта маршрутов.

Проблема проявляется следующим образом: трафик до некоторых сайтов идёт мимо системы фильтрации, в таблицах маршрутизации отсутствуют соответствующие префиксы.

Решение проблемы - перезапуск демона exabgp. Необходимо выполнить следующие команды:

```
service exabgp stop
ps aux | grep exabgp
kill -9 #процесса
service exabgp start
supervisorctl restart zi-update
```

При обнаружении проблем с фильтрацией, следует выполнить следующие проверки

Проверка CPU

Может образоваться ситуация, когда какой-то процесс занимает все существующие ресурсы процессора, и, как следствие, не даёт запуститься новым.

Для диагностики этого используйте команду `htop`, после чего нажмите `Shift+h`, чтобы скрыть порожденные процессами потоки. Если Вы наблюдаете большое количество одинаковых процессов с разными `pid`, незамедлительно обратитесь в службу технической поддержки.

Чтобы убрать все процессы, выполнить команду `kill< -9< <pid1> <pid2>`.

Проверка свободного места

Возможна проблема, когда у Вас закончилось свободное место. Для проверки выполните `df -h`, убедитесь, что в разделе, куда установлен SkyDNS Zapret ISP, есть свободное место.

Проверка сети

Возможно, у Вас присутствуют неполадки в сети. В этом случае с сервера фильтрации проверьте наличие связи с граничным маршрутизатором, а также достижимость Ваших DNS-серверов, достижимость популярных сайтов (yandex.ru, vk.com).

Используйте утилиту `ping` для этих целей.

Ошибка загрузки источника в web-интерфейсе

Если Вы наблюдаете такую ошибку, вызовите команду zi-ctl download для незагрузившегося списка.

После этого откройте конец лог-файла (`/var/log/skydns-zi/isp_filter.log`), найдите в нём запись `Downloading/Parsing module <source_name> has finished with exception:` ниже будет `traceback`.

Обратите внимание на ошибку в конце. Возможно, у сервера фильтрации нет доступа к Вашему DNS-серверу (будет присутствовать строка `no route to host`).

Если проблема не в этом, незамедлительно обратитесь в службу технической поддержки.

Избыточное блокирование

В случае ситуации, когда Ваши пользователи не могут получить доступ к неблокируемым ресурсам, например instagram.com, стоит проверить Squid. Выполните `cat /var/log/squid/cache.log | grep <blocked_domain>`. Если вы наблюдаете:

```
queue overload. Using stale result.
# или
queue overload. Request rejected.
```

следуйте инструкциям Продвинутая настройка Squid. В противном случае, проверьте Ваш Чёрный список.

Диагностика причин пропусков

Наличие незначительного количества пропусков (меньше 10) - теоретически возможная ситуация. В случае если такое произошло, следуйте следующей инструкции:

Зайдите на сервер фильтрации.

1. Выберите произвольный домен из списка пропущенных. Выполните команду `dig @127.0.0.2 <domain>`. В ответ должен вернуться ответ, в котором у всех IP-адресов будет одинаковый ttl равный 3600. Если такого не произошло, выполните команды zi-ctl check-announce затем zi-ctl create-zones.
2. Если после проделанных манипуляций, Вы получаете ответ с флагом REFUSED, перезапустите `zi-update` - `supervisorctl restart zi-update`. Убедитесь, что он запустился `supervisorctl status zi-update`.
3. Спустя пять минут, выполните поиск по логам разрешения доменов: `cat /var/log/skydns-zi/updater.log | grep <domain>`. Убедитесь, что домен разрешился в IP-адрес(а). Если домен отсутствует в логе, используйте Поиск заблокированных ресурсов. Если домен присутствует среди блокируемых, выполните zi-ctl check-announce ещё раз.
4. Выполните команду `zi-ctl ipset-create`.

Зайдите на компьютер из фильтруемой сети.

1. Попробуйте получить доступ к пропущенному домену/URL.
2. Если Вам не вернулась страница блокировки, выполните `dig <domain>`.
3. Если Вы получили ожидаемый ответ (как в п.1, находясь на сервере фильтрации), но произошёл пропуск, выполните трассировку по полученному адресу. Если трафик идёт мимо системы фильтрации, у существует другой путь в этому ресурсу (мимо системы фильтрации).
4. Если трафик идёт на SkyDNS Zapret ISP, выполните поиск запрашиваемого ресурса в `/var/log/skydns-zi/acl.log`. Последняя запись должна быть `BLOCKED`. Если Вы видите `PASSED WHITELISTED URL`, значит этот домен/URL находится в Белом списке.
5. Убедитесь, что ресурс находится в списке запрещенных. Для этого выполните `cat /var/lib/skydns-zi/src/dump-latest/dump.hf.xml | grep <domain>/<URL>`, или используйте фильтры в интерфейсе администратора (Поиск заблокированных ресурсов).

Приложение

Данный раздел предназначен для тех, кто хочет лучше разобраться в устройстве работы SkyDNS Zapret ISP.

Схема работы SkyDNS Zapret ISP

Общая информация

SkyDNS Zapret ISP в автоматическом режиме (по умолчанию раз в час) скачивает источники, которые включены в файле конфигурации (по умолчанию список Роскомнадзор и список URL, подготовленный SkyDNS, на основе списка экстремистских материалов Министерства юстиции РФ). Служба SkyDNS Zapret ISP - `zi-update`, постоянно разрешает доменные имена запрещенных ресурсов в IP-адреса. Полученные IP-адреса анонсируются на внутренний маршрутизатор. Трафик, идущий на эти IP-адреса, маршрутизируется на систему фильтрации.

Оставшийся трафик не меняет маршрут следования (см. раздел Схемы подключения SkyDNS Zapret ISP в сеть).

Попавший на SkyDNS Zapret ISP трафик классифицируется следующим образом:

Классификация трафика и действия с ним.

Тип	Действие
Блокируемый http трафик	Возврат страницы блокировки
Блокируемый https трафик	Возвращается tcp-reset
Неблокируемый трафик	Маршрутизируется без изменений

Ответы на пропущенный без изменений трафик идут через сеть напрямую к клиентам (осуществляется асимметричная маршрутизация).

Трафик, попавший на систему фильтрации, проходит через iptables, после чего попадает на Squid. Squid создаёт две своих копии - одна обрабатывает нешифрованный трафик и располагается на 3128 порту, вторая обрабатывает шифрованный и располагается на 3130 порту. Каждая из копий использует списки доступа (ACL), которые исходя из полученных аргументов принимают решение о том, что вернуть на запрос.

Проверки в *ACL*

ACL на 3128 порту (нешифрованный трафик), основываясь на составляющих http запроса: домен, порт, путь, аргументы, принимает решение о том пропустить трафик или вернуть

страницу блокировки.

ACL на 3130 порту (шифрованный трафик), основываясь на составляющих https запроса: SNI, принимает решение о том пропустить трафик или вернуть tcp-reset.

Порядок проверок в ACL:

1. Происходит проверка на то, является ли значение в заголовке Host IP-адресом, а также какой протокол используется. Если HTTP и в заголовке IP - возвращаем страницу блокировки.
2. Происходит поиск набора параметров среди правил фильтрации из Белого списка, если находим - пропускаем запрос.
3. Происходит поиск набора параметров среди правил фильтрации из Черного списка, если находим - возвращаем страницу блокировки.
4. Происходит поиск по маске, если находим - возвращаем страницу блокировки.
5. Пропускаем запрос.

Фильтрация HTTPS и IPv6 и их особенности

Фильтрация HTTPS

Механизм блокировки https трафика:

1. Трафик поступает на систему фильтрации (443 порт), где посредством iptables перенаправляется на 3130 порт. На этом порте слушает Squid, который принимает входящий пакет.
2. Squid использует технологию SSL Bump. Она включает в себя установление двунаправленного защищенного соединение - одно в сторону клиента (Squid представляется запрашиваемым сервером, используя самоподписанный сертификат), другое в сторону сервера.
3. Если обнаруживается попытка установления соединения с блокируемым ресурсом, Squid закрывает туннель в сторону сервера, а клиенту возвращает tcp-reset.
4. Иначе трафик идёт без изменений.

Фильтрация осуществляется только, если ресурс заблокирован по домену. В этом случае используется SNI, на основании которого и осуществляется проверка. Фильтрация по URL в случае https трафика не работает.

Фильтрация доступа к ресурсам в реестре по протоколу IPv6

SkyDNS Zapret ISP поддерживает фильтрацию по протоколу IPv6 но только в том случае, если провайдер предоставляет своим пользователям IPv6 префикс, на основании которого компьютер пользователя генерирует IPv6 адрес.

В случае если провайдер не предоставляет своим пользователям префикс, то последние версии ОС Windows пытаются настроить использование протокола IPv6 поверх IPv4 в автоматическом режиме - происходит создание туннеля IPv6 поверх IPv4. Осуществлять

фильтрацию IPv6 в таком случае невозможно.

Фильтрующий DNS-сервер

В состав SkyDNS Zapret ISP также входит фильтрующий DNS-сервер (Unbound).

Он запущен на `127.0.0.2:53`. Запросы на него перенаправляются посредством `iptables`. Перенаправление происходит если: порт назначения пакета - 53; IP-адрес, с которого пакет был отправлен, находится в фильтруемых сетях. Если IP-адрес источника находится в нефильтруемых сетях (сети, добавленные командой zi-ctl nets с опцией `--bypass`), запрос перенаправляется на обычный DNS-сервер, который располагается на `127.0.0.1:53`.

Принцип работы: на основании таблиц с хостами и с IP-адресами формируются stub-зоны, тем самым гарантируется возврат того IP-адреса, который присутствует в таблице маршрутизации роутера. Такой подход гарантирует 100% перенаправление запросов к заблокированным ресурсам.

Кэш в Squid

В конфигурационном файле, поставляемом вместе с пакетом, кэш отключен.

```
cache deny all
```

Отключает кэширование данных, полученных от web-серверов (html страницы, иконки, картинки).

```
cache_dir aufs /var/spool/squid 20000 49 256
```

Папка, в которую будет помещаться кэш. Первое число - размер дискового кэша в мегабайтах. Второе число - количество поддиректорий первого уровня. Третье число - количество поддиректорий третьего уровня.

Squid использует ~1 Gb RAM, чтобы адресовать 10 Gb кэша.

```
maximum_object_size 32768 KB
minimum_object_size 3 KB
```

Размер максимального и минимального файлов в кэше.

```
cache_swap_low 80
cache_swap_high 90
```

Определяет процент заполнения кэша, после которого Squid будет его очищать.

```
cache_mem 2500 MB
maximum_object_size_in_memory 512 KB
```

`cache_mem` - объём RAM, который будет использоваться для кэширования.

`maximum_object_size_in_memory` - соответствует параметру `maximum_object_size`, но для RAM.

> cache_mem - не жесткое ограничение для всего RAM, которое будет использовано Squid-ом. Squid использует помимо этого другие кэши, плюс на внутренние нужды.

Таблицы базы данных

Описание таблиц базы данных.

В базу данных входят следующие таблицы:

- filter - содержит правила фильтрации. Ресурс может быть заблокирован по IP-адресу, URL, доменному имени, маски домена.
- route - содержит маршруты, ассоциированные с доменами из таблицы host. В эту же таблицу попадают маршруты, поставляемые вместе с источником (например, IP-адреса, которые заблокировал Роскомнадзор).
- host - содержит доменные имена записей из таблицы filter.
- src_net - содержит сети, из которых на систему фильтрации будет поступать трафик.
- user - содержит данные об учетных записях пользователей web-интерфейса.