

Содержание

FastBypass monitor	3
Требования к оборудованию	3
Основные функции	3
Установка	4
Использование	4
Локальное и глобальное состояние. Режим Bypass	5
Конфигурация	6
Минимальная конфигурация	6
Общая конфигурация	7
Конфигурация Listener	9
Конфигурация интерфейсов для сетевых карт Bypass	10
Утилита <i>fbypass_ctl</i>	11
Доступные команды	11

FastBypass monitor

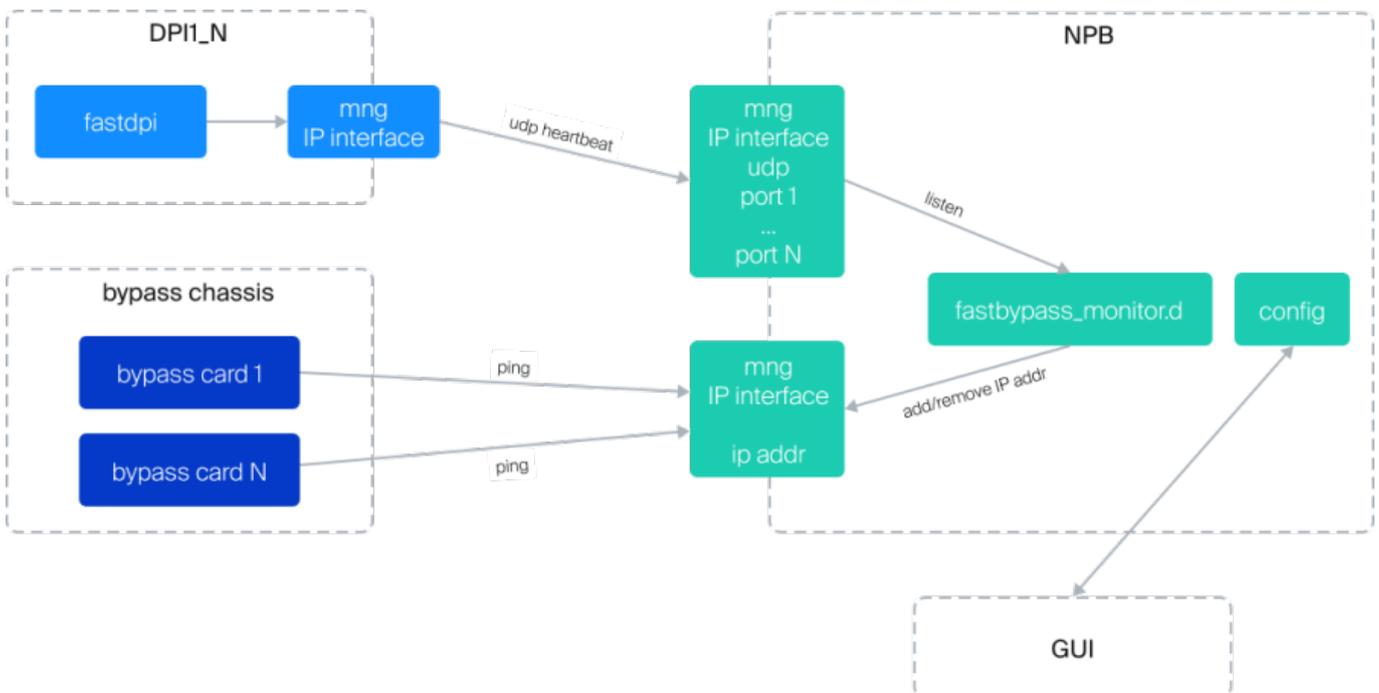
В случае если на DPI произошел сбой ПО, NPB выводит DPI из стека и NPB перебалансирует нагрузку между остальными DPI.

Если падает более 2 DPI узлов — вся система переходит в режим bypass.

Если падает линк на DPI, то NPB перебалансирует нагрузку между остальными DPI.

`fastbypass_monitor` (далее в документации и скрипте называется "демон") представляет собой инструмент для отслеживания и управления состоянием сетевых интерфейсов, к которым подключены сетевые карты Bypass.

Демон реагирует на HEARTBEAT сигналы, поступающие от DPI на определенные порты, указанные в конфигурационном файле. При не получении HEARTBEAT сигналов в соответствии с правилами из конфигурации, демон выполняет определенные действия, такие как удаление или создание IP-адресов, к которым подключены карты Bypass, а также включение или выключение определенных сетевых интерфейсов.



Требования к оборудованию

OS: OpenSwitch 2+ / Debian 9+

Python: 2.7.9

Основные функции

- Отслеживание HEARTBEAT сигналов от DPI на заданных портах.
- Динамическое управление IP-адресами и сетевыми интерфейсами.

Установка

1. Скопировать установочный пакет `fastbypass_monitor-X.X.XX.deb` на хост машину.
2. Из директории, где находится установочный пакет, выполните команду установки:

```
sudo dpkg -i fastbypass_monitor-X.X.XX.deb
```

После установки демон становится доступным для управления с помощью системного менеджера (`systemctl`).

Файл конфигурации доступен по пути `/var/fastbypass_monitor/backend/.env`

Файл с примером полной конфигурации доступен по пути

`/var/fastbypass_monitor/backend/sample.env`

Логи демона сохраняются по пути `/var/fastbypass_monitor/backend/logs/`

Использование

После установки демон автоматически запускается и находится в состоянии работы. При перезагрузке хост-машины демон автоматически запускается после успешного запуска сервиса `network`.

Управление демоном осуществляется командами системного менеджера.

Чтобы запустить демон, выполните следующую команду:

```
sudo systemctl start fastbypass_monitor
```

Чтобы перезапустить демон, выполните следующую команду:

```
sudo systemctl restart fastbypass_monitor
```

Чтобы перезагрузить без остановки демона, выполните следующую команду:

```
sudo systemctl reload fastbypass_monitor
```

Чтобы остановить демон, выполните следующую команду:

```
sudo systemctl stop fastbypass_monitor
```

Чтобы проверить текущий статус демона, выполните следующую команду:

```
sudo systemctl status fastbypass_monitor
```

Для настройки и запуска демона с новой конфигурацией необходимо отредактировать конфигурационный файл и затем выполнить перезагрузку демона или остановить и снова запустить его.

Для настройки параметров работы демона, используется конфигурационный файл, расположенный по пути `/var/fastbypass_monitor/backend/.env`

При запуске и перезагрузке, демон читает конфигурационный файл, и в случае, если он может успешно настроить указанные IP-адреса и интерфейсы, то продолжает свою работу. Однако, если часть конфигурации не прошла проверку, демон запускает только те компоненты, которые могут быть успешно настроены.

В случае возникновения критической ошибки демон автоматически перезагружается.



При использовании команды `sudo systemctl reload fastbypass_monitor` демон перечитывает конфигурацию без остановки своей работы, и останавливает работу удаленных компонентов и создает новые добавленные компоненты, указанные в конфигурации.

Важно отметить, что при запуске и перезагрузке демон не управляет указанными в конфигурации интерфейсами и IP-адресами до того момента, пока не получит состояние всех listener.

При перезапуске демон продолжает находиться в том же состоянии, как и до перезапуска, пока не получит новое состояние от всех listener.

Локальное и глобальное состояние. Режим Bypass

Демон поддерживает два вида управления интерфейсами в зависимости от их состояния: **глобального** (основанного на работе всех listener) или от **локального** (связанного с конкретным listener).

Например, если указать список интерфейсов для управления в глобальных настройках, то их включение или отключение зависит от общего состояния демона. Это означает, что если демон не получает сигналов от достаточного количества listener, то указанные интерфейсы отключаются.

Пример:

```
LISTEN_CUBRO_IFS=<список интерфейсов>  
LISTEN_SHUTDOWN_CUBRO_IFS_WHEN_BYPASS=1
```

Также для каждого listener можно указать свой собственный список интерфейсов, который будет управляться в зависимости от состояния этого конкретного listener.

Пример:

```
LISTEN_CUBRO_IFS[0]=<список интерфейсов>  
LISTEN_SHUTDOWN_CUBRO_IFS_WHEN_BYPASS[0]=1
```

Если интерфейсы, указанные в `LISTEN_CUBRO_IFS[N]` дублируются у нескольких listener, то они переходят в bypass mode в случае если один из соответствующих listener перестает

получать сигнал. Переход интерфейсов в режим normal осуществляется только если все соответствующие listener получают сигналы.

В случае, если интерфейсы указаны как в локальных, так и глобальных настройках, то интерфейсы находятся в режиме bypass mode, пока соответствующий listener не начнет получать сигналы, а демон не перейдет в режим normal mode.

Конфигурация

Минимальная конфигурация

Минимальная конфигурация для работы демона включает указание хотя бы одного интерфейса, IP-адреса и порта для приема HEARTBEAT сигналов, а также одного интерфейса и IP-адреса для подключения карты Bypass.

Пример:

```
# уровень логирования - сообщения об ошибках и информационные сообщения
LOG_LEVEL=INFO

# интерфейс для работы listener по умолчанию
LISTEN_HEARTBEAT_IFS=eth0
# интерфейс для работы с картами Bypass по умолчанию
BYPASS_CARD_IFS=eth0

# количество неудачных HEARTBEAT listener для перехода в режим Bypass
LISTEN_HEARTBEAT_FAILED=1
# количество попыток получения HEARTBEAT сигнала по умолчанию для listener
LISTEN_HEARTBEAT_ATTEMPTS=1
# время ожидания HEARTBEAT сигнала в миллисекундах по умолчанию для listener
LISTEN_HEARTBEAT_TIMEOUT=3000

# IP-адрес, на котором демон ожидает HEARTBEAT сигналы
LISTEN_HB_HOST[0]=192.168.1.202
# порт, на котором демон ожидает HEARTBEAT сигналы
LISTEN_HB_PORT[0]=3000

# IP-адрес, на котором демон ожидает HEARTBEAT сигналы
LISTEN_HB_HOST[1]=192.168.1.202
# порт, на котором демон ожидает HEARTBEAT сигналы
LISTEN_HB_PORT[1]=3100

# IP-адрес карты Bypass, к которой демон будет подключен
BYPASS_CARD_HOST[0]=192.168.1.211

# IP адрес карты Bypass, к которой демон будет подключен
BYPASS_CARD_HOST[1]=192.168.1.212
```

Выше приведен пример конфигурации для получения HEARTBEAT сигналов с использованием

интерфейса eth0 на IP-адресе 192.168.1.202 и портах 3000 и 3100

Вурасс карты подключены по интерфейсу eth0 на IP-адреса 192.168.1.211 и 192.168.1.212.

По умолчанию для всех listener установлены значения:

LISTEN_HEARTBEAT_ATTEMPTS — количество попыток получения сигнала: 1

LISTEN_HEARTBEAT_TIMEOUT — время ожидания сигнала: 3000 миллисекунд

Когда listener не получает сигнал после одной попытки в течение 3000 миллисекунд, он считается неудачным.

Если количество неудачных слушателей равно или превышает установленный порог (LISTEN_HEARTBEAT_FAILED), демон переходит в режим Вурасс и удаляет указанные для Вурасс карт IP-адреса.

При восстановлении сигналов, listener считается рабочим.

Если общее количество неудачных слушателей становится меньше порога, то демон возвращается в режим NORMAL и восстанавливает указанные IP-адреса для Вурасс карт.

Общая конфигурация

Приведенные ниже настройки конфигурации распространяются на всё функционирование демона в целом.

Блок с настройками listener служит в качестве значений по умолчанию для тех listener, для которых не указаны соответствующие конфигурации. То же самое относится и к настройкам интерфейсов для подключения сетевых карт Вурасс.

Кроме того, демон предоставляет возможность подключать свои собственные команды для управления интерфейсами и IP-адресами. Это позволяет адаптировать демон под особенности сети и внедрить собственные скрипты, оптимизированные под специфические требования сетевого окружения.

```
# уровень логирования (необязательная настройка, по умолчанию соответствует INFO) :  
# INFO - сообщения об ошибках и информационные сообщения  
# DEBUG - сообщения об ошибках, информационные сообщения и сообщения отладки  
LOG_LEVEL=
```

```
# маска сети для указанных IP адресов (необязательная настройка, по умолчанию - 32)  
//NETWORK_MASK=//
```

```
# интерфейс по умолчанию для тех listener у которых не указано имя интерфейса для  
прослушивания HEARTBEAT (необязательная настройка)  
LISTEN_HEARTBEAT_IFS=
```

```
# IP-адрес по умолчанию для тех listener у которых не указан IP-адрес для  
прослушивания HEARTBEAT сигналов (необязательная настройка)  
//LISTEN_HEARTBEAT_HOST=//
```

количество попыток получения HEARTBEAT сигнала по умолчанию для тех listener у которых не указана соответствующая конфигурация (необязательная настройка, по умолчанию - 3)

```
///LISTEN_HEARTBEAT_ATTEMPTS=//
```

время ожидания HEARTBEAT сигнала в миллисекундах по умолчанию для тех listener у которых не указана соответствующая конфигурация (необязательная настройка, по умолчанию - 3000)

```
///LISTEN_HEARTBEAT_TIMEOUT=//
```

количество неудачных HEARTBEAT listener для перехода в режим Bypass (необязательная настройка, по умолчанию - 1)

```
///LISTEN_HEARTBEAT_FAILED=//
```

интерфейс для работы с картами Bypass по умолчанию для карт, у которых не указано имя интерфейса (необязательная настройка)

```
BYPASS_CARD_IFS=
```

список интерфейсов для выключения при переходе в режим Bypass (необязательная настройка)

```
LISTEN_CUBRO_IFS=
```

настройка работы интерфейсов указанных в //LISTEN_CUBRO_IFS// (необязательная настройка, по умолчанию - 0)

1 - выключить указанные интерфейсы при переходе в режим Bypass

0 - не выполнять действий с указанными интерфейсами при переходе в режим Bypass

```
LISTEN_SHUTDOWN_CUBRO_IFS_WHEN_BYPASS=
```

абсолютный путь к скрипту для включения интерфейса (необязательная настройка, по умолчанию - см. примечание 2)

```
CMD_SET_UP_INTFS=
```

абсолютный путь к скрипту для отключения интерфейса (необязательная настройка, по умолчанию - см. примечание 2)

```
CMD_SET_DOWN_INTFS=
```

абсолютный путь к скрипту для добавления IP адреса(необязательная настройка, по умолчанию - см. примечание 2)

```
CMD_ADD_IP=
```

абсолютный путь к скрипту для удаления IP адреса (необязательная настройка, по умолчанию - см. примечание 2)

```
CMD_DEL_IP=
```

Примечание 1



Все IP-адреса, указываемые в конфигурации, могут быть представлены в формате 192.168.1.202 или с указанием маски сети, например, 192.168.1.202/16. По умолчанию маска сети равна 32 (в случае, если не

указана в глобальной настройке NETWORK_MASK или в соответствующей конфигурации и listener или карты Bypass).



Важно учитывать следующее: если IP-адрес совпадает с IP-адресом управления (Management IP-адрес), который используется для SSH соединения, то маска сети не меняется и остается такой, как была указана в операционной системе.

Это важное условие следует учитывать при настройке IP-адресов в конфигурации, чтобы избежать конфликтов с управляющим IP-адресом и ненамеренных изменений маски сети.

Примечание 2

Конфигурация демона предоставляет возможность указывать оригинальные скрипты для выполнения базовых операция таких как включение/выключение интерфейсов и создания/удаления IP-адресов.

Демон ожидает получить в соответствующей конфигурации абсолютный путь до shell скрипта с указанием используемых переменных в конце строки в формате %(<имя переменной>)s

Используемые переменные:



- `intfs` — имя интерфейса
- `ip` — IP адрес
- `netmask` — маска сети

По умолчанию используются следующие конфигурации:

- `CMD_SET_UP_INTFS=/var/fastbypass_monitor/backend/app_bash/cmd_set_up_intfs.sh %(intfs)s`
- `CMD_SET_DOWN_INTFS=/var/fastbypass_monitor/backend/app_bash/cmd_set_down_intfs.sh %(intfs)s`
- `CMD_ADD_IP=/var/fastbypass_monitor/backend/app_bash/cmd_add_ip.sh %(ip)s %(netmask)s %(intfs)s`
- `CMD_DEL_IP=/var/fastbypass_monitor/backend/app_bash/cmd_del_ip.sh %(ip)s %(netmask)s %(intfs)s`

Конфигурация Listener

Каждый listener обеспечивает возможность получения HEARTBEAT сигналов от различных устройств DPI. Каждый последующий listener указывается в конфигурации со следующим индексом (например, [0], [1], [2]).

Listener имеет следующие параметры для полной конфигурации:

```
# listener идентификатор (необязательная настройка, по умолчанию соответствует
```

```
индексу)
LISTEN_HB_ID[0]=0

# имя интерфейса, на котором listener ожидает HEARTBEAT сигналы
LISTEN_HB_IFS[0]=eth0

# IP адрес/маска подсети для прослушивания HEARTBEAT сигналов
LISTEN_HB_HOST[0]= 192.168.1.202/32

# порт для прослушивания HEARTBEAT сигналов
LISTEN_HB_PORT[0]=3000

# количество попыток получения HEARTBEAT сигнала (необязательная настройка, по
умолчанию: 3)
LISTEN_HB_ATTEMPTS[0]=3

# время ожидания HEARTBEAT сигнала в миллисекундах (необязательная настройка, по
умолчанию: 3000)
LISTEN_HB_TIMEOUT[0]=3000

# настройка немедленного переключения в Bypass режим (необязательная настройка)
# 1 - если HEARTBEAT сигнал не получен, немедленно перейти в режим Bypass
# 0 - если HEARTBEAT сигнал не получен, перейти в режим Bypass после всех попыток
получить сигнал (по умолчанию)
LISTEN_HB_SWITCH_IMMEDIATELY[0]=0

# список интерфейсов для выключения при переходе в режим Bypass (необязательная
настройка)
LISTEN_CUBRO_IFS[0]=e101-001-0,e101-002-0

# настройка выключения интерфейсов указанных в настройке LISTEN_CUBRO_IFS[N]
(необязательная настройка)
# 1 - выключить указанные интерфейсы при переходе в режим Bypass
# 0 - не выполнять действий с указанными интерфейсами при переходе в режим Bypass (по
умолчанию)
LISTEN_SHUTDOWN_CUBRO_IFS_WHEN_BYPASS[0]=1
```

Конфигурация интерфейсов для сетевых карт Bypass

Демон автоматически управляет (удаляет/создает) IP-адресами на соответствующих интерфейсах при переходе в режим BYPASS или NORMAL в соответствии с настройками карт Bypass.

Для каждой Bypass карты в конфигурации указывается следующий индекс (например, [0], [1], [2]).

Интерфейсы для карт Bypass имеют следующие параметры для полной конфигурации:

```
# идентификатор карты Bypass (необязательная настройка, по умолчанию соответствует
индексу)
BYPASS_CARD_ID[0]=
```

```
# IP адрес/маска подсети для прослушивания HEARTBEAT сигналов
BYPASS_CARD_HOST[0]=

# режим работы карты Bypass
# 0 - удалить указанный IP адрес при включении режима Bypass
# 1 - удалить указанный IP адрес при включении режима Bypass, добавить обратно при
выключении режима Bypass (по умолчанию)
BYPASS_CARD_ACTIVE[0]=

# форсированный режим карты Bypass
# 0 - отключить форсированный режим карты (по умолчанию)
# 1 - включить форсированный режим, IP адрес карты работает вне зависимости от
состояния демона
BYPASS_CARD_FORCE[0]=
```

Утилита `fbypass_ctl`

Утилита `fbypass_ctl` содержит набор команд для включения / выключения `bypass`, а также для запуска сервиса `fastbypass_monitor` и получения статуса.

Утилита `fbypass_ctl` представляет собой алиас на `bash`-скрипт. Использование утилиты возможно только в режиме `sudo`. Перед началом использования утилиты введите команду `sudo su -` и пароль.

Доступные команды

Команда `fbypass_ctl force_on`

Останавливает службу `fastbypass_monitor`, удаляет IP-адреса, указанные в конфиге для `bypass` карт. Тем самым принудительно переводит систему в состояние `bypass`.

Команда `fbypass_ctl force_off`

Останавливает службу `fastbypass_monitor`, добавляет IP-адреса, указанные в конфиге для `bypass` карт. Тем самым принудительно переводит систему в состояние `normal`.

Команда `fbypass_ctl start`

Запускает службу `fastbypass_monitor`. Служба запускается в состоянии `unknown`, т.е. не включает и не выключает `bypass`. После поднятия всех ресиверов и определения их состояния, система переходит в режим `normal` или `bypass` в зависимости от конфигурации и состояния ресиверов.

Команда `fbypass_ctl restart`

Перезапускает службу `fastbypass_monitor`.

Команда `fbypass_ctl status`

Отображает статус службы `fastbypass_monitor`, а также вывод команды `ip a`

Команда `fbypass_ctl enable`

Добавляет службу `fastbypass_monitor` в автозагрузку.

Команда `fbypass_ctl disable`

Удаляет службу `fastbypass_monitor` из автозагрузки.

Команда `fbypass_ctl tail 100`

Выводит последние 100 строк лога службы `fastbypass_monitor`. Сам лог находится по пути `/var/fastbypass_monitor/backend/logs/fastbypass_monitor.log`

Команда `fbypass_ctl tailf`

Алиас команды `tail -f`
`/var/fastbypass_monitor/backend/logs/fastbypass_monitor.log`