Table of Contents

Вопросы администрирования

1. Как узнать текущий релиз (CCC)?

Командой

```
fastdpi -re
```

2. Как узнать текущую версию?

Командой

```
fastdpi -ve
```

3. Как откатиться на предыдущую версию?

Пример отката с 2.7 версии на 2.6:

```
yum downgrade fastdpi-2.6
```

4. Что означает ошибка 'error loading DSCP settings, res=-4'?

Ошибка выводится из-за отсутствия DSCP по автономным системам. Можно проигнорировать.

5. Что делать в случае, если не всегда все команды обрабатываются и появляется ошибка 'ERROR : Can't connect to 127.0.0.1:29000, errcode=99 : Cannot assign requested address Autodetected fastdpi params : dev='lo', port=29000 connecting 127.0.0.1:29000 ...'?

`fdpi_ctrl` для общения с DPI использует обычный Linux стек, так что рекомендации по тюнингу аналогичны как для WEB-серверов (типа nginx) под высокой нагрузкой
Настройки подобны для nginx, которые рекомендуют вставить в файл `/etc/sysctl.conf` (для того чтобы они сохранились при перезагрузке):

```
# Оптимизация работы сетевого стека ОС
net.core.netdev_max_backlog=10000
net.core.somaxconn=262144
net.ipv4.tcp_syncookies=1
net.ipv4.tcp_max_syn_backlog = 262144
net.ipv4.tcp_max_tw_buckets = 720000
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_tw_reuse = 1
net.ipv4.tcp_fin_timeout = 30
net.ipv4.tcp_keepalive_time = 1800
net.ipv4.tcp_keepalive_probes = 7
net.ipv4.tcp_keepalive_intvl = 30
net.core.wmem_max = 33554432
net.core.rmem_max = 33554432
net.core.rmem_default = 8388608
net.core.wmem_default = 4194394
net.ipv4.tcp_rmem = 4096 8388608 16777216
```

```
net.ipv4.tcp_wmem = 4096 4194394 16777216
```

для 1Гбит интерфейса:

```
net.core.netdev_max_backlog=10000
```

для 10Гбит интерфейса:

```
net.core.netdev_max_backlog=30000
```

Чтобы не делать ребут, их можно изменить на лету, применив команду

```
sysctl -w <настройка>
```

Например:

```
sysctl -w net.ipv4.tcp_tw_reuse=1
```

Это должно решить проблему.

Для CentOS 7
Пример:

```
# Оптимизация работы сетевого стека ОС
net.core.netdev_max_backlog=65536
net.core.optmem_max=25165824
net.core.somaxconn=1024
net.ipv4.tcp_max_orphans = 60000
net.ipv4.tcp_no_metrics_save = 1
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_sack = 1
net.ipv4.tcp_syncookies=1
net.ipv4.tcp_max_syn_backlog = 262144
net.ipv4.tcp_max_tw_buckets = 720000
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_tw_reuse = 1
net.ipv4.tcp_fin_timeout = 30
net.ipv4.tcp_keepalive_time = 1800
net.ipv4.tcp_keepalive_probes = 7
net.ipv4.tcp_keepalive_intvl = 30
net.core.wmem_max = 33554432
net.core.rmem_max = 33554432
net.core.rmem_default = 8388608
net.core.wmem_default = 4194394
net.ipv4.tcp_rmem = 4096 8388608 16777216
net.ipv4.tcp_wmem = 4096 4194394 16777216
```

Команда обновления:

```
sysctl –system
```

Дополнительная информация по CentOS7

Скрипты для миграции из SCE SM в БД СКАТ, описание внутри

6. Как посмотреть загрузку по ядрам и понять, почему они загружены неравномерно?

Для просмотра загрузки процессора по ядрам в утилите `top` нажмите 1.
Для просмотра загрузки по задачам DPI выполните команду:

```
ps -p `pidof fastdpi` H -o %cpu,lwp,pri,psr,comm
```

Пример вывода:

```
 %CPU   LWP PRI PSR COMMAND
0.0  23141  41   0 fastdpi_main
0.0  23146  41   0 fastdpi_dl
0.3  23147  41   0 fastdpi_ctrl
35.8 23148  41   0 fastdpi_ajb
32.7 23152  41   1 fastdpi_rx_1
34.1 23165  41   2 fastdpi_wrk0
34.1 23170  41   3 fastdpi_wrk1
```

В DPI задачи COMMAND функционально разделены по ядрам PSR, чтобы не мешать работе друг друга:

1. потоки `wrk` выполняют анализ данных в сетевых пакетах
2. поток `rx` отвечает за транзит данных между сетевыми портами
3. остальные потоки выполняют прикладные и вспомогательные задачи (генерация Netflow, прием управляющих команд, загрузка списков, запись PCAP и т.п.) и могут создавать пиковые нагрузки на CPU, поэтому вынесены на отдельное ядро.

7. Что делать в случае ошибки в fastdpi_alert.log '[CRITICAL][2017/10/06-16:36:44:616019][0x7fdb297ac700] metadata_storage : Can't allocate memory [repeat 1], cntr=188889, allocated=188889'?

В DPI все предварительно аллоцировано, по умолчанию на приведенное в ошибке количество абонентов (188889). Это регулируется параметром в конфигурации `mem_ip_metadata_recs`.
Например, для увеличения до 500000 абонентов поставьте в конфигурации `/etc/dpi/fastdpi.conf`:

```
mem_ip_metadata_recs=500000
```

После изменения параметра потребуется рестарт:

```
service fastdpi restart
```

8. Какие файлы рекомендовано архивировать?

```
cp /etc/dpi /BACKUPDIR/etc/
```

```
mdb_copy /var/db/dpi /BACKUPDIR/db/
```

С mdb_copy можно делать бекап при работающем fastDPI.

9. Что делать в случае, если ipmi задействует 100% CPU и мешает работе DPI?

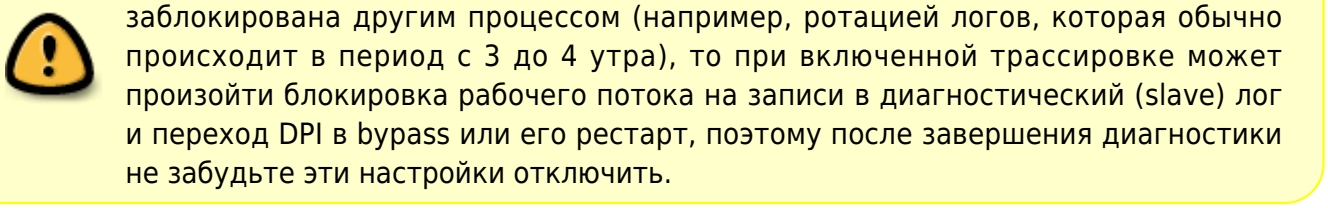
Выполните команду

```
echo 100 > /sys/module/ipmi_si/parameters/kipmid_max_busy_us
```

Чтобы настройка не потерялась при перезагрузке сервера, эту команду можно добавить в /etc/rc.local

10. Что делать в случае, если возникла ошибка в алерт логе '[ERROR] bpm : thread #1 - does not change self-monitoring counters', DPI рестартовал и образовалась корка (или перешел в bypass)?

DPI в процессе работы производит самодиагностику и если один рабочих потоков завис и больше не может проводить обработку трафика, то DPI детектирует это состояние и перезапускается с генерацией корки по сигналу Abort.



Важно: trace и dbg настройки в fastdpi.conf предназначены для диагностики и отладки, а не для постоянной работы, в частности если запись на диск заблокирована другим процессом (например, ротацией логов, которая обычно происходит в период с 3 до 4 утра), то при включенной трассировке может произойти блокировка рабочего потока на записи в диагностический (slave) лог и переход DPI в bypass или его рестарт, поэтому после завершения диагностики не забудьте эти настройки отключить.

Проблема проявляется только на некоторых серверах и если ваш сервер попал в это число, то рекомендуем изменение стандартного дискового планировщика на deadline:

```
echo deadline > /sys/block/sda/queue/scheduler
echo deadline > /sys/block/sdb/queue/scheduler
```

11. Почему в процессе работы растет память, потребляемая процессом

DPI выделяет память статически: при старте процесса и в момент создания некоторых профилей услуг (таких как NAT, черные и белые списки), в процессе работы дополнительная память не выделяется. Почему же тогда растет потребление?
ОС Linux различает резидентную (обозначена в top как RES) и виртуальную (обозначена в top как VIRT) память процесса. Особенность в том, что пока память не инициализирована (фактически инициализирована нулем), то она не записывается Linux в резидентную и перемещается туда по мере ее инициализации.
Настройкой mem_preset=1 в /etc/dpi/fastdpi.conf можно указать, чтобы DPI инициализировал всю выделенную память (точнее почти всю), тогда размер резидентной части не будет расти по мере работы, но этот вариант замедляет старт и хорош когда физической оперативной памяти достаточно, поэтому лучше просто учитывать этот фактор и следить отдельно за расходом виртуальной памяти (VIRT) и резидентной (RES).

12. Что делать в случае, если на одном из СКАТ много "зомби" процессов с именами ''wd_*''?

```
166206 ?        Z      0:00  \_ [wd_fastdpi.sh] <defunct>
166219 ?        Z      0:00  \_ [wd_fastpcrf.sh] <defunct>
```

Достаточно перезапустить watchdog:

```
service watchdog restart
```

13. Проблема детектирования протоколов или сигнатур

В случае проблем детектирования протоколов или сигнатур необходимо выполнить по три теста на каждом из перечисленных устройств:

- персональный компьютер
- смартфон на операционной системе IOS
- смартфон на операционной системе Android

Следующие рекомендации позволят избавиться от лишнего трафика:

- тест на ПК рекомендуется проводить в браузере в режиме инкогнито
- выполняя тест на смартфоне, необходимо включить на нем режим экономии энергии

Выполнение теста:

1. Проверьте, включены ли в файле `/etc/dpi/fastdpi.conf` такие параметры, как:

   ```
   trace_ip=“ip абонента”
   ajb_save_ip=“ip абонента”
   plc_trace_ip=“ip абонента”
   ```

   Если какой-то из этих параметров включен – закомментируйте его и сделайте `service fastdpi reload`.

2. Выполните команду

   ```
   find /var/log/dpi -type f -name "fastdpi_slave_*.log" -exec sh -c 'cat /dev/null > {}'  \;
   ```

   Команда должна удалить данные из файлов `fastdpi_slave_*.log`.

3. Удалите все файлы из `/var/dump/dpi/`.
4. Откройте в текстовом редакторе файл `/etc/dpi/fastdpi.conf`. Добавьте в файл параметры:

   ```
   trace_ip=“ip абонента”
   ajb_save_ip=“ip абонента”
   plc_trace_ip=“ip абонента” #Для работы этого параметра на тестовом абоненте должен быть установлен профиль полисинга
   ```

5. Подготовьте тестового абонента к запуску, чтобы сгенерировать проблемный трафик.
6. Сделайте `service fastdpi reload`.
7. Начните генерировать трафик. Записывайте трафик в течение 1 минуты.
8. Откройте файл `fastdpi.conf`. Закомментируйте параметры:

```
trace_ip=“ip абонента”
ajb_save_ip=“ip абонента”
plc_trace_ip=“ip абонента”
```

9. Сделайте `service fastdpi reload`.
10. Подготовьте вывод следующих команд в файлы:

```
“fastdpi -ve”
“dscp2lst /etc/dpi/protocols.dscp”
“fdpi_ctrl list --policing --ip “ip абонента”
“dscp2as /etc/dpi/asnum.dscp”.
```

11. Подготовьте архив с файлами из пункта 10, а также с файлом `fastdpi.conf`.
    Из `/var/log/dpi` — `fastdpi_stat.log`, `fastdpi_slave_*.log`.
    Из `/var/dump/dpi` — `udp_*.pcap`.
12. Повторите необходимое количество тестов с разными устройствами. В названии архива или в самом архиве в файле `readme.txt` обозначьте, на каких типах устройств проходили тесты.
13. Прикрепите архивы к тикету. Если архивы получились слишком большие – загрузите их на любой облачный файлообменник и пришлите нам ссылку.