Содержание

Вывод статистики по flow

Команда: `fdpi_cli dump flow cache format`

Формат вывода

Пример:

```
nthr=1 slic=3 proto=6 ip_1=192.168.4.20:65163 ip_2=217.69.133.145:443
ssid=1675E5CF5FB1337 dpip=91 ittr=16 tmlb='2019/10/30 02:02:51, -357.642147s
(4148500652028035 ticks)' ialf=0 drct=0x1 iown=1 ilst=1 btsip=0x2
tcpbts_0='-APRSF' tcpbts_1='-AP-S-' qoest=0 qoef_0=0 qoef_1=0 qoer_0=6
qoer_1=6 whip=94.140.198.86:33326 itrnsld=1 igcache=0 gre_pid=0 gre_mtd=0
```

По полям:

- `nthr=1` — номер потока, куда помещена запись (для мультикластера может не совпадать с `iown`)
- `slic=3` — номер slice кэша
- `proto=6` — IP протокол
- `ip_1=192.168.4.20:65163 ip_2=217.69.133.145:443` — пара IP адресов и портов идентифицирующие запись. Если у протокола нет портов — последние `0`
- `ssid=1675E5CF5FB1337` — идентификатор сессии
- `dpip=91` — протокол DPI
- `ittr=16` — индекс в очереди использования повторных записей
- `tmlb='2019/10/30 02:02:51, -357.642147s (4148500652028035 ticks)'` — время последнего обращения к записи
- `ialf=0` — номер очереди обработки :
  - `en_nalfs_shrt = 0` — очередь с коротким временем жизни
  - `en_nalfs_long = 1` — долгоиграющая очередь
- `drct=0x1` — при каких условиях создана запись. Младшие 4 бита задают направление пакета, при котором создан ключ и соответственно принадлежность `src_ip` и `dst_ip`
  drct = h_ip_1 < h_ip_2 :
  - `drct == 0` — `h_ip_1` — `src_ip`
  - `drct == 1` — `h_ip_1` — `dst_ip`
    старшие 4 бита задают `flw_dir`, при котором был создан ключ
- `iown=1` — номер потока, который создал запись
- `ilst=1` — номер потока, который последний раз обрабатывал запись
- `btsip=0x2` — служебные биты обработки flow
- `tcpbts_0='-APRSF' tcpbts_1='-AP-S-'` — биты TCP соединения в двух направлениях:

  ```
  ( tcp_bits_ & 0x0020 ) ? 'U' : '-'
  ( tcp_bits_ & 0x0010 ) ? 'A' : '-'
  ( tcp_bits_ & 0x0008 ) ? 'P' : '-'
  ( tcp_bits_ & 0x0004 ) ? 'R' : '-'
  ( tcp_bits_ & 0x0002 ) ? 'S' : '-'
  ```

```
( tcp_bits_ & 0x0001 ) ? 'F' : '-'
```

- `qoest=0` — статус QoE:
  - `enst_none = 0`,
  - `enst_ack` — ждем подтверждающий ACK от клиента на SYN+ACK от сервера
  - `enst_fin_ack` — ждем подтверждающий FIN+ACK от сервера на FYN от клиента
  - `enst_ack_srvfin` — ждем подтвердающий ACK от сервера на FIN+ACK от клиента (сервер первый послал FIN)
- `qoef_0=0 qoef_1=0` — кол-во фрагментированных пакетов в двух направлениях
- `qoer_0=6 qoer_1=6` — кол-во ретрансмитов в двух направлениях
- `pktp_0=1 pktp_1=0` — количество пакетов с payload в двух направлениях, но не более 65000
- `btsp_0=1 btsp_1=0` — объем payload в двух направлениях, но не более 65K
- `whoisc=0` или `1` — кто инициировал соединение
- **Опционально** — если есть NAT трансляция:
  - `whip=94.140.198.86:33326` — выделенный белый адрес+порт
  - `itrnsld=1` — индекс данных профиля по которому был выделен белый адрес
  - `igcache=0` — индекс в соответствующем кэше-slice перекодировки серый -→ белый
  - `gre_pid=0` — опередеренный callid
  - `gre_mtd=0` — метод выделения белого адреса для GRE