

# Table of Contents

<b>Beta-version 14.2</b> .....	3
<b><i>Changes in version 14.2 BETA8</i></b> .....	3
<b><i>Changes in version 14.2 BETA7</i></b> .....	3
<b><i>Changes in version 14.2 BETA6</i></b> .....	4
<b><i>Changes in version 14.2 BETA5</i></b> .....	5
<b><i>Changes in version 14.2 BETA4</i></b> .....	6
<b><i>Changes in version 14.2 BETA3</i></b> .....	7
<b><i>Changes in version 14.2 BETA2.1</i></b> .....	7
<b><i>Changes in version 14.2 BETA2</i></b> .....	8
<b><i>Changes in version 14.2 BETA1</i></b> .....	8



# Beta-version 14.2

## Changes in version 14.2 BETA8

1. [DPI] Changed: after IPSNI check, fallback to base protocol or protocol defined by SNI (if detected)
2. [DPI] Changed: reduced inspection depth when attempting to decode cname/sni
3. [CLI] Added pcap capture command from port:

```
dev pcap <dev-name> rx|tx|any|off
```

- rx — record packets received from the port
- tx — record packets sent to the port
- any — rx and tx
- off — stop recording

pcap file prefixes (dev - port name):

- rx-dev — for rx
- tx-dev — for tx

4. [NAT] Fixed consistency of private address queue
5. [DPI] Fixed issue with tx-port selection for multi-valued configurations: return packet is now preferentially sent to the port from which the original packet arrived
6. [NAT] Fixes and optimization of private address port queue:
  1. Private address port queue is distributed across threads
  2. Private address port queue is split into "short" and "long"
7. [CLI][RG] Added: command `rg show <IP>` for viewing current rating group data for a subscriber

## Changes in version 14.2 BETA7

1. [DPI] Changed: FakeSNI check is not performed if protocol is determined by IP and there is no mark1
2. [CG-NAT] Optimized statistics command `fdpi_ctrl list all status --service 11`
3. [DPDK] Increased maximum memory size to 256 GB
4. [DPI][BRAS] Added service 20: rating group (RG) policing and volume quota control.

Creating service 20 profile:

1. Enable RG support in `fastdpi.conf`  
`rating_group_count=0` — number of rating groups, 0 — RG disabled. Default value: 0
2. Prepare a text file where each rating group defines TBF policing, quota, and action upon quota exhaustion, example:

```
rg4 tbf rate 1Mbit burst 1Mbit inbound.rate 8Mbit inbound.burst  
1Mbit quota 100MB report  
rg5 tbf rate 8Mbit burst 1Mbit inbound.rate 8Mbit inbound.burst  
1Mbit quota 1GB block
```

report and block are available actions upon quota reach: report — notify about quota

reached but continue traffic forwarding; `block` — notify and block traffic for this rating group

3. Convert text file to binary format:

```
cat rg.txt | lst2rg rg.bin
```

4. Place resulting binary file into directory from which DPI will read it:

```
cp rg.bin /var/lib/dpi/rg.bin
```

5. Create service profile:

```
fdpi_ctrl load profile --service 20 --profile.name rg1 --  
profile.json '{ "rg_list" : "/var/lib/dpi/rg.bin" }'
```

`max_profiles_serv20` — maximum number of profiles. Default — 32.

Utility `rg2lst` allows decoding binary file into readable form:

```
rg2lst rg.bin > rg.txt
```

5. [DPIUTILS] Added utilities `lst2rg` and `rg2lst` for converting service 20 profiles

## Changes in version 14.2 BETA6

1. [DPI] Added `viber_cl` check by container
2. [DPI] Fixed: override of cloud protocols by some built-in ones
3. [DPI] Fixed: added protocol detection for addresses when SNI is already in the first packet to preserve IP/SNI priority
4. [DPI] Fixed: DSCP detection from the first packet for cloud protocols defined by addresses
5. [NAT] Added explicit TCP connection close when port is reused by another subscriber
6. [CLI] Added new fields in `fdpi_cli dump flow cache` command. [Description](#)
7. [NAT] Changed public port queue handling: ports with short lifetime and long lifetime are now in separate queues. Ports are now elements of a private address subqueue. A port accessed from a non-owner flow thread can be reused immediately
8. [CLI] Added rating group and tethering control via service 18, where new optional fields were added to the profile configuration:

`tethN`, possible values:

- `teth0` — no tethering control (default)
- `teth1` — tethering control enabled: tethering present
- `teth2` — tethering control enabled: no tethering

`rgN`, possible values:

- `rg0` default (rg not set)
- `rg1` `rg=1` is set
- ..
- `rg65535` `rg=65535` is set

**Example of service 18 configuration:**

1. prepare configuration file example.txt

```
http cs0 teth1 rg1
https cs0 teth1 rg1
http cs0 teth2 rg2
https cs0 teth2 rg2

dns cs1 teth1 rg1
dns cs1 teth2 rg2

default cs7 teth0 rg3
```



**In this example, tethering is tracked for http/https protocols and corresponding RG is assigned depending on it. Note that policing class cs is the same. Similarly for dns protocol. For ALL other protocols (default), tethering control is disabled and a separate RG is specified.**

2. convert to internal format

```
cat example.txt|lst2dscp /tmp/example.bin
```

3. optionally verify with reverse conversion

```
dscp2lst /tmp/example.bin
```

4. create service 18 profile and assign to subscriber (or assign unnamed profile directly)

```
fdpi_ctrl load profile --service 18 --profile.name test_dscp
--profile.json '{ "dscp" : "/tmp/example.bin" }'
fdpi_ctrl load --service 18 --profile.name test_dscp --login
test_subs
```

check

```
fdpi_ctrl list --service 18 --login test_subs
```

In trace, field rg=N is added

9. [BRAS][DHCPv6] Fixed periodic ICMPv6 Router Adv sending for DHCPv6 subscribers
10. [BRAS][pppoe] Fixed modification of src/dst MAC in Ethernet header during termination. For PPPoE packets, Ethernet termination must always be performed. But with bras\_term\_by\_as=1 enabled, when srcAS is not marked as term, Ethernet src/dst MAC was not changed

## Changes in version 14.2 BETA5

1. [BASE] Added LLDP support.

When LLDP support is enabled, fastDPI emits LLDP packets (LLDPDU) on specified ports. Incoming LLDP packets are unconditionally dropped.

New fastdpi.conf parameters (all parameters are hot, set in lldp section):

- `enable` — LLDP support enable flag. By default, if `lldp` section exists in config, `enable=on`; otherwise `enable=off`
- `chassis` — string — Chassis-Id value. Chassis-Id TLV is mandatory in LLDP packet. If not set, it is taken as MAC address from `bras_arp_mac`, otherwise port MAC
- `tll` — number — TTL value in seconds, default 120
- `src_mac` — MAC address — source MAC in Ethernet header of LLDPDU. If not set, taken from `bras_arp_mac` or port MAC
- `dest_mac` — MAC address — destination MAC in Ethernet header of LLDPDU. Default — `01:80:c2:00:00:0e` (LLDPDU multicast)
- `system_name` — string — System-Name TLV value. If not set, TLV is not included in LLDPDU
- `system_desc` — string — System-Desc TLV value. If not set, TLV is not included in LLDPDU
- `device=<port_name>;enable=<on|off>;desc=<port_desc>` — ports for which LLDPDU should be sent. Each port is defined via separate device parameter; port name is from `in_dev/out_dev`. Per-port options:
  - `enable=on|off` — enable/disable LLDPDU sending for this port, default on
  - `desc=string` — Port-Desc TLV value; if not set, TLV is not included

Debug options:

- `trace` — enable LLDP tracing (boolean)
  - `pcap` — write LLDP packets to PCAP (boolean)
2. [CLI] New CLI commands: `lldp enable`, `lldp disable` — allow enabling/disabling LLDP packet generation
  3. [NAT] Improvements in session limit management: for `nat_tcp_max_sessions/nat_udp_max_sessions` limits, which define the number of allocated public ports, fixed decrement of allocated port counter which could lead to slight limit overflow. Counters `whpf`, `whp_salfs`, `whp_lalfs`, `whp_ruse`, `whp_ruse_salfs`, `whp_ruse_lalfs` and similar flow statistics counters (`thr_salfs` etc.), as well as `nat show` output, now reflect current actual usage instead of cumulative usage
  4. [NAT] Fixed: NAT translation validity check in FullCone mode when `nat_whp_lifetime < lifetime_flow`: if session becomes active again while NAT port is already reused, a new port is allocated

## Changes in version 14.2 BETA4

1. [DPI] Added detection of FakeTLS protocol with validation
2. [DPI] Fixed: switching from `QUIC_UNKNOWN` to `QUIC` upon successful SNI parsing
3. [DHCP6-Proxy] Added DHCPv6 option 79 Client-LinkLayer-Address, containing subscriber MAC address, in Relay-Forward requests to DHCPv6 Framed-Pool server
4. [VLAN-Rule][PPPoE] Added to `vlan rule show` command output of all permissions for Service-Name
5. [VLAN-Rule][PPPoE] Added full support for Service-Name in QinQ. Supported rules:
  1. without CVLAN selectivity: rules of type `SVLAN.*` with and without SName
  2. full QinQ (`svlan.cvlan`) with SName selectivity
6. [VLAN-Rule][PPPoE] Refactoring of Service-Name support. `vlan rule add/rm` commands now support PPPoE and Service-Name.  
Adding PPPoE processing rule for a given `<Range>` VLAN/QinQ:

```
vlan rule add <Range> pppoe [enable | drop | pass | delay N]
```

Adding PPPoE Service-Name processing rule for a given <Range> VLAN/QinQ:

```
vlan rule add <Range> pppoe sname <Service-Name> [enable | drop | pass  
| delay N]
```

Here <Service-Name> is the PPPoE Service-Name in single or double quotes (or without quotes if it is an identifier ([a-zA-Z\_][a-zA-Z\_0-9]\*))

Permissions:

1. enable - PPPoE processing allowed
2. drop - drop PPPoE packets
3. pass - pass PPPoE packets through without processing
4. delay N - establish PPPoE session with a delay of N seconds (0 < N < 16)
7. [IPFIX] Added ability to send data over UDP exceeding MTU size (with IP fragmentation)
8. [DNS] Added parameters `ajb_save_dns_answer_types` and `ajb_save_dns_request_types` allowing definition of DNS request/response types for file logging and IPFIX export
9. [IPFIX] Fixed default timeout configuration error
10. [DHCP-Dual] Fixed incorrect IPv6 PD prefix formation for addresses from Framed-IPv6-Pool
11. [DHCP-Dual] Fixed crash when enabling tracing by MAC `bras_dhcp_trace_mac`
12. [DHCP-Dual] Fixed issue where DHCPv6 followed by DHCPv4 request sequence caused extra authorization
13. [DHCP-Dual] Fixed DHCPv6 response tracing when MAC address tracing is enabled
14. [DNS] Added utility `dic2dns`. [Description](#)

## Changes in version 14.2 BETA3

### DPI

1. [DPI] Added GRE ERSPAN tunnel parsing for `check_tunnels=1` mode
2. [DPI] Message "Can't allocate record `http_state`" is now printed once per 50000 occurrences
3. [DPI] Added MARK2 flag check for redefinition into `QUIC_UNKNOWN_MARKED` when QUIC protocol is still being identified via SNI. [Description](#)

### BRAS

1. [BRAS][Router] Changed Linux route table parsing at router startup. [Description](#)

## Changes in version 14.2 BETA2.1

### NAT

1. [CG-NAT] NAT optimization changes

## Changes in version 14.2 BETA2

### DPI

1. [DPI][DNS] Fixed issue with service 19 for IPv6 traffic

### BRAS

1. [BRAS][DHCP-Dual] Added Lease-Time accounting  
In DHCP Dual mode, it is critical that `Session-Timeout` is at least 4 times greater than `Lease-Time`. If this condition is violated, `Lease-Time` is set to 1/4 of `Session-Timeout`. `Lease-Time` is taken from RADIUS authentication response (in order of priority):
  1. attribute `DHCP-IP-Address-Lease-Time`;
  2. attribute `VasExperts-DHCP-Option-Num` defining option 51;
  3. DHCP option 51 if address is allocated from `Framed-Pool`.

If `Lease-Time` is not specified by any of the above methods, it is set to 1/16 of `Session-Timeout`.

Minimum values:

- `Session-Timeout` — 600 seconds
- `Lease-Time` — 60 seconds

### NAT

1. [CG-NAT] Added support for disabling white address cache for NAT export. Setting `nat_dstaddr_cache_size=0` in `/etc/dpi/fastdpi.conf`

## Changes in version 14.2 BETA1

### DPI

1. [DPDK] Migration to new DPDK version 25.11. [Description](#)
2. [DPI][NAT] Optimization under private-to-public cache overflow
3. [CLI][VLAN] Added parameter to `vlan rule dump` command defining rule type output: `vlan rule dump [type]`  
`type` — rule type: `perm`, `dhcp`, `all` (default)  
Show VLAN permissions:

```
vlan rule dump perm
```

Show DHCP-only rules:

```
vlan rule dump dhcp
```

Show all rules:

```
vlan rule dump
```

4. [CLI][DPI] Extended output of `fdpi_cli dump flow cache format` with new fields.

[Description](#)

5. [BALANCER] Added ability to use vlan rule for packet filtering
6. [DPDK] Added new option `dpdk_max_memzone [cold]` — sets DPDK max memzone count. Default in DPDK is 5120 (depends on DPDK version)  
0 — use default value from DPDK. Increasing this is useful for huge configurations with many NICs if startup error occurs: "Number of requested memzone segments exceeds maximum 5120"
7. [CLI][DHCP-Dual] Added support for command `dhcp show stat vrf`
8. [DPDK] New engine `dpdk_engine=7` with explicit dispatcher assignment  
This engine supports heterogeneous configurations where ports of different types exist in one cluster — e.g. in-dev 100G port and multiple 10G out-dev ports.  
Dispatchers are defined in `dpdk_dispatch`:

```
dpdk_dispatch=<port-list>[;params]*
```

- `<port-list>` defines which ports are handled by this dispatcher
- `params` — additional options:
  - `rss=N` — enable RSS on all ports in dispatcher; creates N dispatchers per RX queue
  - `mempool_size=N` — size of `mbuf_pool` for dispatcher; each dispatcher has its own mempool

Multiple `dpdk_dispatch` entries may exist; each defines a separate dispatcher (or group if RSS is enabled). Each cluster port must belong to exactly one `dpdk_dispatch`. On-stick ports must reference the base physical port.

Configuration errors:

- cluster port is not included in any `dpdk_dispatch`
- cluster port appears in multiple `dpdk_dispatch` entries
- ports from different clusters are mixed in one dispatcher

Example mappings:

```
dpdk_engine=0: single dispatcher for all ports
dpdk_engine=1: dispatcher per direction
dpdk_engine=3: bridge dispatcher
dpdk_engine=4: per-port dispatcher
dpdk_engine=6: bridge dispatcher with RSS
```

9. [IPFIX] Fixed error when changing `ipfix_dev` option