Содержание

Load balancer

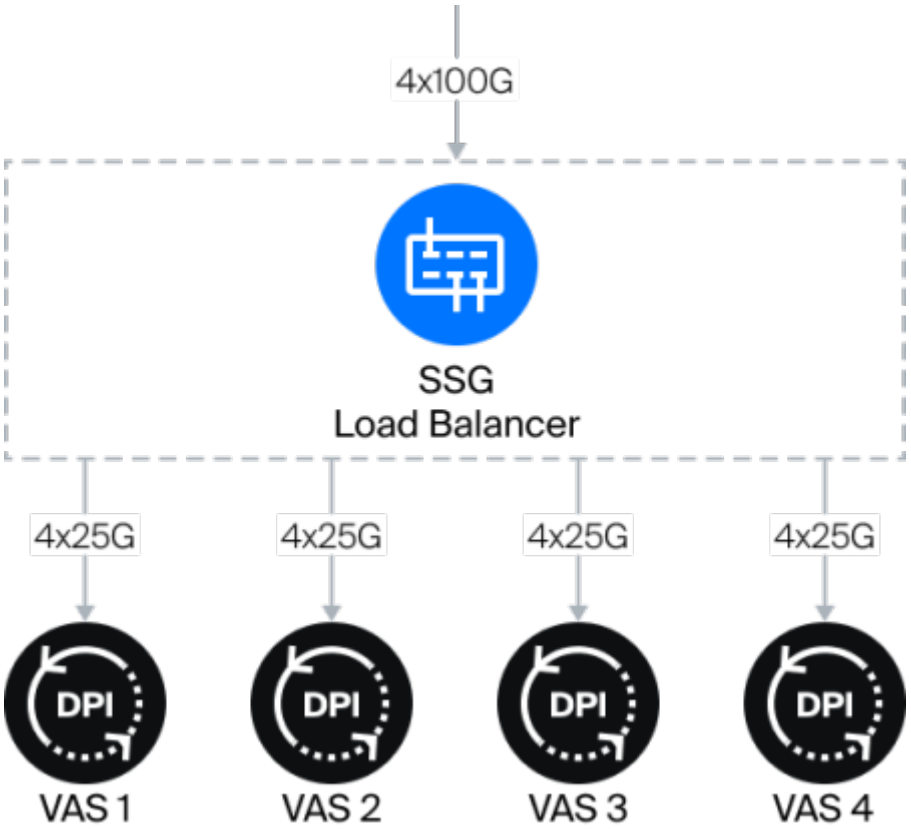
Description and cases

The SSG can act as a traffic balancer based on IP addresses belonging to an AS defined as `local` in `asnum.dscp`.
In this case, the SSG-LB acts as an L2-bridge in the network, hence the name L2 traffic balancer.

DPI functionality does not work in this mode.

Case: Balancer on a traffic mirror

Suppose a 400Gbps traffic mirror needs to be evenly distributed among four VAS platforms (Value-Added Services) performing traffic analysis and various detection.
In this case, SSG-LB will evenly distribute traffic with equal port utilization on the pullers and maintain traffic symmetry (traffic from one session will be directed to only one puller).



Example configuration /etc/dpi/fastdpi.conf:

- Traffic mirroring is fed into 4x100G interfaces.
- Traffic is balanced between four pullers, each puller is connected by 4x25G links.

```
in_dev=05-00.0:05-00.1:05-00.2:05-00.3
out_dev=01-00.0:01-00.1:01-00.2:01-00.3:02-00.0:02-00.1:02-00.2:02-00.3:03-
```

```
0.0:03-00.1:03-00.2:03-00.3:04-00.0:04-00.1:04-00.2:04-00.3

#FastDPI Control
ctrl_port=29000
ctrl_dev=lo

#Turn on Load Balancing
enable_l2_lb=1

#Balance algorithm
maglev=2

#Hash table
lb_hash_out_dev_type=1
```

Setting

Setting up subnets for balancing

Balancing applies only to IP addresses belonging to the AS defined as `local` in `asnum.dscp`.

1. Define an autonomous system with IP addresses that are used by subscribers:

   ```
   vi aslocal.txt
   10.0.0.0/8 64511
   172.16.0.0/12 64511
   192.168.0.0/16 64511
   cat aslocal.txt | as2bin /etc/dpi/aslocal.bin
   ```

   1. Examples for IPv4
   2. Example for IPv6
2. Mark a given autonomous system as local:

   ```
   vi my_as_dscp.txt
   64511 local
   10415 local
   cat my_as_dscp.txt | as2dscp /etc/dpi/asnum.dscp
   ```

   You can use either a custom AS or a public AS as the AS.
   Read more at the link

3. A reload must be performed to apply the parameters:

   ```
   service fastdpi reload
   ```

Setting /etc/dpi/fastdpi.conf

1. Define the input and output interfaces in the configuration file.

The input interfaces to which the traffic mirror is fed are specified in the `in_dev` parameter, and the output interfaces are specified in `out_dev`.
Interfaces do not form pairs and species constructs are allowed:

```
in_dev=05-00.0:05-00.1:05-00.2:05-00.3:0b-00.0:0b-00.1:0b-00.2:0b-00.3
out_dev=08-00.0:08-00.1:08-00.2:08-00.3
```

Or

```
in_dev=05-00.0
out_dev=out_dev=08-00.0:08-00.1:08-00.2:08-00.3
```

2. Enable balancing mode in the `enable_l2_lb` parameter, where:
   - `0` — deactivate balancing;
   - `1` — activate balancing.
3. Determine by what value to initialize the hash table in the `lb_hash_out_dev_type` parameter, where:
   - `0` — use the internal index of the output interface;
   - `1` — use the interface name from [in|out]_dev.
4. Select an engine to handle thread dispatchers in the `dpdk_engine` parameter, where:
   - `0` — read/write **default** engine, one dispatcher for everything;
   - `1` — read/write engine with two dispatcher threads: a dispatcher for each direction;
   - 2 — Read/write engine with RSS support: for each direction, `dpdk_rss` dispatchers are created (default `dpdk_rss`=2), so total number of dispatchers = 2 * `dpdk_rss`— to work with in_dev, and a separate dispatcher to work with out_dev.
     When the parameter value is 2, the `mqrx_lb_engine` engine is activated. The principle of operation is the same as in the usual mode `dpdk_engine`=2, only rss is enabled on in_dev, and only one rx queue is created on out_dev.

     Learn more about the `dpdk_engine` parameter by clicking here.
5. Select a balancing algorithm. The `maglev` algorithm with fixed hash table size is used for traffic balancing, where:
   - `1` — if src and dst ip are both local, then hash is calculated based on these two addresses;
   - `2` — if only src ip local, then hash is calculated based on src ip;
   - `3` — if only dst ip local, hash is calculated on the basis of dst ip;
   - `4` — hash is calculated based on src and dst ip.

     Based on the calculated hash value, the output interface is determined by determining the index of the hash table cell containing the interface index from the array of output interfaces.

SSG-LB equipment requirements

Max IN traffic Gbps (Mirror on SSG-LB)	**Max OUT traffic Gbps (Balanced traffic to the removers)**	**SSG-LB version**	**Number of cores of 2.5 GHz or more**	**RAM GB**	**Type and minimum number of ports**	**Packet per second in millions at a base CPU frequency of 2.5GHz or more**
up to 100	up to 100	**SSG-100-LB**	28 Intel 6258R, Intel 5320, 32 AMD 7502P	64 (8x8GB)	2x100G + 14x10/25G	40M pps
up to 200	up to 200	**SSG-200-LB**	64 AMD 9534	64 (8x8GB)	9x25/40/50/100G	60M pps
up to 300	up to 300	**SSG-300-LB**	96 AMD 9654	64 (8x8GB)	9x25/40/50/100G	80M pps
up to 400	up to 400	**SSG-400-LB**	128 AMD 9754	64 (8x8GB)	9x25/40/50/100G	120M pps
up to 800	up to 800	**SSG-800-LB**	2x128 AMD 9754	128 (16x8GB)	9x25/40/50/100G	240M pps