Table of Contents

SSG Platform Device

1. Is it normal for the fastdpi_1gb process to show a 140% load on an idle server (4-core CPU) and 160-220% under a 50 Mbps load? Does anything need to be adjusted?

The high idle load is due to the use of polling instead of interrupts for card operations to ensure low latency. As data flow increases, this load shifts more towards useful work.
We recommend monitoring CPU load with the `mpstat -P ALL` utility.

2. How can I check the license expiration date?

```
grep 'expiration_date=' /etc/dpi/fastdpi.lic
expiration_date=20991231
```

Format: YYYYMMDD

3. Which files need to be archived to save license information?

```
/etc/dpi/fastdpi.lic
/etc/dpi/fastdpi.sig
/etc/pf_ring/*
```

4. How can I view the list of changes before installation?

First option: Go to the required version from the main Wiki page, where announcements of new versions are published. Each version has a list of changes.

Second option: Install the `yum-plugin-changelog` package and use the `changelog` command:

```
yum install yum-plugin-changelog
yum changelog 4 fastdpi
```

5. Does the software work on FreeBSD OS?

There is no version for FreeBSD. Only **VEOS** is supported.
We strongly recommend using the OS image specified in this article.

6. Can third-party monitoring solutions be used?

For monitoring, you can use solutions that utilize SNMP. For example, Zabbix Agent. Description

7. How is the load distributed across CPU cores (12 cores/2 clusters)?

Cores are functionally distributed among various DPI tasks so they do not interfere with each other. You can view the function distribution with the command:

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

8. Why are there unused CPU cores (12th core is idle)?

DPI has service functions such as Netflow generation, Clickstream, PCAP recording, command processing, etc.

Their load is uneven and they can temporarily load a core to 100%, so a separate core is allocated to them to avoid interfering with traffic.

9. Why can one core be 100% loaded under typical DPI load? The server becomes inaccessible, and only a reboot helps.

It may be due to the `kipmi` process of the remote server control interface `ipmi`, or perhaps the FW was not closed from external attacks.
When a process loads any core to 100%, softRAID stops working, making it impossible to access the server. There is also an article on other possible issues with `ipmi`: Kipmi0 eating up to 99.8% CPU on CentOS 6.4

10. Are the utilities url2dic and ip2bin available in source code or for FreeBSD 9 x64?

The source code of the utilities is not available and is not planned to be provided. FreeBSD allows running native Linux applications. Also, for FreeBSD 9.2, a binary version of the utilities is available.