

How to enable logging in Open vSwitch for debugging and troubleshooting

Author : Dan Nanni

Categories : [Development](#), [Networking](#), [Virtualization](#)

Tagged as : [log file](#), [openvswitchsyslog](#)

Question: I am trying to troubleshoot my Open vSwitch deployment. For that I would like to inspect its debug messages generated by its built-in logging mechanism. How can I enable logging in Open vSwitch, and change its logging level (e.g., to INFO/DEBUG level) to check more detailed debug information?

Open vSwitch (OVS) is the most popular open-source implementation of virtual switch on the Linux platform. As the today's data centers increasingly rely on the software-defined network (SDN) architecture, OVS is fastly adopted as the de-facto standard network element in data center's SDN deployments.

Open vSwitch has a built-in logging mechanism called VLOG. The VLOG facility allows one to enable and customize logging within various components of the switch. The logging information generated by VLOG can be sent to a combination of console, syslog and a separate log file for inspection. You can configure OVS logging dynamically at run-time with a command-line tool called `ovs-appctl`.

```
2015-07-21T20:02:19Z|00001|vlog|INFO|opened log file /openvswitch-2.3.1/var/log/openvswitch/ovs-vswitchd.log
2015-07-21T20:02:19Z|00002|reconnect|INFO|unix: /openvswitch-2.3.1/var/run/openvswitch/db.sock: connecting...
2015-07-21T20:02:19Z|00003|reconnect|INFO|unix: /openvswitch-2.3.1/var/run/openvswitch/db.sock: connected
opened datapath ovs-netdev of type netdev: 0
2015-07-21T20:02:19Z|00004|ofproto_dpif|INFO|netdev@ovs-netdev: Datapath supports recirculation
2015-07-21T20:02:19Z|00005|ofproto_dpif|INFO|netdev@ovs-netdev: MPLS label stack length probed as 3
2015-07-21T20:02:19Z|00006|bridge|INFO|bridge ovsbr0: added interface ovsbr0 on port 65534
2015-07-21T20:02:19Z|00007|netdev_linux|WARN|ovsbr0: obtaining netdev stats via vport failed (No such device)
2015-07-21T20:02:19Z|00008|bridge|INFO|bridge ovsbr0: using datapath ID 00009e6351003344
2015-07-21T20:02:19Z|00009|connmgr|INFO|ovsbr0: added service controller "unix: /openvswitch-2.3.1/var/run/openvswitch/ovsbr0.mgmt"
2015-07-21T20:02:19Z|00010|bridge|INFO|ovs-vswitchd (Open vSwitch) 2.3.1
2015-07-21T20:02:24Z|00011|netdev_linux|WARN|ovsbr0: obtaining netdev stats via vport failed (No such device)
2015-07-21T20:02:29Z|00012|memory|INFO|7092 kB peak resident set size after 10.0 seconds
2015-07-21T20:02:29Z|00013|memory|INFO|handlers:17 ports:1 revalidators:7 rules:6
```

Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

Here is how to enable logging and customize logging levels in Open vSwitch with `ovs-appctl`.

The syntax of `ovs-appctl` to customize VLOG is as follows.

```
$ sudo ovs-appctl vlog/set module[:facility[:level]]
```

- **Module:** name of any valid component in OVS (e.g., `netdev`, `ofproto`, `dpif`, `vswitchd`, and many others)
- **Facility:** destination of logging information (must be: `console`, `syslog` or `file`)
- **Level:** verbosity of logging (must be: `emerg`, `err`, `warn`, `info`, or `dbg`)

In OVS source code, module name is defined in each source file in the form of:

```
VLOG_DEFINE_THIS_MODULE( );
```

For example, in `lib/netdev.c`, you will see:

```
VLOG_DEFINE_THIS_MODULE(netdev);
```

which indicates that `lib/netdev.c` is part of `netdev` module. Any logging messages generated in `lib/netdev.c` will belong to `netdev` module.

In OVS source code, there are multiple severity levels used to define several different kinds of logging messages: `VLOG_INFO()` for informational, `VLOG_WARN()` for warning, `VLOG_ERR()` for error, `VLOG_DBG()` for debugging, `VLOG_EMERG` for emergency. Logging level and facility determine which logging messages are sent where.

To see a full list of available modules, facilities, and their respective logging levels, run the following commands. This command must be invoked after you have started OVS.

```
$ sudo ovs-appctl vlog/list
```

Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

```
:~$ sudo ovs-appctl vlog/list
      console      syslog      file
-----
backtrace      INFO      INFO      INFO
bfd            INFO      INFO      INFO
bond           INFO      INFO      INFO
bridge         INFO      INFO      INFO
bundle        INFO      INFO      INFO
bundles       INFO      INFO      INFO
cfm           INFO      INFO      INFO
classifier     INFO      INFO      INFO
collectors    INFO      INFO      INFO
command_line  INFO      INFO      INFO
connmgr       INFO      INFO      INFO
coverage     INFO      INFO      INFO
daemon       INFO      INFO      INFO
daemon_unix  INFO      INFO      INFO
dpif         INFO      INFO      INFO
dpif_linux   INFO      INFO      INFO
dpif_netdev  INFO      INFO      INFO
```

The output shows the debug levels of each module for three different facilities (`console`, `syslog`, `file`). By default, all modules have their logging level set to `INFO`.

Given any one OVS module, you can selectively change the debug level of any particular facility. For example, if you want to see more detailed debug messages of `dpif` module at the console screen, run the following command.

```
$ sudo ovs-appctl vlog/set dpif:console:dbg
```

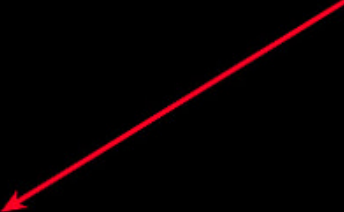
You will see that `dpif` module's `console` facility has changed its logging level to `DBG`. The logging level of two other facilities, `syslog` and `file`, remains unchanged.

Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

```
:~$ sudo ovs-appctl vlog/set dpif:console:dbg
:~$
:~$ sudo ovs-appctl vlog/list
      console      syslog      file
-----
backtrace      INFO      INFO      INFO
bfd            INFO      INFO      INFO
bond           INFO      INFO      INFO
bridge         INFO      INFO      INFO
bundle         INFO      INFO      INFO
bundles        INFO      INFO      INFO
cfm            INFO      INFO      INFO
classifier      INFO      INFO      INFO
collectors     INFO      INFO      INFO
command_line   INFO      INFO      INFO
connmgr        INFO      INFO      INFO
coverage       INFO      INFO      INFO
daemon         INFO      INFO      INFO
daemon unix    INFO      INFO      INFO
dpif           DBG       INFO      INFO
dpif_linux     INFO      INFO      INFO
dpif_netdev    INFO      INFO      INFO
```



If you want to change the logging level for all modules, you can specify "ANY" as the module name. For example, the following command will change the console logging level of every module to DBG.

```
$ sudo ovs-appctl vlog/set ANY:console:dbg
```

```
:~$ sudo ovs-appctl vlog/set ANY:console:dbg
:~$
:~$ sudo ovs-appctl vlog/list
      console      syslog      file
-----
backtrace      DBG       INFO      INFO
bfd            DBG       INFO      INFO
bond           DBG       INFO      INFO
bridge         DBG       INFO      INFO
bundle         DBG       INFO      INFO
bundles        DBG       INFO      INFO
cfm            DBG       INFO      INFO
classifier      DBG       INFO      INFO
collectors     DBG       INFO      INFO
command_line   DBG       INFO      INFO
connmgr        DBG       INFO      INFO
```

Also, if you want to change the logging level of all three facilities at once, you can specify "ANY" as the facility name. For example, the following command will change the logging level of all facilities for every

Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

module to DBG.

```
$ sudo ovs-appctl vlog/set ANY:ANY:dbg
```