

NAME

ovs-controller – simple OpenFlow controller reference implementation

SYNOPSIS

ovs-controller [*options*] *method* [*method*]...

DESCRIPTION

ovs-controller manages any number of remote switches over OpenFlow protocol, causing them to function as L2 MAC-learning switches or hub.

ovs-controller controls one or more OpenFlow switches, specified as one or more of the following OpenFlow connection methods:

pssl:*[port][:ip]*

Listens for OpenFlow SSL connections on *port* (default: 6633). The **—private-key**, **—certificate**, and **—ca-cert** options are mandatory when this form is used. By default, connections are not bound to a particular local IP address, but *ip* may be specified to listen only for connections to the given *ip*.

ptcp:*[port][:ip]*

Listens for OpenFlow TCP connections on *port* (default: 6633). By default, connections are not bound to a particular local IP address, but *ip* may be specified to listen only for connections to the given *ip*.

punix:*file*

Listens for OpenFlow connections on the Unix domain server socket named *file*.

ssl:*ip[:port]*

The specified SSL *port* (default: 6633) on the host at the given *ip*, which must be expressed as an IP address (not a DNS name). The **—private-key**, **—certificate**, and **—ca-cert** options are mandatory when this form is used.

tcp:*ip[:port]*

The specified TCP *port* (default: 6633) on the host at the given *ip*, which must be expressed as an IP address (not a DNS name).

unix:*file*

The Unix domain server socket named *file*.

OPTIONS

-n

—noflow

By default, **ovs-controller** sets up a flow in each OpenFlow switch whenever it receives a packet whose destination is known due through MAC learning. This option disables flow setup, so that every packet in the network passes through the controller.

This option is most useful for debugging. It reduces switching performance, so it should not be used in production.

—max-idle=secs|permanent

Sets *secs* as the number of seconds that a flow set up by the controller will remain in the switch's flow table without any matching packets being seen. If **permanent** is specified, which is not recommended, flows will never expire. The default is 60 seconds.

This option has no effect when **-n** (or **—noflow**) is in use (because the controller does not set up flows in that case).

-H

—hub By default, the controller acts as an L2 MAC-learning switch. This option changes its behavior to that of a hub that floods packets on all but the incoming port.

If **-H** (or **—hub**) and **-n** (or **—noflow**) are used together, then the cumulative effect is that every packet passes through the controller and every packet is flooded.

This option is most useful for debugging. It reduces switching performance, so it should not be used in production.

-w*[wildcard_mask]*

--wildcards*[=wildcard_mask]*

By default, **ovs-controller** sets up exact-match flows. This option allows it to set up wildcarded flows, which may reduce flow setup latency by causing less traffic to be sent up to the controller.

The optional *wildcard_mask* is an OpenFlow wildcard bitmask in hexadecimal that specifies the fields to wildcard. If no *wildcard_mask* is specified, the default value 0x2820F0 is used which specifies L2-only switching and wildcards L3 and L4 fields. Another interesting value is 0x2000EC, which specifies L3-only switching and wildcards L2 and L4 fields.

This option has no effect when **-n** (or **--noflow**) is in use (because the controller does not set up flows in that case).

-N

--normal

By default, **ovs-controller** directs packets to a particular port or floods them. This option causes it to direct non-flooded packets to the OpenFlow **OFPP_NORMAL** port. This allows the switch itself to make decisions about packet destinations. Support for **OFPP_NORMAL** is optional in OpenFlow, so this option may not well with some non-Open vSwitch switches.

--mute

Prevents ovs-controller from replying to any OpenFlow messages sent to it by switches.

This option is only for debugging the Open vSwitch implementation of “fail open” mode. It must not be used in production.

-q *id*

--queue*=id*

By default, **ovs-controller** uses the default OpenFlow queue for sending packets and setting up flows. Use one of these options, supplying *id* as an OpenFlow queue ID as a decimal number, to instead use that specific queue.

This option is incompatible with **-N** or **--normal** and with **-H** or **--hub**. If more than one is specified then this option takes precedence.

This option may be useful for testing or debugging quality of service setups.

-Q *port-name:queue-id*

--port-queue *port-name:queue-id*

Configures packets received on the port named *port-name* (e.g. **eth0**) to be output on OpenFlow queue ID *queue-id* (specified as a decimal number). For the specified port, this option overrides the default specified on **-q** or **--queue**.

This option may be specified any number of times with different *port-name* arguments.

This option is incompatible with **-N** or **--normal** and with **-H** or **--hub**. If more than one is specified then this option takes precedence.

This option may be useful for testing or debugging quality of service setups.

--with-flows *file*

When a switch connects, push the flow entries as described in *file*. Each line in *file* is a flow entry in the format described for the **add-flows** command in the **Flow Syntax** section of the **ovs-ofctl(8)** man page.

Use this option more than once to add flows from multiple files.

Public Key Infrastructure Options

-p *privkey.pem*

- private-key=privkey.pem**
Specifies a PEM file containing the private key used as **ovs-controller**'s identity for outgoing SSL connections.
- c cert.pem**
--certificate=cert.pem
Specifies a PEM file containing a certificate that certifies the private key specified on **-p** or **--private-key** to be trustworthy. The certificate must be signed by the certificate authority (CA) that the peer in SSL connections will use to verify it.
- C cacert.pem**
--ca-cert=cacert.pem
Specifies a PEM file containing the CA certificate that **ovs-controller** should use to verify certificates presented to it by SSL peers. (This may be the same certificate that SSL peers use to verify the certificate specified on **-c** or **--certificate**, or it may be a different one, depending on the PKI design in use.)
- C none**
--ca-cert=none
Disables verification of certificates presented by SSL peers. This introduces a security risk, because it means that certificates cannot be verified to be those of known trusted hosts.
- peer-ca-cert=peer-cacert.pem**
Specifies a PEM file that contains one or more additional certificates to send to SSL peers. *peer-cacert.pem* should be the CA certificate used to sign **ovs-controller**'s own certificate, that is, the certificate specified on **-c** or **--certificate**. If **ovs-controller**'s certificate is self-signed, then **--certificate** and **--peer-ca-cert** should specify the same file.
- This option is not useful in normal operation, because the SSL peer must already have the CA certificate for the peer to have any confidence in **ovs-controller**'s identity. However, this offers a way for a new installation to bootstrap the CA certificate on its first SSL connection.
- pidfile[=pidfile]**
Causes a file (by default, **ovs-controller.pid**) to be created indicating the PID of the running process. If the *pidfile* argument is not specified, or if it does not begin with */*, then it is created in **/usr/local/var/run**.
- If **--pidfile** is not specified, no pidfile is created.
- overwrite-pidfile**
By default, when **--pidfile** is specified and the specified pidfile already exists and is locked by a running process, **ovs-controller** refuses to start. Specify **--overwrite-pidfile** to cause it to instead overwrite the pidfile.
- When **--pidfile** is not specified, this option has no effect.
- detach**
Causes **ovs-controller** to detach itself from the foreground session and run as a background process.
- monitor**
Creates an additional process to monitor the **ovs-controller** daemon. If the daemon dies due to a signal that indicates a programming error (e.g. **SIGSEGV**, **SIGABRT**), then the monitor process starts a new copy of it. If the daemon die or exits for another reason, the monitor process exits.
- This option is normally used with **--detach**, but it also functions without it.
- no-chdir**
By default, when **--detach** is specified, **ovs-controller** changes its current working directory to the root directory after it detaches. Otherwise, invoking **ovs-controller** from a carelessly chosen directory would prevent the administrator from unmounting the file system that holds that directory.

Specifying **—no-chdir** suppresses this behavior, preventing **ovs-controller** from changing its current working directory. This may be useful for collecting core files, since it is common behavior to write core dumps into the current working directory and the root directory is not a good directory to use.

This option has no effect when **—detach** is not specified.

-v[spec]

—verbose=[spec]

Sets logging levels. Without any *spec*, sets the log level for every module and facility to **dbg**. Otherwise, *spec* is a list of words separated by spaces or commas or colons, up to one from each category below:

- A valid module name, as displayed by the **vlog/list** command on **ovs-appctl(8)**, limits the log level change to the specified module.
- **syslog**, **console**, or **file**, to limit the log level change to only to the system log, to the console, or to a file, respectively.
- **off**, **emer**, **err**, **warn**, **info**, or **dbg**, to control the log level. Messages of the given severity or higher will be logged, and messages of lower severity will be filtered out. **off** filters out all messages. See **ovs-appctl(8)** for a definition of each log level.

Case is not significant within *spec*.

Regardless of the log levels set for **file**, logging to a file will not take place unless **—log-file** is also specified (see below).

For compatibility with older versions of OVS, **any** is accepted as a word but has no effect.

-v

—verbose

Sets the maximum logging verbosity level, equivalent to **—verbose=dbg**.

—log-file[=file]

Enables logging to a file. If *file* is specified, then it is used as the exact name for the log file. The default log file name used if *file* is omitted is **/usr/local/var/log/openvswitch/ovs-controller.log**.

—unixctl=socket

Sets the name of the control socket on which **ovs-controller** listens for runtime management commands (see **RUNTIME MANAGEMENT COMMANDS**, below). If *socket* does not begin with **/**, it is interpreted as relative to **/usr/local/var/run**. If **—unixctl** is not used at all, the default socket is **/usr/local/var/run/ovs-controller.pid.ctl**, where *pid* is **ovs-controller**'s process ID. Specifying **none** for *socket* disables the control socket feature.

-h

—help Prints a brief help message to the console.

-V

—version

Prints version information to the console.

EXAMPLES

To bind locally to port 6633 (the default) and wait for incoming connections from OpenFlow switches:

```
% ovs-controller ptcp:
```

BUGS

Configuring a Citrix XenServer to connect to a particular controller only points the remote OVSDDB management connection to that controller. It does not also configure OpenFlow connections, because the manager is expected to do that over the management protocol. **ovs-controller** is not an Open vSwitch manager and does not know how to do that.

As a stopgap workaround, **ovs-vsctl** can wait for an OVSDDB connection and set the controller, e.g.:

```
% ovs-vsctl -t0 --db=pssl: --certificate=cert.pem --ca-cert=none --private-key=privkey.pem --peer-ca-cert=cacert.pem set-controller ssl:ip
```

SEE ALSO

ovs-appctl(8), ovs-ofctl(8), ovs-dpctl(8)