NAME
　　　　ovn-ic − Open Virtual Network interconnection controller

SYNOPSIS
　　　　ovn−ic [*options*]

DESCRIPTION
　　　　ovn−ic, OVN interconnection controller, is a centralized daemon which communicates with global inter-
　　　　connection databases IC_NB/IC_SB to configure and exchange data with local NB/SB for interconnecting
　　　　with other OVN deployments.

OPTIONS
　　　　−−ovnnb−db=*database*
　　　　　　　　The OVSDB database containing the OVN Northbound Database. If the **OVN_NB_DB** environ-
　　　　　　　　ment variable is set, its value is used as the default. Otherwise, the default is
　　　　　　　　unix:/ovnnb_db.sock.

　　　　−−ovnsb−db=*database*
　　　　　　　　The OVSDB database containing the OVN Southbound Database. If the **OVN_SB_DB** environ-
　　　　　　　　ment variable is set, its value is used as the default. Otherwise, the default is
　　　　　　　　unix:/ovnsb_db.sock.

　　　　−−ic−nb−db=*database*
　　　　　　　　The OVSDB database containing the OVN Interconnection Northbound Database. If the
　　　　　　　　OVN_IC_NB_DB environment variable is set, its value is used as the default. Otherwise, the de-
　　　　　　　　fault is **unix:/ovn_ic_nb_db.sock**.

　　　　−−ic−sb−db=*database*
　　　　　　　　The OVSDB database containing the OVN Interconnection Southbound Database. If the
　　　　　　　　OVN_IC_SB_DB environment variable is set, its value is used as the default. Otherwise, the de-
　　　　　　　　fault is **unix:/ovn_ic_sb_db.sock**.

　　　　database in the above options must be an OVSDB active or passive connection method, as described in
　　　　ovsdb(7).

　　Daemon Options
　　　　−−pidfile[**=***pidfile*]
　　　　　　　　Causes a file (by default, *program***.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 .

　　　　　　　　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, the daemon 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
　　　　　　　　Runs this program as a background process. The process forks, and in the child it starts a new ses-
　　　　　　　　sion, closes the standard file descriptors (which has the side effect of disabling logging to the con-
　　　　　　　　sole), and changes its current directory to the root (unless **−−no−chdir** is specified). After the
　　　　　　　　child completes its initialization, the parent exits.

　　　　−−monitor
　　　　　　　　Creates an additional process to monitor this program. If it dies due to a signal that indicates a pro-
　　　　　　　　gramming error (**SIGABRT**, **SIGALRM**, **SIGBUS**, **SIGFPE**, **SIGILL**, **SIGPIPE**, **SIGSEGV**,
　　　　　　　　SIGXCPU, or **SIGXFSZ**) then the monitor process starts a new copy of it. If the daemon dies 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, the daemon changes its current working directory to the root directory after it detaches. Otherwise, invoking the daemon 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 the daemon 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.

−−no−self−confinement

By default this daemon will try to self-confine itself to work with files under well-known directories determined at build time. It is better to stick with this default behavior and not to use this flag unless some other Access Control is used to confine daemon. Note that in contrast to other access control implementations that are typically enforced from kernel-space (e.g. DAC or MAC), self-confinement is imposed from the user-space daemon itself and hence should not be considered as a full confinement strategy, but instead should be viewed as an additional layer of security.

−−user=_user_**:**_group_

Causes this program to run as a different user specified in _user_**:**_group_, thus dropping most of the root privileges. Short forms _user_ and **:**_group_ are also allowed, with current user or group assumed, respectively. Only daemons started by the root user accepts this argument.

On Linux, daemons will be granted **CAP_IPC_LOCK** and **CAP_NET_BIND_SERVICES** before dropping root privileges. Daemons that interact with a datapath, such as **ovs−vswitchd**, will be granted three additional capabilities, namely **CAP_NET_ADMIN**, **CAP_NET_BROAD-CAST** and **CAP_NET_RAW**. The capability change will apply even if the new user is root.

On Windows, this option is not currently supported. For security reasons, specifying this option will cause the daemon process not to start.

Logging Options

−v[_spec_]
−−verbose=[_spec_]

Sets logging levels. Without any _spec_, sets the log level for every module and destination 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. (If **−−detach** is specified, the daemon closes its standard file descriptors, so logging to the console will have no effect.)

  On Windows platform, **syslog** is accepted as a word and is only useful along with the **−−syslog−target** option (the word has no effect otherwise).

- **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**.

−vPATTERN**:***destination***:***pattern*
−−verbose=PATTERN:*destination***:***pattern*
   Sets the log pattern for *destination* to *pattern*. Refer to **ovs−appctl**(8) for a description of the valid
   syntax for *pattern*.

−vFACILITY**:***facility*
−−verbose=FACILITY:*facility*
   Sets the RFC5424 facility of the log message. *facility* can be one of **kern**, **user**, **mail**, **daemon**,
   auth, **syslog**, **lpr**, **news**, **uucp**, **clock**, **ftp**, **ntp**, **audit**, **alert**, **clock2**, **local0**, **local1**, **local2**, **local3**,
   local4, **local5**, **local6** or **local7**. If this option is not specified, **daemon** is used as the default for the
   local system syslog and **local0** is used while sending a message to the target provided via the
   −−syslog−target option.

−−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/ovn/***program***.log**.

−−syslog−target=*host***:***port*
   Send syslog messages to UDP *port* on *host*, in addition to the system syslog. The *host* must be a
   numerical IP address, not a hostname.

−−syslog−method=*method*
   Specify *method* as how syslog messages should be sent to syslog daemon. The following forms are
   supported:

   •   **libc**, to use the libc **syslog()** function. Downside of using this options is that libc adds
       fixed prefix to every message before it is actually sent to the syslog daemon over **/dev/log**
       UNIX domain socket.

   •   **unix:***file*, to use a UNIX domain socket directly. It is possible to specify arbitrary mes-
       sage format with this option. However, **rsyslogd 8.9** and older versions use hard coded
       parser function anyway that limits UNIX domain socket use. If you want to use arbitrary
       message format with older **rsyslogd** versions, then use UDP socket to localhost IP ad-
       dress instead.

   •   **udp:***ip***:***port*, to use a UDP socket. With this method it is possible to use arbitrary message
       format also with older **rsyslogd**. When sending syslog messages over UDP socket extra
       precaution needs to be taken into account, for example, syslog daemon needs to be con-
       figured to listen on the specified UDP port, accidental iptables rules could be interfering
       with local syslog traffic and there are some security considerations that apply to UDP
       sockets, but do not apply to UNIX domain sockets.

   •   **null**, to discard all messages logged to syslog.

   The default is taken from the **OVS_SYSLOG_METHOD** environment variable; if it is unset, the
   default is **libc**.

PKI Options
   PKI configuration is required in order to use SSL/TLS for the connections to the Northbound and South-
   bound databases.

       −p *privkey.pem*
       −−private−key=*privkey.pem*
          Specifies a PEM file containing the private key used as identity for outgoing SSL/TLS
          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/TLS connections will use to verify it.

−C *cacert.pem*

−−ca−cert=*cacert.pem*

> Specifies a PEM file containing the CA certificate for verifying certificates presented to this program by SSL/TLS peers. (This may be the same certificate that SSL/TLS 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/TLS peers. This introduces a security risk, because it means that certificates cannot be verified to be those of known trusted hosts.

Other Options

−−unixctl=*socket*

> Sets the name of the control socket on which *program* listens for runtime management commands (see *RUNTIME MANAGEMENT COMMANDS,* below). If *socket* does not begin with **/**, it is interpreted as relative to . If **−−unixctl** is not used at all, the default socket is **/***program***.***pid***.ctl**, where *pid* is *program*'s process ID.
>
> On Windows a local named pipe is used to listen for runtime management commands. A file is created in the absolute path as pointed by *socket* or if **−−unixctl** is not used at all, a file is created as *program* in the configured *OVS_RUNDIR* directory. The file exists just to mimic the behavior of a Unix domain socket.
>
> 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.

RUNTIME MANAGEMENT COMMANDS

> **ovs−appctl** can send commands to a running **ovn−ic** process. The currently supported commands are described below.

> **exit**   Causes **ovn−ic** to gracefully terminate.
>
> **pause** Pauses the ovn-ic operation from processing any database changes. This will also instruct ovn-ic to drop any lock on SB DB.
>
> **resume** Resumes the ovn-ic operation to process database contents. This will also instruct ovn-northd to aspire for the lock on SB DB.
>
> **is−paused**
> > Returns "true" if ovn-ic is currently paused, "false" otherwise.
>
> **status** Prints this server's status. Status will be "active" if ovn-ic has acquired OVSDB lock on SB DB, "standby" if it has not or "paused" if this instance is paused.

ACTIVE-STANDBY FOR HIGH AVAILABILITY

> You may run **ovn−ic** more than once in an OVN deployment. When connected to a standalone or clustered DB setup, OVN will automatically ensure that only one of them is active at a time. If multiple instances of

ovn−ic are running and the active **ovn−ic** fails, one of the hot standby instances of **ovn−ic** will automatically take over.