NAME

virsh - management user interface for guest domains

SYNOPSIS

virsh subcommand [arguments]

DESCRIPTION

The main interface for command and control of both xVM and guest domains is virsh. Users should use virsh wherever possible, as it provides a generic and stable interface to controlling virtualized operating systems. Some xVM operations are not yet implemented by virsh. In those cases, the legacy utility xm(1M) can be used for detailed control.

With minor exceptions, the basic form of a virsh command is:

```
# virsh subcommand domain-id | name | uuid [options]
```

The components of a virsh command are described as follows:

subcommand

One of the subcommands described below.

domain-id | name | uuid

An identifier for a specific domain.

options

A subcommand-specific option.

Exceptions to command form described above occur when a subcommand acts on all domains, the entire machine, or directly on the Solaris xVM hypervisor.

Most virsh subcommands require root privileges or that you assume the Primary Administrator role.

Many virsh commands act asynchronously, so that the system prompt returns immediately while activity proceeds in the background. Many operations on domains, such as create and shutdown, can take considerable time (30 seconds or more) to

reach completion. Use the list subcommand to determine whether such an operation is complete.

DOMAIN SUBCOMMANDS

The following subcommands manipulate domains directly. Most take a domain identifier as their first argument. In the following description, the notation domain can be either a symbolic domain name, a numeric domain id, or a UUID, any of which uniquely identify a domain.

attach-device domain file

Attach a device defined by the given XML file (file) to the specified domain.

capabilities

Return capabilities of hypervisor/driver.

connect --readonly

Connect to local hypervisor. With the --readonly option, the connection is read-only.

console domain

Connect the virtual serial console for the guest.

create file

Create a domain based on the parameters contained in the XML file file, where file is an absolute pathname. Such a file can be created using virsh dumpxml subcommand. Directly editing XML configuration is not recommended.

define file

Define (but do not start) a domain from the specified XML file.

destroy domain

Immediately terminate a domain. This is the equivalent of abruptly terminating power to a machine. In most cases, you should use the shutdown subcommand instead.

detach-device domain file

Detach a device defined by the given XML file (file) from the specified domain.

domid domain_name

Converts a domain name to a numeric domain ID.

dominfo domain

Returns basic information about a domain.

domname domain_id

Converts a numeric domain id to a domain name.

domstate domain

Returns the state of a running domain. See the description of the list subcommand.

domuuid domain

Convert the specified domain name or ID to a domain UUID.

dump domain file

Dump the core of the domain specified by domain to the file specified by file for analysis.

dumpxml domain

Output the configuration of the given domain in XML format. Captured in a file, this data can be used as the argument to a subsequent create subcommand.

help subcommand

Displays a brief description of the specified subcommand.

help subcommand

Displays a detailed description of the specified subcommand.

list [domain...]

Displays information about one or more domains. If no domains are specified, displays information about all running domains. This subcommand takes the following options:

--inactive Display only non-running domains.

--all Display both running and non-running domains.

By default, only running domains are displayed.

An example of list output is as follows:

```
% virsh list
Id Name                 State

----------------------------------

0 Domain-0              running
2 fedora                paused
```

Id is the numeric id for a domain; Name is the symbolic name. State is the run state and can be one of the following:

running

The domain is currently running on a CPU.

blocked

The domain is not currently running on any CPU. This can be because the domain is waiting on I/O (a traditional wait state) or has gone to sleep because of inactivity.

paused

The domain has been paused, usually as a result of the administrator running virsh suspend. When in a paused state the domain still consumes allocated resources, such as memory, but is not eligible for scheduling by the xVM hypervisor.

in shutdown

The domain is in process of shutting down, but has not completely shutdown or crashed.

shutoff

The domain is down.

crashed

The domain has crashed as a result of a sudden event. Normally, this state can occur only if the

domain has been configured not to restart following a crash.

nodeinfo domain

Returns basic information about a node.

quit

Quit this interactive terminal.

reboot domain

Reboot a domain. The effect of this command is identical to the effect of running init 6. The command returns immediately, however the entire reboot process might take a minute or more.

restore state-file

Restores a domain from an virsh save state file. See the description of the save subcommand.

resume domain

Moves a domain out of the paused state, making the domain eligible for scheduling by the underlying hypervisor.

save domain state-file

Saves a running domain to a file state-file, so that it can later be restored, using the restore subcommand. Once saved, the domain will no longer be running on the system, thus the memory allocated for the domain will be free for the use of other domains.

Note that network connections present before the save operation might be severed, as TCP timeouts might have expired.

schedinfo domain

Show or set the scheduling paramaters for the specified domain name, ID or UUID. This subcommand takes the following options:

--weight number weight for XEN_CREDIT

--cap number cap for XEN_CREDIT

setmaxmem domain kilobytes

Change the maximum memory allocation limit in the specified guest domain. The kilobytes parameter is the maximum memory limit in kilobytes.

setmem domain kilobytes

Change the current memory allocation in the specified guest domain. The kilobytes parameter is the number of kilobytes of memory.

setvcpus domain count

Change the number of virtual CPUs active in the specified guest domain. The count parameter is the number of virtual CPUs.

shutdown domain

Coordinates with the domain OS to perform graceful shutdown. The effect of this command is identical to the effect of running init 5. There is no guarantee that the subcommand will succeed and it might take an unexpected length of time depending on what services must be shutdown in the domain.

start domain

Start a (previously defined) inactive domain.

suspend domain

Suspend a domain. When in this state, a domain still consumes allocated resources, such as memory, but is not eligible for scheduling by the xVM hypervisor.

undefine domain

Undefine the configuration for the inactive domain which is specified by either its domain name or UUID.

vcpuinfo domain

Return basic information about the domain virtual CPUs.

vcpupin domain vcpu cpulist

Pin domain VCPUs to the host physical CPUs. The domain parameter is the domain name, ID, or uuid. The vcpu

parameter is the VCPU number. The cpulist parameter is a list of host CPU numbers, separated by commas.

version

Display version information about this instance of virsh.

vncdisplay domain

Output the IP address and port number for the VNC display.

AUTHORS

- Andrew Puch, apuch at redhat dot com
- Daniel Veillard, veillard at redhat dot com
- Karel Zak, kzak at redhat dot com

The preceding authors credit the xm man page authored by:

- Sean Dague, sean at dague dot net
- Daniel Stekloff, dsteklof at us dot ibm dot com

ATTRIBUTES

See attributes(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	SUNWlibvirt
Interface Stability	Volatile

SEE ALSO

init(1M), xend(1M), xentop(1M), xm(1M), attributes(5), xVM(5)

NOTES

Terminology differs between xm(1M) and virsh. In particular, the suspend and resume commands have different meanings.

virsh	xm
suspend	pause
resume	unpause
save	suspend (without an output file argu-

ment)

restore	resume (without an output file argument)