Oracle® VM

Administrator's Guide for Release 3.4

E64083-19
January 2022

Oracle Legal Notices

Abstract

Document generated on: 2022-01-11 (revision: 7595)

Table of Contents

Preface

The Oracle VM Administrator's Guide explains how to manage Oracle VM and perform administrative tasks, such as configuring the system configuration, using Oracle VM Guest Additions, backing up and restoring components, troubleshooting common issues.

Audience

This document is intended for Oracle VM administrators with privileged access to the physical and virtual resources of the Oracle VM environment. This guide assumes that you have a detailed knowledge of Oracle VM and that you are familiar with Oracle Linux system administration and Linux command line operation.

Related Documents

For more information, see the following documents in the Oracle VM documentation set:

- *Oracle VM Release Notes*
- *Oracle VM Installation and Upgrade Guide*
- *Oracle VM Concepts Guide*
- *Oracle VM Manager Getting Started Guide*
- *Oracle VM Manager User's Guide*
- *Oracle VM Manager Command Line Interface User's Guide*
- *Oracle VM Administrator's Guide*
- *Oracle VM Paravirtual Drivers for Microsoft Windows Guide*
- *Oracle VM Web Services API Developer's Guide*
- *Oracle VM Security Guide*
- *Oracle VM Manager Third-Party Licensing Information*

You can also get the latest information on Oracle VM by going to the Oracle VM Web site:

http://www.oracle.com/us/technologies/virtualization/oraclevm

> **Note**
>
> Information on performance optimization goals and techniques for Oracle VM Server for x86 can be found in the Oracle Technical Paper entitled *Optimizing Oracle VM Server for x86 Performance*, which can be downloaded from the Oracle Technology Network: https://www.oracle.com/technetwork/server-storage/vm/ovm-performance-2995164.pdf

Command Syntax

Oracle Linux command syntax appears in `monospace` font. The dollar character ($), number sign (#), or percent character (%) are Oracle Linux command prompts. Do not enter them as part of the command. The following command syntax conventions are used in this guide:

<table>
<tr><th>Convention</th><th>Description</th></tr>
<tr><td>backslash \</td><td>A backslash is the Oracle Linux command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash:
<code>dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \</code>
<code>count=10000</code></td></tr>
<tr><td>braces { }</td><td>Braces indicate required items:
<code>.DEFINE {macro1}</code></td></tr>
<tr><td>brackets [ ]</td><td>Brackets indicate optional items:
<code>cvtcrt <i>termname</i> [<i>outfile</i>]</code></td></tr>
<tr><td>ellipses ...</td><td>Ellipses indicate an arbitrary number of similar items:
<code>CHKVAL fieldname value1 value2 ... valueN</code></td></tr>
<tr><td><i>italics</i></td><td>Italic type indicates a variable. Substitute a value for the variable:
<code><i>library_name</i></code></td></tr>
<tr><td>vertical line |</td><td>A vertical line indicates a choice within braces or brackets:
<code>FILE <i>filesize</i> [K|M]</code></td></tr>
<tr><td>forward slash /</td><td>A forward slash is used to escape special characters within single or double quotes in the Oracle VM Manager Command Line Interface, for example:
<code>create Tag name=MyTag description="HR/'s VMs"</code></td></tr>
</table>

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit
https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Chapter 1 Configuring Oracle VM Server After Installation

After you successfully install Oracle VM Server, you can perform configuration tasks to customize your environment. These configuration tasks include installing vendor-specific Oracle VM Storage Connect plug-ins, enabling multipathing, optionally installing and configuring diagnostics tools, and changing the memory size of the management domain. If you are using Oracle VM Server for SPARC, you can also create local ZFS volumes or configure a secondary service domain.

1.1 Installing Oracle VM Storage Connect plug-ins

Vendor-specific (non-generic) Oracle VM Storage Connect plug-ins are available directly from your storage vendor. Generic Oracle VM Storage Connect plug-ins are already installed by default during the installation of Oracle VM Server and no further action is required if you select to use only the generic plug-ins. Vendor-specific Oracle VM Storage Connect plug-ins usually facilitate additional management functionality that you can take advantage of from within Oracle VM Manager.

You can find more information about Oracle VM Storage Connect plug-ins at:

https://www.oracle.com/virtualization/storage-connect-partner-program.html

Oracle VM Storage Connect plug-ins are delivered as an RPM, usually a single RPM, but your storage vendor may provide multiple RPMs. When you have the Oracle VM Storage Connect plug-in RPM from your storage vendor, install the RPM on your Oracle VM Servers. You must install the RPM on all the Oracle VM Servers that will use the particular storage.

To install the Oracle VM Storage Connect plug-in RPM, on the command line of the Oracle VM Server, enter

```
# rpm -ivh filename.rpm
```

If you are upgrading an existing Oracle VM Storage Connect plug-in, use the RPM upgrade parameter:

```
# rpm -Uvh filename.rpm
```

If you are installing or upgrading an Oracle VM Storage Connect plug-in on an Oracle VM Server already managed by Oracle VM Manager, rediscover the Oracle VM Server to update the database repository with the latest configuration information about the Oracle VM Server.

Read the install and configuration documentation for the Oracle VM Storage Connect plug-in from your storage vendor before you install and use it. There may be extra configuration required that is not documented here.

1.2 Enabling Multipath I/O Support

In case user action is required to enable multipathing, this sections explains how to do so. The required steps depend on the storage hardware implemented. Consequently, the steps below are intended as a guideline and priority should be given to the SAN hardware documentation. Note that some guidelines have already been provided for the configuration of multipathing on SPARC hardware in the *Installing Oracle VM Server on SPARC Hardware* section of the *Oracle VM Installation and Upgrade Guide* . Not all steps apply to your environment. Consult the SAN hardware vendor documentation for a complete list of steps, the order in which to run them, and their relevance to your specific environment.

General steps to configure multipathing:

1. Design and document the multipathing configuration you intend to apply to the SAN hardware used in your Oracle VM environment.

2. Ensure that the drivers for your Host Bus Adapters (HBAs) are present. If not, install the drivers.
3. Configure the appropriate zoning on the fibre channel switches.
4. Configure LUN masking on the storage arrays.
5. Configure path optimization features (ALUA or similar) on your disk subsystem, if so instructed by your vendor's documentation.
6. Check the fabric information on each Oracle VM Server that has access to the SAN hardware. Use `multipath -ll` and related commands.
7. Make the necessary changes to the file `/etc/multipath.conf` on the Oracle VM Servers.

   > **Note**
   >
   > You must make the exact same changes to the multipath configuration file on all Oracle VM Servers in your environment.

   > **Important**
   >
   > It is critical that the configuration parameter `user_friendly_names` remain set to **no** within the `/etc/multipath.conf` configuration file.

   > **Important**
   >
   > Under the `multipath` section, the `multipaths` configuration subsection is not supported within the `/etc/multipath.conf` configuration file.

8. Restart the multipath daemon, `multipathd`.
9. Check the fabric information again to verify the configuration.
10. If instructed by the vendor documentation, rebuild `initrd`.
11. Reboot the Oracle VM Servers to verify that the SAN and multipathing configuration come up after a restart.

For detailed information and instructions, consult the SAN hardware vendor documentation.

> **Note**
>
> Booting from a multipath SAN is supported.

1.3 Configuring Software RAID for Storage

You can use software RAID devices for storage repositories or virtual disks. However you must first configure these devices on Oracle VM Server before Oracle VM Manager can discover the array for storage.

> **Important**
>
> As a best practice, you should use software RAID devices as storage repositories in a deployment environment before using them in a production environment.
>
> In environments where you use software RAID devices as storage repositories for server pools, unexpected behavior can occur with certain virtual machine migration operations. For example, if you clone a virtual machine and then attempt to live

migrate it to an instance of Oracle VM Server in the same server pool, the migration fails with an error that indicates the virtual machine disk does not exist. In this case, you must stop the virtual machine and then move it to the appropriate instance of Oracle VM Server.

To configure software RAID devices as storage, do the following:

1. Connect to Oracle VM Server as the root user.
2. Ensure the local disks or multipath LUNs you want to configure as software RAID devices are available as mapped devices.

   ```
   # ls /dev/mapper
   ```

3. Run the `multipath -ll` command to find the WWIDs for the devices, as follows:

   ```
   # multipath -ll

   device1-WWID dm-0 LSI,MR9261-8i
   size=558G features='1 queue_if_no_path' hwhandler='0' wp=rw
   `-+- policy='round-robin 0' prio=1 status=active
   `- 2:2:1:0 sdb 8:16 active ready running

   device2-WWID dm-1 LSI,MR9261-8i
   size=558G features='1 queue_if_no_path' hwhandler='0' wp=rw
   `-+- policy='round-robin 0' prio=1 status=active
   `- 2:2:2:0 sdc 8:32 active ready running
   ```

   Note

   The multipathing service, `multipathd`, uses the underlying devices to create a single device that routes I/O from Oracle VM Server to those underlying devices. For this reason, you should not use the `udev` device names to create a software RAID, such as `/dev/sdb`. You should only the WWIDs of the devices to create a software RAID. If you attempt to use a `udev` device name, an error occurs to indicate that the device is busy.

4. Create a software RAID configuration with the devices.

   ```
   # mdadm --create --verbose /dev/md0 --level=1 --raid-devices=2 \
     /dev/mapper/device1-WWID /dev/mapper/device2-WWID
   ```

5. Open `/etc/mdadm.conf` for editing.
6. Comment out the `DEVICE /no/device` line.
7. Specify each device to include in the software RAID configuration on separate `DEVICE` lines, as in the following example:

   ```
   DEVICE /dev/mapper/device1-WWID
   DEVICE /dev/mapper/device2-WWID
   ```

8. Save and close `/etc/mdadm.conf`.
9. Run the following command to scan for software RAID devices and include them in `mdadm.conf`:

   ```
   # mdadm --detail --scan >> /etc/mdadm.conf
   ```

   Note

   This command is optional. However, including the software RAID devices in `mdadm.conf` helps the system assemble them at boot time.

If any software RAID devices already exist, this command creates duplicate entries for them in `mdadm.conf`. In this case, you should use a different method to include the new software RAID device, as in the following example:

```
# mdadm --detail --scan

ARRAY /dev/md0 metadata=1.2 name=hostname UUID=RAID1_UUID
ARRAY /dev/md1 metadata=1.2 name=hostname UUID=RAID2_UUID
ARRAY /dev/md2 metadata=1.2 name=hostname UUID=RAID3_UUID

# cp /etc/mdadm.conf /etc/mdadm.conf.backup

# echo "ARRAY /dev/md2 metadata=1.2 name=hostname UUID=RAID3_UUID" >> /etc/mdadm.conf
```

10. Confirm that the configuration includes the software RAID device.

```
# cat /etc/mdadm.conf
# For OVS, don't scan any devices
#DEVICE /no/device
DEVICE /dev/mapper/device1-WWID
DEVICE /dev/mapper/device2-WWID

ARRAY /dev/md0 metadata=1.2 name=hostname UUID=RAID_UUID
```

11. Check the status of the software RAID device.

```
# mdadm --detail /dev/md0
    /dev/md0:
          Version : 1.2
    Creation Time : time_stamp
       Raid Level : raid1
       Array Size : 55394112 (52.83 GiB 56.72 GB)
    Used Dev Size : 55394112 (52.83 GiB 56.72 GB)
     Raid Devices : 2
    Total Devices : 2
      Persistence : Superblock is persistent

      Update Time : time_stamp
            State : clean
   Active Devices : 2
  Working Devices : 2
   Failed Devices : 0
    Spare Devices : 0

             Name : hostname:0
             UUID : RAID_UUID
           Events : 17

      Number   Major   Minor   RaidDevice State
         0      251        0        0      active sync   /dev/dm-0
         1      251        1        1      active sync   /dev/dm-1
```

You can find more information about software RAID in the Oracle Linux documentation at:

https://docs.oracle.com/cd/E37670_01/E41138/html/ch18s04.html

1.3.1 Removing Software RAID Devices

You cannot use Oracle VM Manager to remove software RAID devices. You must manually remove these devices on Oracle VM Server as follows:

1. Connect to Oracle VM Server as the root user.
2. Stop the software RAID device.

```
# mdadm --stop /dev/md0
```

3. Remove the software RAID superblock from the devices.

   ```
   # mdadm --zero-superblock /dev/mapper/device1-WWID /dev/mapper/device2-WWID
   ```

4. Remove the software RAID device from `/etc/mdadm.conf`.

5. Remove the software RAID device from Oracle VM Manager.

> **Note**
>
> After you remove the software RAID device, Oracle VM Manager displays an event with a severity of warning. The event message is similar to the following:
>
> ```
> Warning time_stamp storage.device.offline. Physical disk is Offline
>     No Description: OVMEVT_007005D_001 Rescan storage layer on server [hostname] did
>     not return physical disk [md-UUID] for storage array [Generic Local Storage Array
> ```
>
> You can ignore this warning.

1.4 Diagnostic Tools for Oracle VM Server

As an optional post-installation step, Oracle recommends that you also install and configure diagnostics tools on all Oracle VM Servers. These tools can be used to help debug and diagnose issues such as system crashes, hanging, unscheduled reboots, and *OCFS2* cluster errors. The output from these tools can be used by Oracle Support and can significantly improve resolution and response times.

Obtaining a system memory dump, vmcore, can be very useful when attempting to diagnose and resolve the root cause of an issue. To get a useful vmcore dump, a kdump service configuration is required. See Section 1.4.2, “Manually Configuring kdump for Oracle VM Server” below for more information on this.

In addition, you can install netconsole, a utility allowing system console messages to be redirected across the network to another server. See the Oracle Support Document, *How to Configure "netconsole" for Oracle VM Server 3.0*, for information on how to install netconsole.

https://support.oracle.com/

Additional information on using diagnostic tools is provided in the Oracle Linux documentation. See the chapter titled *Support Diagnostic Tools* in the *Oracle Linux Administrator's Solutions Guide*.

https://docs.oracle.com/cd/E37670_01/E37355/html/ol_diag.html

1.4.1 Working with the OSWatcher Utility on Oracle VM Server

OSWatcher (oswbb) is a collection of shell scripts that collect and archive operating system and network metrics to diagnose performance issues with Oracle VM Server. OSWatcher operates as a set of background processes to gather data with standard UNIX utilities such as vmstat, netstat and iostat.

By default, OSWatcher is installed on Oracle VM Server and is enabled to run at boot. The following table describes the OSWatcher program and main configuration file:

Name	Description
`/usr/sbin/OSWatcher`	The main OSWatcher program. If required, you can configure certain parameters for statistics collection. However, you should do so only if Oracle Support advises you to change the default configuration.

Name	Description
`/etc/sysconfig/oswatcher`	This file defines the directory where OSWatcher log files are saved, the interval between statistics collection, and the maximum amount of time to retain archived statistics. **Important** It is not possible to specify a limit to the data that the OSWatcher utility collects. For this reason, you should be careful when modifying the default configuration so that the OSWatcher utility does not use all available space on the system disk.

To start, stop, and check the status of OSWatcher, use the following command:

```
# service oswatcher {start|stop|status|restart|reload|condrestart}
```

For detailed information on the data that OSWatcher collects and how to analyze the output, as well as for instructions on sending the data to Oracle Support, see the *OSWatcher User Guide* in the following directory on Oracle VM Server: `/usr/share/doc/oswatcher-x.x.x/`

1.4.2 Manually Configuring kdump for Oracle VM Server

While Oracle VM Server uses the robust UEK4 kernel which is stable and fault-tolerant and should rarely encounter errors that crash the entire system, it is still possible that a system-wide error results in a kernel crash. Information about the actual state of the system at the time of a kernel crash is critical to accurately debug issues and to resolve them. The kdump service is used to capture the memory dump from dom0 and store it on the filesystem. The service does not dump any system memory used by guest virtual machines, so the memory dump is specific to dom0 and the Xen hypervisor itself. The memory dump file that is generated by kdump is referred to as the `vmcore` file.

A description of the actions required to manually configure Oracle VM Server so that the kdump service is properly enabled and running is provided here, so that you are able to set up and enable this service after an installation. The Oracle VM Server installer provides an option to enable kdump at installation where many of these steps are performed automatically. See the *Kdump Setting* section of the *Oracle VM Manager Command Line Interface User's Guide* for more information on this.

Checking Pre-requisite Packages

By default, the required packages to enable the kdump service are included within the Oracle VM Server installation, but it is good practice to check that these are installed before continuing with any configuration work. You can do this by running the following command:

```
# rpm -qa | grep kexec-tools
```

If the `kexec-tools` package is not installed, you must install it manually.

Updating the GRUB2 Configuration

Oracle VM Server makes use of GRUB2 to handle the boot process. In this step, you must configure GRUB2 to pass the `crashkernel` parameter to the Xen kernel at boot. This can be done by editing the `/etc/default/grub` file and modifying the `GRUB_CMDLINE_XEN` variable by appending the appropriate `crashkernel` parameter.

The `crashkernel` parameter specifies the amount of space used in memory to load the crash kernel that is used to generate the dump file, and also specifies the offset which is the beginning of the crash kernel region in memory. The minimum amount of RAM that may be specified for a crash kernel is 512 MB and this should be offset by 64 MB. This would result in a configuration that looks similar to the following:

```
GRUB_CMDLINE_XEN="dom0_mem=max:6144M allowsuperpage dom0_vcpus_pin \
dom0_max_vcpus=20 crashkernel=512M@64M"
```

This setting is sufficient for the vast majority of systems, however on systems that make use of a significant number of large drivers, the crash kernel may need to be allocated more space in memory. If you force a dump and it fails to generate a core file, you may need to increase the amount of memory allocated to the crash kernel.

> **Important**
>
> While UEK4 supports the `crashkernel=auto` option, the Xen hypervisor does not. You must specify values for the RAM reservation and offset used for the crash kernel or the kdump service is unable to run.

When you have finished modifying `/etc/default/grub`, you must rebuild the system GRUB2 configuration that is used at boot time. This is done by running:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

Optionally Preparing a Local Filesystem to Store Dump Files

Kdump is able to store vmcore files in a variety of locations, including network accessible filesystems. By default, vmcore files are stored in `/var/crash/`, but this may not be appropriate depending on your disk partitioning and available space. The filesystem where the vmcore files are stored must have enough space to match the amount of memory available to Oracle VM Server for each dump.

Since the installation of Oracle VM Server only uses as much disk space as is required, a 'spare' partition is frequently available on a new installation. This partition is left available for use for hosting a local repository or for alternate use such as for hosting vmcore files generated by kdump. If you opt to use it for this purpose, you must first correctly identify and take note of the UUID of the partition and then format it with a usable filesystem.

The following steps serve as an illustration of how you might prepare the local spare partition.

- Identify the partition that the installer left 'spare' after the installation. This is usually listed under `/dev/mapper` with a filename that starts with `OVM_SYS_REPO_PART`. If you can identify this device, you can format it with an ext4 filesystem:

  ```
  # mkfs.ext4 /dev/mapper/OVM_SYS_REPO_PART_VBd64a21cf-db4a5ad5
  ```

  If you don't have a partition mapped like this, you may need to use a utilities like `blkls`, `parted`, `fdisk` or `gdisk` to identify any free partitions on your available disk devices.

- Obtain the UUID for the filesystem. You can do this by running the `blkid` command:

  ```
  # blkid /dev/mapper/OVM_SYS_REPO_PART_VBd64a21cf-db4a5ad5
  /dev/mapper/OVM_SYS_REPO_PART_VBd64a21cf-db4a5ad5:
  UUID="51216552-2807-4f17-ab27-b8135f69896d" TYPE="ext4"
  ```

  Take note of the UUID as you will need to use this later when you configure kdump.

Modifying the kdump Configuration

System configuration directing how the kdump service runs is defined in `/etc/sysconfig/kdump`, while specific kdump configuration variables are defined in `/etc/kdump.conf`. Changes may need to be made to either of these files depending on your environment. However, the default configuration should be sufficient to run kdump initially without any problems. The following list identifies potential configuration changes that you may wish to make:

- On systems with lots of memory (e.g. over 1 TB), it is advisable to disable the IO Memory Management Unit within the crash kernel for performance and stability reasons. This is achieved by editing /etc/sysconfig/kdump and appending the iommu=off kernel boot parameter to the KDUMP_COMMANDLINE_APPEND variable:

  ```
  KDUMP_COMMANDLINE_APPEND="irqpoll maxcpus=1 nr_cpus=1 reset_devices cgroup_disable=memory
        mce=off selinux=0 iommu=off"
  ```

- If you intend to change the partition where the vmcore files are stored, using the spare partition on the server after installation for instance, you must edit /etc/kdump.conf to provide the filesystem type and device location of the partition. If you followed the instructions above, it is preferable that you do this by specifying the UUID that you obtained for the partition using the blkid command. A line similar to the following should appear in the configuration:

  ```
  ext4 UUID=51216552-2807-4f17-ab27-b8135f69896d
  ```

- You may edit the default path where vmcore files are stored, but note that this path is relative to the partition that kdump is configured to use to store vmcores. If you have configured kdump to store vmcores on a separate filesystem, when you mount the filesystem, the vmcore files are located in the path specified by this directive on the mounted filesystem:

  ```
  path /var/crash
  ```

- If you are having issues obtaining a vmcore or you are finding that your vmcore files are particularly large using the makedumpfile utility, you may reconfigure kdump to use the cp command to copy the vmcore in sparse mode. To do this, edit /etc/kdump.conf to comment out the line containing setting the core_collector to use the makedumpfile utility and uncomment the lines to enable the cp command:

  ```
  # core_collector makedumpfile -EXd 1 --message-level 1 --non-cyclic
  core_collector cp --sparse=always
  extra_bins /bin/cp
  ```

  Your mileage with this may vary, and the makedumpfile utility is generally recommended instead.

Enabling the kdump Service

You can enable the kdump service to run at every boot by running the following command:

```
# chkconfig kdump on
```

You must restart the kdump service at this point to allow it to detect the changes that have been made to the kdump configuration and to determine whether a kdump crash kernel has been generated and is up to date. If the kernel image needs to be updated, kdump does this automatically, otherwise it restarts without any attempt to rebuild the crash kernel image:

```
# service kdump restart
Stopping kdump:                   [  OK  ]
 Detected change(s) the following file(s):
    /etc/kdump.conf
Rebuilding /boot/initrd-4.1.12-25.el6uek.x86_64kdump.img

Starting kdump:                   [  OK  ]
```

Confirming that kdump is Configured and Working Correctly

You can confirm that the kernel loaded for dom0 is correctly configured, by running the following command and checking that output is returned to show that your crashkernel parameter is in use:

```
# xl dmesg|grep -i crashkernel
(XEN) Command line: placeholder dom0_mem=max:6144M allowsuperpage dom0_vcpus_pin
        dom0_max_vcpus=20 crashkernel=512M@64M
```

You can also check that the appropriate amount of memory is reserved for kdump by running the following:

```
# xl dmesg|grep -i kdump
(XEN) Kdump: 512MB (524288kB) at 0x4000000
```

or alternately:

```
# kexec --print-ckr-size
536870912
```

You can check that the kdump service is running by checking the service status:

```
# service kdump status
Kdump is operational
```

If there are no errors in `/var/log/messages` or on the console, you can assume that kdump is running correctly.

To test that kdump is able to generate a vmcore and store it correctly, you can trigger a kernel panic by issuing the following commands:

```
# echo 1 > /proc/sys/kernel/sysrq
# echo c > /proc/sysrq-trigger
```

Note

These commands cause the kernel on the Oracle VM Server to panic and crash. If kdump is working correctly, the crash kernel should take over and generate the vmcore file which is copied to the configured location before the server reboots automatically. If kdump fails to load the crash kernel, the server may hang with the kernel panic and requires a hard-reset to reboot.

After you have triggered a kernel panic and the system has successfully rebooted, you may check that the vmcore file was properly generated:

- If you have not configured kdump to use an alternate partition, you should be able to locate the vmcore file in `/var/crash/127.0.0.1-date-time/vmcore`, where *`date`* and *`time`* represent the date and time when the vmcore was generated.

- If you configured kdump to use an alternate partition to store the vmcore file, you must mount it first. If you used the spare partition generated by a fresh installation of Oracle VM Server, this can be done in the following way:

  ```
  # mount /dev/mapper/OVM_SYS_REPO_PART_VBd64a21cf-db4a5ad5 /mnt
  ```

  You may then find the vmcore file in `/mnt/var/crash/127.0.0.1-date-time/vmcore`, where *`date`* and *`time`* represent the date and time when the vmcore was generated, for example:

  ```
  # file /mnt/var/crash/127.0.0.1-2015-12-08-16\:12\:28/vmcore
  /mnt/var/crash/127.0.0.1-2015-12-08-16:12:28/vmcore: ELF 64-bit LSB
        core file x86-64, version 1 (SYSV), SVR4-style
  ```

  Remember to unmount the partition after you have obtained the vmcore file for analysis, so that it is free for use by kdump.

If you find that a vmcore file is not being created or that the system hangs without automatically rebooting, you may need to adjust your configuration. The most common problem is that there is insufficient memory allocated for the crash kernel to run and complete its operations. Your starting point to resolving issues with kdump is always to try increasing the reserved memory that is specified in your GRUB2 configuration.

1.5 Disabling Paravirtualized Guests on Oracle VM Server

Paravirtualization (PVM) is considered a less secure guest domain type. To keep your virtualized environment safe and secure, you should prevent paravirtualized guest VMs from starting and running within Oracle VM.

As of Release 3.4.5, the Xen hypervisor allows you to disable PVM guests through a configuration file setting. After you upgrade your servers to Oracle VM Server Release 3.4.5, PVM guests are not disabled by default, because that would cause a variety of problems in existing PVM guests. Oracle recommends that you switch to PV-HVM guests and disable PVM guests as described in this section.

As of Release 3.4.6, support for PVM guests is removed. With the removal of PVM guest support, the following new behavior restrictions exist:

- A new virtual machine cannot be created of the PVM doman type from the Oracle VM Manager Web Interface, Oracle VM Manager Command Line Interface, or Oracle VM Web Services API.
- An existing virtual machine of the PVM domain type can be converted to a supported type from the Oracle VM Manager Web Interface, Oracle VM Manager Command Line Interface, or Oracle VM Web Services API.
- During server discovery, warnings are raised for each virtual machine of the PVM domain type. The warnings appear of type "vm.unsupported.domain" on the **Error Conditions** subtab of the **Health** tab. The error event cannot be acknowledged by the user.

  Note

  Existing virtual machines of the PVM domain type continue to work as before; however, the error event that is raised goes away only after the PVM domain type issue is resolved.

- After editing the domain type to a supported type, the event is then acknowledged.

Tip

If you have existing PVM guests, you should convert them to HVM with PV drivers before you disable PVM on your Oracle VM Servers. For details about changing the guest virtualization mode, please consult the Support Note with ID 2247664.1.

Disabling PVM Guests on Oracle VM Server

1. Using SSH, log into the Oracle VM Server.
2. Open the file `xend-config.sxp` and locate the entry "`xend-allow-pv-guests`".

   ```
   vi /etc/xen/xend-config.sxp
   # -*- sh -*-
   #
   # Xend configuration file.
   [...]
   #
   # By default allow PV guests to be created
   #(xend-allow-pv-guests 1)
   ```

3. Uncomment the line by removing the "`#`" and set the parameter to "0" to disable PV guests. Save the changes to the file.

```
# By default allow PV guests to be created
(xend-allow-pv-guests 0)
```

4. Stop and start the xend service on the Oracle VM Server for the new settings to take effect.

   ```
   # service xend stop
   # service xend status
   xend daemon is stopped

   # service xend start
   # service xend status
   xend daemon (pid 9641) is running...
   ```

   Any attempt to start a PVM guest on an Oracle VM Server with PVM guests disabled, or to migrate a PVM guest to it, results in a failure: "`Error: PV guests disabled by xend`".

   Note

   If secure VM migration is enabled – which is the default setting –, the wrong error message may be displayed. A known issue may lead to a confusing error message containing "[Errno 9] Bad file descriptor".

5. Repeat these steps for each of the remaining Oracle VM Servers to protect your entire virtualized environment.

1.6 Changing the Memory Size of the Management Domain

When you install Oracle VM Server, the installer sets a default memory size for *dom0*. The algorithm used is:

(768 + 0.0205 * Physical memory (MB)) round to 8

You can use this calculation to determine memory allocation for the Oracle VM Server installation. However, you should not make the memory allocation for dom0 smaller than the calculated value. You can encounter performance issues if the dom0 memory size is not set appropriately for your needs on the Oracle VM Server.

Example sizes are set out in table Table 1.1.

Table 1.1 Default Dom0 Memory Size

Physical Memory	Dom0 Memory
2 GB	816 MB
4 GB	856 MB
8 GB	936 MB
16 GB	1104 MB
32 GB	1440 MB
64 GB	2112 MB
128 GB	3456 MB
256 GB	6144 MB
512 GB	11520 MB
1024 GB	22264 MB
2048 GB	32768 MB

Physical Memory	Dom0 Memory
	Note 32768 MB is the maximum allowed memory for dom0.

To change the dom0 memory allocation, edit your grub configuration on the Oracle VM Server to adjust the value for the *dom0_mem* parameter. If you are using UEFI boot, the grub configuration file is located at /boot/efi/EFI/redhat/grub.cfg, otherwise the grub configuration file is located at /boot/grub2/grub.cfg. Edit the line starting with multiboot2 /xen.mb.efi and append the required boot parameters. For example, to change the memory allocation to 1440 MB, edit the file to contain:

```
multiboot2 /xen.mb.efi dom0_mem=max:1440M placeholder ${xen_rm_opts}
```

1.7 Configuring Oracle VM Server for SPARC

This section describes configuration tasks for Oracle VM Server for SPARC only.

Note

Access the Oracle VM Server for SPARC documentation at https://www.oracle.com/technetwork/documentation/vm-sparc-194287.html. To determine the version of the Oracle VM Server for SPARC documentation to reference, run the pkg list ldomsmanager command.

1.7.1 Creating ZFS Volumes

Local ZFS volumes are supported as local physical disks on Oracle VM Server for SPARC. While Oracle VM Manager does not provide tools to create or manage ZFS volumes, it does detect ZFS volumes as local physical disks that can either be used for virtual disks by the virtual machines hosted on the Oracle VM Server where the volume resides, or for use as a local repository to store virtual machine resources. In this section, we describe the steps required to manually create ZFS volumes on a SPARC-based Oracle VM Server and how to detect these within Oracle VM Manager.

Note

See Creating ZFS Volumes on NVMe Devices if you plan to create a ZFS volume on an NVMe devices, such as an SSD.

In the control domain for the Oracle VM Server where you wish to create the ZFS volumes that you intend to use, use the zfs create command to create a new ZFS volume:

```
# zfs create -p -V XG pool/OVS/volume
```

The size of the volume, represented by *X*G can be any size that you require as long as your hardware supports it. The *pool*, that the volume belongs to can be any ZFS pool. Equally, the *volume* name can be of your choosing. The only requirement is that the volume resides under *OVS* within the pool, so that Oracle VM Manager is capable of detecting it. The following example shows the creation of two ZFS volumes of 20 GB in size:

```
# zfs create -V 20G rpool/OVS/DATASET0
# zfs create -V 20G rpool/OVS/DATASET1
```

Once you have created the ZFS volumes that you wish to use, you must rediscover your server within Oracle VM Manager. See the *Discover Servers* section of the *Oracle VM Manager User's Guide* for more information on how to do this. Once the server has been rediscovered, the ZFS volumes appear as physical disks attached to the server in the **Physical Disks** perspective within the Oracle VM Manager

Web Interface. See the *Physical Disks Perspective* section of the *Oracle VM Manager User's Guide* for more information on this perspective.

As long as a ZFS volume is unused and Oracle VM Manager is able to detect it as a local physical disk attached to the server, you can create a repository on the ZFS volume by selecting to use this disk when you create the repository. See the *Create New Repository* section of the *Oracle VM Manager User's Guide* on creating repositories.

Using this feature, you can use a single SPARC server to create virtual machines without any requirement to use an NFS repository or any additional physical disks.

Creating ZFS Volumes on NVMe Devices

If you plan to create a ZFS volume on an NVM Express (NVMe) device, use the following procedure:

1. Determine the LUN of the NVMe device with the `format` command, as in the following example:

   ```
   # format

   ....
       5. c1t1d0 <INTEL-SSDPE2ME016T4S-8DV1-1.46TB>
             /pci@306/pci@1/pci@0/pci@4/nvme@0/disk@1
             /dev/chassis/SYS/DBP/NVME0/disk
   ...
   ```

   In the preceding example, the NVMe device has the following LUN: `c1t1d0`.

   Note

   In most cases, NVMe devices have the following path: `/SYS/DBP/NVME[0..n]`.

2. Create a ZFS pool with the NVMe device, as follows:

   ```
   # zpool create pool_name c1t1d0
   ```

   Where:

   - *`pool_name`* is any valid ZFS pool name.
   - *`c1t1d0`* is the LUN of the NVMe device.

3. Create a ZFS volume on the ZFS pool, as follows:

   ```
   # zfs create -p -V sizeG pool_name/OVS/volume_name
   ```

   Where:

   - *`size`* is an integer value that specifies the size of the ZFS volume in Gigabytes. Ensure that the size of the ZFS pool is not greater than the size of the NVMe disk.
   - *`pool_name`* is the name of the ZFS pool on which you are creating the volume.
   - *`volume_name`* is the name of the ZFS volume.

   Important

   The first path element of the ZFS pool must be `/OVS/` as in the preceding example. This path element ensures that Oracle VM Manager discovers the ZFS volume as a local physical disk.

4. Repeat the preceding step to create additional ZFS volumes, as required.

5. From Oracle VM Manager, discover, or re-discover, the instance of Oracle VM Server for SPARC that has the NVMe device attached to it.

   After the discovery process completes, the Oracle VM Manager Web Interface displays each ZFS volume as a physical disk attached to Oracle VM Server for SPARC in the **Physical Disks** perspective. See the *Physical Disks Perspective* section of the *Oracle VM Manager User's Guide* .

1.7.2 Configuring a Secondary Service Domain

The default configuration of the Oracle VM Agent uses a single service domain, the primary domain, which provides virtual disk and virtual network services to guest virtual machines (guest domains). To increase the availability of guest domains, you can configure a secondary service domain to provide virtual disk and virtual network services through both the primary and the secondary service domains. With such a configuration, guest domains can use virtual disk and virtual network multipathing and continue to be fully functional even if one of the service domains is unavailable.

The primary domain is always the first service domain and this is the domain that is discovered by Oracle VM Manager. The second service domain, named secondary, is a root domain that is configured with a PCIe root complex.The secondary domain should be configured similarly to the primary domain; it must use the same operating system version, same number of CPUs and same memory allocation. Unlike the primary domain, the secondary service domain is not visible to Oracle VM Manager. The secondary domain mimics the configuration of the primary service domain and is transparently managed by the Oracle VM Agent. In the case where the primary service domain becomes unavailable, the secondary service domain ensures that guest domains continue to have access to virtualized resources such as disks and networks. When the primary service domain becomes available again, it resumes the role of managing these resources.

From a high level, the following tasks should be performed to configure the Oracle VM Agent to use a secondary service domain:

1. Install the Oracle VM Agent as described in the *Installing Oracle VM Agent for SPARC* section of the *Oracle VM Installation and Upgrade Guide* .

2. Create the secondary service domain.

3. Install the secondary service domain.

4. Configure the Oracle VM Agent to use the secondary service domain.

Note

If you have a secondary service domain already configured and you have successfully updated your system to Oracle Solaris 11.3 on the primary domain, the secondary service domain can also be upgraded using the same Oracle Solaris IPS repository as the primary domain. To upgrade the secondary service domain, you should upgrade from the Oracle Solaris command line using the following command:

```
# pkg update --accept
```

Reboot the system after the upgrade completes, as follows:

```
# init 6
```

For detailed install and upgrade instructions for Oracle Solaris 11.3, see https://docs.oracle.com/cd/E53394_01/.

1.7.2.1 Requirements

To configure the Oracle VM Agent with a secondary service domain, your SPARC server must meet the minimum requirements listed in this section, in addition to the standard installation requirements described in the *Installing Oracle VM Server on SPARC Hardware* section of the *Oracle VM Installation and Upgrade Guide* .

Hardware

Use a supported Oracle SPARC T-series server, M-series, or S-series server. See *Supported Platforms* in the *Oracle VM Server for SPARC Installation Guide*. The SPARC server must have at least two PCIe buses, so that you can configure a root domain in addition to the `primary` domain. For more information, see *I/O Domain Overview* in the *Oracle VM Server for SPARC Administration Guide*.

Both domains must be configured with at least one PCIe bus. The PCIe buses that you assign to each domain must be unique. You cannot assign the same PCIe bus to two different domains.

By default, after a fresh installation, all PCIe buses are assigned to the primary domain. When adding a new service domain, some of these PCIe buses must be released from the primary domain and then assigned to the secondary domain.

For example, a SPARC T5-2 server with two SPARC T5 processors has 4 PCIe buses. This server can be configured with a primary domain and a secondary domain. You can assign two PCIe buses to the primary domain, and two PCIe buses to the secondary domain.

Network

The network ports used by the primary domain must all be connected to the PCIe buses that are assigned to the primary domain.

Similarly the network ports used by the secondary domain must all be connected to the PCIe buses that are assigned to the secondary domain.

In addition, the primary and secondary domains must have the same number of network ports. Each network port in the primary domain must have a corresponding network port in the secondary domain, and they must be connected to the same physical network.

For example, a SPARC T5-2 server with two SPARC T5 processors has 4 PCIe buses (pci_0, pci_1, pci_2, and pci_3). The server also has 4 onboard network ports. Two network ports are connected to pci_0, and the other two are connected to pci_3. You can assign 2 PCIe buses (pci_0 and pci_1) to the primary domain, and 2 PCIe buses (pci_2 and pci_3) to the secondary domain. That way, both domains have two ports configured. You must ensure that each port is connected to the same physical network as the port in the corresponding domain.

Storage

Physical disks or LUNs used by the primary domain must all be accessible through one or several host bus adapters (HBAs) connected to the PCIe buses that are assigned to the primary domain. The primary domain requires at least one disk for booting and hosting the operating system. The primary domain usually has access to all, or a subset of, local SAS disks present on the server through an onboard SAS HBA connected to one of the PCIe buses of the server.

Similarly, physical disks or LUNs used by the secondary domain must all be accessible through one or several HBAs connected to the PCIe buses assigned to the secondary domain. The secondary domain needs at least one disk for booting and hosting the operating system. Depending on the server used, the secondary domain might not have access to any local SAS disks present on the server, or it might have access to a subset of the local SAS disks. If the secondary domain does not have access to any of the local SAS disks then it must have an HBA card on one of its PCIe buses and access to an external storage array LUN that it can use for booting.

> **Warning**
>
> If the boot disk of the secondary domain is on a storage array shared between multiple servers or multiple domains, make sure that the boot disk is accessible by the secondary domain only. Otherwise the disk might be used by mistake by another server or domain, which can corrupt the boot disk of the secondary domain. Depending on the storage array and the storage area network, this can usually be achieved using zoning or LUN masking.

In addition, if a Fibre Channel (FC) storage area network (SAN) is used, then the primary and the secondary domains must have access to the same FC disks. So one or more FC HBAs must be connected to the FC SAN and to the PCIe buses that are assigned to the primary domain. And, one or more FC HBAs must be connected to the FC SAN and to the PCIe buses that are assigned to the secondary domain.

> **Note**
>
> The primary and the secondary domain do not need to have access to same SAS or iSCSI disks. Only the SAS or iSCSI disks accessible from the primary domain are visible to Oracle VM Manager. Oracle VM Manager does not have visibility of any SAS or iSCSI disks accessible only from the secondary domain. If a virtual machine is configured with SAS or iSCSI disks, then the corresponding virtual disks in the virtual machine have a single access path, through the primary domain. If a virtual machine is configured with FC disks, then the corresponding virtual disks in the virtual machine have two access paths: one through the primary domain; and one through the secondary domain.

For example, a SPARC T5-2 server with two SPARC T5 processors has 4 PCIe buses (pci_0, pci_1, pci_2, pci_3). The server also has 2 onboard SAS HBAs to access the 6 internal SAS disks. One SAS HBA is connected to PCIe bus pci_0 and accesses 4 internal disks. The other SAS HBA is connected to PCIe bus pci_4 and accesses the 2 other internal SAS disks. You can assign 2 PCIe buses (pci_0 and pci_1) to the primary domain, and 2 PCIe buses (pci_2 and pci_3) to the secondary domain. That way, both domains have access to internal SAS disks that can be used for booting. The primary domain has access to four SAS disks, and the secondary domain has access to two SAS disks.

If you want to connect the server to an FC SAN, then you can add an FC HBA to the primary domain (for example on PCIe bus pci_1) and an FC HBA to the secondary domain (for example, on PCIe bus pci_2). Then you should connect both FC HBAs to the same SAN.

1.7.2.2 Limitations

While using secondary service domains can improve the availability of guest virtual machines, there are some limitations to using them with Oracle VM. The following list outlines each of these limitations:

- **Clustering:** Clustering cannot be used with a secondary service domain. If a server is configured with a secondary service domain then that server cannot be part of a clustered server pool.

- **Network Configuration:** Network bonds/aggregation and VLANs are not automatically configured on the secondary domain. If you configure bonds/aggregation or VLANs on the primary domain using Oracle VM Manager, then corresponding bonds/aggregation or VLANs are not automatically configured on the secondary domain. To use any such bond/aggregation or VLANs with virtual machines, the corresponding bonds/aggregation or VLANs must be manually configured on the secondary domain.

- **Storage:** NFS, SAS, iSCSI, and ZFS volumes accessible only from the secondary domain cannot be used or managed using Oracle VM Manager.

> **Important**
>
> Secondary service domains cannot access NFS repositories. For this reason, virtual machine I/O to virtual disks is served by the control domain only. If the control domain stops or reboots, virtual machine I/O to virtual disks is suspended until the control domain resumes operation. Use physical disks (LUNs) for virtual machines that require continuous availability during a control domain reboot.

- **Virtual Machine Disk Multipathing:** When assigning a disk to a virtual machine, only fibre channel (FC) disks are configured with disk multipathing through the primary and the secondary domains. NFS, SAS, iSCSI or ZFS disks assigned to a virtual machine are configured with a single path through the primary domain.

- **Virtual Machine Network Port:** When assigning a network port to a virtual machine, two network ports are effectively configured on the virtual machine: one connected to the primary domain, and one connected to the secondary domain. The network port connected to the primary domain is configured with a MAC address that can be defined from within Oracle VM Manager. The MAC address must be selected in the range [00:21:f6:00:00:00, 00:21:f6:0f:ff:ff]. The network port connected to the secondary domain is configured with a MAC address derived from the MAC address of the network port connected to the primary domain. This MAC address starts with 00:21:f6:8.

  For example, if the MAC address defined in Oracle VM Manager is 00:21:f6:00:12:34 then this MAC address is used on the network port connected to the primary domain. The derived MAC address is then 00:21:f6:80:12:34 and should be used on the network port connected to the secondary domain. Oracle VM Manager uses a default dynamic MAC address range of [00:21:f6:00:00:00, 00:21:f6:ff:ff:ff]. When using a secondary service domain, this range must be reduced to [00:21:f6:00:00:00, 00:21:f6:0f:ff:ff]. See the *Virtual NICs* section in the Oracle VM Manager Online Help for more information on changing the default range of MAC addresses within the Oracle VM Manager Web Interface.

- **Live Migration:** A virtual machine cannot be live migrated to a server configured with a different number of service domains. In other words, you cannot migrate a virtual machine running on a server with a secondary service domain to a server without a secondary service domain; and you cannot migrate a virtual machine running on a server without a secondary service domain to a server with a secondary service domain.

1.7.2.3 Creating a Secondary Service Domain

The following requirements apply to secondary service domains within an Oracle VM context:

- No domain, other than the primary domain, must exist before you start to set up a secondary domain. You can see all existing domains in the output of the `ldm list` command.

- No virtual switch must exist before you start to set up a secondary domain. You can see all virtual switches in the VSW section in the output of the `ldm list-services` command.

- The name of the secondary service domain must be `secondary`.

- The secondary service domain should be a root domain.

- The secondary service domain should be configured with 1 CPU core.

- The secondary service domain should be configured with 8 GB of memory.
- The secondary service domain should have virtual disk service (VDS) with the name `secondary-vds0`.
- The secondary service domain should be completely independent of any other domain, in particular of the primary domain. For this reason, the secondary domain should have no virtual disks and no virtual network interfaces, and use only physical disks and physical network interfaces.

For more information about creating a root domain, see *Creating a Root Domain by Assigning PCIe Buses* in the *Oracle VM Server for SPARC Administration Guide*.

Use the `ovs-agent-secondary` command to make sure that you meet these requirements, and to simplify the process of setting up and configuring the secondary service domain. See Section 1.7.2.6, “Automatically Creating and Setting Up a Secondary Domain”.

The following instructions describe how to create a secondary service domain manually:

Manually Creating a Secondary Service Domain

1. Create the service domain and set the core CPU and memory requirements using the following commands:

   ```
   # ldm add-domain secondary
   # ldm set-core 1 secondary
   # ldm set-memory 8g secondary
   ```

2. Assign the PCI buses that you wish the secondary service domain to use. For each bus, issue the following command, substituting *`pci_2`* with the correct bus identifier:

   ```
   ldm add-io pci_2 secondary
   ```

3. Add the secondary virtual disk service to the secondary domain, using the following command:

   ```
   ldm add-vds secondary-vds0 secondary
   ```

4. Remove any PCI buses that you added to the secondary service domain from the primary domain. To begin reconfiguring the primary domain, enter the following command:

   ```
   # ldm start-reconf primary
   ```

   For each bus that you added to the secondary domain, enter the following command to remove it from the primary domain, substituting *`pci_2`* with the correct bus identifier:

   ```
   # ldm remove-io pci_2 primary
   ```

5. When you have finished reconfiguring the primary domain, you must reboot it:

   ```
   # reboot
   ```

1.7.2.4 Installing the Secondary Service Domain

After the secondary service domain has been created and the primary domain has finished rebooting, start the secondary service domain using the following commands in the control domain:

```
# ldm bind-domain secondary
# ldm start-domain secondary
```

Once the secondary service domain has been started, you can access its console by obtaining the console port using the following command:

```
# ldm list secondary
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
secondary        active     -t--v-  5000    8     8G       0.0%  0.0%  0s
```

Note that the console port is listed in the CONS column. You can open a telnet connection to this port as follows:

```
# telnet 0 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "secondary" in group "secondary" ....
Press ~? for control options ..

{0} ok
```

Now you must install the Oracle Solaris 11 operating system into the secondary domain. This can be achieved by following the instructions provided in the *Oracle Solaris 11.3* documentation available at:

https://docs.oracle.com/cd/E53394_01/html/E54756/index.html

Do ***not*** attempt to install either the Oracle VM Agent or the Logical Domains Manager into the secondary service domain. Only the Oracle Solaris 11 operating system is required.

Make sure that the secondary service domain is properly configured so that it can boot automatically. In particular, the OpenBoot PROM (OBP) variables of the domain must be correctly set. For instance, the `auto-boot?` parameter should be set to **true**, and `boot-device` parameter should contain the device path of the boot disk that is configured for the secondary domain.

1.7.2.5 Manually Configuring the Oracle VM Agent to Support the Secondary Domain

You can use the `ovs-agent-secondary` command to assist you with the process of setting the Oracle VM Agent to support the secondary domain, see Section 1.7.2.6, “Automatically Creating and Setting Up a Secondary Domain”. The instructions that follow describe how to configure the Oracle VM Agent manually.

1. Create a configuration file in `/etc/ovs-agent/shadow.conf` on the primary domain. This configuration file is in JSON format and, at absolute minimum, should contain the following content to enable support for the secondary domain:

   ```
   {
       "enabled": true
   }
   ```

   Note

   Ensure that the JSON file is correctly formatted as defined at http://json.org/.

   - Each network link in the primary domain should have a corresponding network link in the secondary domain connected to the same physical network. By default, a network link in the primary domain is associated with the network link with the same name in the secondary domain. If a network link in the primary domain should be associated with a network link in the secondary domain with a different name, then you need to define a network link mapping. To define a network mapping, you need to add a '*nic-mapping*' entry in `/etc/ovs-agent/shadow.conf`. Typically, entries of this sort look similar to the following:

     ```
     {
         enabled": true,
     ```

```
    nic-mapping": [
      ["^net4$", "net2" ],
      ["^net5$", "net3" ]
    ]
}
```

In the above example, `net4` is a network interface in the primary domain and is connected to the same physical network as the network interface named `net2` in the secondary domain. Equally, `net5` is a network interface in the primary domain and is connected to the same physical network as the network interface named `net3` in the secondary domain. Note that network interface names in the primary domain are encapsulated with the regular expression characters caret (^) and dollar ($) to ensure an exact match for the network interface name in the primary domain.

- Each Fibre Channel (FC) disk accessible from the primary domain domain should also be accessible from the secondary domain. By default, a FC disk is accessed using the same device path in the primary domain and in the secondary domain. In particular, each disk is accessed using the same disk controller name. If a disk controller in the primary domain should be associated with a disk controller in the secondary domain with a different name, then you need to define a disk controller mapping.

  It is recommended that Solaris I/O multipathing is enabled in the primary and in the secondary domain on all multipath-capable controller ports, in particular on all FC ports. In that case, all FC disks appear under a single disk controller (usually c0), and disk controller mapping is usually not needed in that case.

  To define a disk controller mapping, add a '*disk-mapping*' entry in the `/etc/ovs-agent/shadow.conf` file. For example:

```
{
    "enabled": true,
    "disk-mapping": [
        [ "c0t", "c1t" ]
    ]
}
```

  In the preceding example, `c0t` is a disk controller in the primary domain that is connected to the same FC disk as the disk controller named `c1t` in the secondary domain.

- An example of an `/etc/ovs-agent/shadow.conf` file that requires both network interface and disk controller mapping follows:

```
{
    "enabled": true,
    "nic-mapping": [
        [ "^net4$", "net2" ],
        [ "^net5$", "net3" ]
    ],
    "disk-mapping": [
        [ "c0t", "c1t" ]
    ]
}
```

2. Save the logical domain configuration with the secondary service domain to the service processor.

> **Warning**
>
> Before saving the configuration, ensure that the secondary service domain is active. If the configuration is saved while the secondary service domain is not active, then the secondary service domain won't start automatically after a power-cycle of the server

```
# ldm add-spconfig ovm-shadow
```

3. To complete the configuration, reconfigure the Oracle VM Agent by running the following command:

   ```
   # ovs-agent-setup configure
   ```

   The configuration values that are used for this process map onto the values that you entered for the configuration steps when you first configured Oracle VM Agent for your primary control domain, as described in the *Configuring Oracle VM Agent for SPARC* section of the *Oracle VM Installation and Upgrade Guide*

When the Oracle VM Agent configuration has completed, the secondary domain is running and Oracle VM Agent is able to use it in the case that the primary domain becomes unavailable.

1.7.2.6 Automatically Creating and Setting Up a Secondary Domain

The `ovs-agent-secondary` command can be used to automatically create and setup a secondary domain. In particular, the command indicates whether the server is suitable for creating a secondary service domain, and which PCIe buses are available for the secondary service domain.

Note

A system reboot is not equivalent to powering off the system and restarting it. Furthermore, you should ensure that the system does not power off until you complete each step in the procedure to create a secondary service domain.

To create a secondary service domain, run the following command on the control domain:

```
# ovs-agent-secondary create
```

Important

The `ovs-agent-secondary` command is a helper script that is provided *as is*. This command might not work with some servers or configurations. If the command does not work, create the secondary service domain manually, as described in Section 1.7.2.3, “Creating a Secondary Service Domain”.

Listing PCIe Buses Present on the Server

The list of all PCIe buses present on the server is displayed, with information indicating whether or not they are available for creating a secondary service domain. Example output from the `ovs-agent-secondary` command, for this step, is displayed below:

```
Gathering information about the server...
The server has 2 PCIe buses.
----------------------------------------------------------------------
This is the list of PCIe buses present on the server, and whether
or not they are available for creating a secondary service domain
  Bus         Available   Reason
  ---         ---------   ------
  pci_0              no   Bus is assigned and used by the primary domain
  pci_1             yes   Bus is assigned to the primary domain but it is not used
Enter + or - to show or hide details about PCIe buses.
  +) Show devices in use
Or select one of the following options.
  0) Exit and do not create a secondary service domain
  1) Continue and select PCIe buses to create a secondary service domain
```

```
Choice (0-1): 1
```

Use this information to figure out which PCIe buses are available, and which buses you want to use for the secondary service domain. You can display more or less information about the PCIe buses by entering "+" or "-".

A PCIe bus is not available for creating a secondary service in the following cases:

- **The PCIe bus is assigned to a domain other than the primary domain.**

  If you want to use such a PCIe bus for the secondary service domain then you must first remove it from the domain it is currently assigned to.

- **The PCIe bus is assigned to the primary domain and devices on that bus are used by the primary domain.**

  If you want to use such a PCIe bus for the secondary service domain then you must reconfigure the primary domain so that it stops using devices from that bus.

> **Warning**
>
> When a PCIe bus is assigned to the primary domain, the tool may not always be able to figure out if devices from the bus are used by the primary domain. Furthermore, the tool only identifies common devices (such as network interfaces and disks) and the common usage of these devices (including link aggregation, IP configuration or ZFS pool). If you want to create a secondary domain with a PCIe bus that is currently assigned to the primary domain, make sure that this bus is effectively not used by the primary domain at all.

Selecting PCIe Buses for the Secondary Service Domain

The next step provided by the `ovs-agent-secondary` command allows you to actually select the PCIe buses that are to be used for the secondary service domain. Typically, this step may appear as follows:

```
The following PCIe buses can be selected for creating a secondary
service domain.
  Bus        Selected   Slot                 Devices Count
  ---        --------   ----                 -------------
  pci_1            no
                        /SYS/MB/PCIE5
                        /SYS/MB/PCIE6
                        /SYS/MB/PCIE7        ETH(2)
                        /SYS/MB/PCIE8        FC(2)
                        /SYS/MB/SASHBA1      DSK(2)
                        /SYS/MB/NET2         ETH(2)
Enter + or - to show or hide details about PCIe buses.
  +) Show devices
  -) Hide PCIe slots
Or enter the name of one or more buses that you want to add to the
selection of PCIe buses to create a secondary service domain.
Or select one of the following option.
  0) Exit and do not create a secondary service domain
  1) Add all PCIe buses to the selection
  2) Remove all PCIe buses from the selection
Choice (0-2): pci_1
adding bus pci_1 to selection
```

Note that in addition to the menu options, which allow you to add all available PCIe buses to the secondary service domain, you can also manually specify a space separated list of PCIe buses by bus name to individually add particular buses to the secondary service domain.

As soon as at least one PCIe bus is marked as selected, the menu options change to allow you to create the secondary service domain with the selected PCIe buses:

```
The following PCIe buses can be selected for creating a secondary
service domain.
  Bus        Selected   Slot                 Devices Count
  ---        --------   ----                 -------------
  pci_1           yes
                        /SYS/MB/PCIE5
                        /SYS/MB/PCIE6
                        /SYS/MB/PCIE7        ETH(2)
                        /SYS/MB/PCIE8        FC(2)
                        /SYS/MB/SASHBA1      DSK(2)
                        /SYS/MB/NET2         ETH(2)
Enter + or - to show or hide details about PCIe buses.
  +) Show devices
  -) Hide PCIe slots
Or enter the name of one or more buses that you want to add to the
selection of PCIe buses to create a secondary service domain.
Or select one of the following option.
  0) Exit and do not create a secondary service domain
  1) Add all PCIe buses to the selection
  2) Remove all PCIe buses from the selection
  3) Create a secondary services domain with the selected buses

Choice (0-3): 3
```

Confirming the Selection of PCIe Buses for the Secondary Service Domain

A final confirmation screen displays the buses selected for the secondary service domain, before you can proceed to create the secondary service domain. This confirmation screen looks as follows:

```
You have selected the following buses and devices for the secondary
domain.
  Bus        Current Domain       Slot                 Devices Count
  ---        --------------       ----                 -------------
  pci_1      primary
                                  /SYS/MB/PCIE5
                                  /SYS/MB/PCIE6
                                  /SYS/MB/PCIE7        ETH(2)
                                  /SYS/MB/PCIE8        FC(2)
                                  /SYS/MB/SASHBA1      DSK(2)
                                  /SYS/MB/NET2         ETH(2)
Verify that the selection is correct.
  0) Exit and do not create a secondary service domain
  1) The selection is correct, create a secondary domain with pci_1
  2) Go back to selection menu and change the selection
Choice (0-2): 1
```

Creating the Secondary Service Domain

After the selection of PCIe buses for the secondary service domain has been confirmed, the secondary domain is created and instructions for configuring the secondary service domain are displayed. The output from the tool looks similar to the following:

```
ldm add-domain secondary
ldm set-core 1 secondary
ldm set-memory 8G secondary
ldm add-vds secondary-vds0 secondary
ldm add-io pci_1 secondary
ldm start-reconf primary
ldm remove-io pci_1 primary

----------------------------------------------------------------------
```

```
The secondary service domain has been created. Next, you need to
install Solaris on that domain. Then you can configure the Oracle
VM Agent to run with the secondary domain.

Once the secondary service domain is up and running with Solaris,
run the following command to configure the Oracle VM Agent to run
with the secondary domain:

  # ovs-agent-secondary configure
```

If a reboot is required to complete the creation of the secondary service domain then a corresponding menu is displayed, otherwise the tool terminates and the creation of secondary service domain is already finished. The following menu is displayed if a reboot is required:

```
To complete the configuration of the Oracle VM Agent, the
system has to be rebooted.

Do you want to reboot the system now?
  1) Yes, reboot the system now
  2) No, I will reboot the system later

Choice (1-2): 1

Server Reboot

!!! WARNING !!!
You are not connected to the system console. Rebooting
the server will close this connection with the server.

!!! WARNING !!!
Are you sure that you want to continue?
  1) Yes, continue and reboot the system now
  2) No, cancel the reboot, I will reboot the system later

Choice (1-2): 1

Rebooting the system...
```

Installing the Service Domain

When you have finished creating the new service domain, you need to install it. Complete the instructions in Section 1.7.2.4, “Installing the Secondary Service Domain”.

Configuring the Oracle VM Agent for the Secondary Domain

Once the secondary service domain is correctly installed, you must configure the Oracle VM Agent to use it by running the `ovs-agent-secondary` command on the control domain, as follows:

```
# ovs-agent-secondary configure
```

Checking the Installation of the Secondary Service Domain

The first step in the configuration process requires you to confirm that the secondary domain is installed and running. This step is displayed as follows:

```
The secondary service domain exists and is active. It should be up
and running Solaris 11.3.

Confirm that the secondary service domain is up and running Solaris 11.3

  1) Yes, the secondary service domain is up and running Solaris 11.3.
  2) No, the secondary service domain is not running Solaris 11.3
```

```
Choice (1-2): 1
```

Removing Virtual Switches

The configuration process notifies you if virtual switches are defined.

```
The secondary domain can only be configured when no virtual
switches are defined. Remove any virtual switch, and restart
the configuration.

The following virtual switches are defined: 0a010000
```

You must remove any virtual switches defined in the secondary service domain before you can configure it, as in the following example:

```
# ldm list-services
VCC
    NAME             LDOM             PORT-RANGE
    primary-vcc0     primary          5000-5127
VSW
    NAME             LDOM             MAC               NET-DEV   ID   DEVICE
    0a010000         primary          00:14:4f:fb:53:0e net0      0    switch@0

    LINKPROP   DEFAULT-VLAN-ID   PVID VID   MTU   MODE   INTER-VNET-LINK
               1                 1          1500         on
VDS
    NAME             LDOM         VOLUME     OPTIONS      MPGROUP    DEVICE
    primary-vds0     primary
VDS
    NAME             LDOM         VOLUME     OPTIONS      MPGROUP    DEVICE
    secondary-vds0   secondary

# ldm remove-vsw 0a010000
```

Restart the configuration of the secondary service domain after you remove the virtual switch.

```
# ovs-agent-secondary configure
The secondary service domain exists and is active. It should be up
and running Solaris 11.3.
Confirm that the secondary service domain is up and running Solaris 11.3
  1) Yes, the secondary service domain is up and running Solaris 11.3.
  2) No, the secondary service domain is not running Solaris 11.3
Choice (1-2): 1
```

Mapping Network Interfaces Between the Primary and the Secondary Domain

Each network link in the primary domain should have a corresponding network link in the secondary domain connected to the same physical network. By default, a network link in the primary domain is associated with the network link with the same name in the secondary domain. If a network link in the primary domain should be associated with a network link in the secondary domain with a different name, then you need to define a network link mapping. This is achieved in the next step of the configuration process, which is displayed as follows:

```
Each network link in the primary domain should have a corresponding
network link in the secondary domain connected to the same physical
network. By default, a network link in the primary domain will be
associated with the network link with the same name in the secondary
domain.

Network links in the primary domain and corresponding link in the
secondary domain:
    Primary    Secondary
    -------    ---------
    net0       net0
    net1       net1
    net4       net4
    net5       net5
    net6       net6
    net7       net7

If a network link in the primary domain should be associated with
a network link in the secondary domain with a different name, then
you need to define a network link mapping.

Do you need to define a network link mapping?

  1) Yes, I need to map a network link in the primary domain to
     a network link in the secondary domain with a different name.
  2) No, each network link in the primary domain has a corresponding
     network link in the secondary domain with the same name.

Choice (1-2): 1
```

Ideally, you should be able to select option 2 here to continue. However, it is possible that network link names may not correspond correctly. In this case, you should select option 1 and redefine the mapping as follows:

```
Enter the mapping for net0 [net0]:
Enter the mapping for net1 [net1]:
Enter the mapping for net4 [net4]: net2
Enter the mapping for net5 [net5]: net3
Enter the mapping for net6 [net6]:
Enter the mapping for net7 [net7]:

Network links in the primary domain and corresponding link in the
secondary domain:

    Primary    Secondary
    -------    ---------
    net0       net0
    net1       net1
    net4       net2
    net5       net3
    net6       net6
    net7       net7

Is the mapping correct?

  1) Yes, the mapping is correct.
  2) No, the mapping is not correct, redo the mapping.

Choice (1-2): 1
```

Note that you are prompted for the mapping for each network link in the primary domain. If you enter a blank line, the existing default mapping is used. If you need to change a mapping, you must specify the network link name in the secondary domain that is connected to the same physical network as the network link listed in the primary domain.

When you have finished redefining the mappings, you should select option 1 to continue to the next step in the configuration process.

Mapping Fibre Channel Disk Controllers Between the Primary and the Secondary Domain

Each Fibre Channel (FC) disk accessible from the primary domain should also be accessible from the secondary domain. By default, a FC disk is accessed using the same device path in the primary domain and in the secondary domain. In particular, each disk is accessed using the same disk controller name. If a disk controller in the primary domain should be associated with a disk controller in the secondary domain with a different name, then you must define a disk controller mapping.

It is recommended that Solaris I/O multipathing is enabled in the primary and in the secondary domain on all multipath-capable controller ports, in particular on all FC ports. In this case, all FC disks appear under a single disk controller (usually c0), and disk controller mapping is usually not needed.

The following screen is displayed for this step in the configuration process:

```
Each Fibre Channel (FC) disk accessible from the primary domain
domain should also be accessible from the secondary domain. By
default, a FC disk will be access using the same device path in
the primary domain and in the secondary domain. In particular,
each disk will be accessed using the same disk controller name.

FC disk controllers in the primary domain and corresponding
controller in the secondary domain:

    Primary    Secondary
    -------    ---------
    c0         c0

If a disk controller in the primary domain should be associated with
a disk controller in the secondary domain with a different name, then
you need to define a disk controller mapping.

Do you need to define a disk controller mapping?

  1) Yes, I need to map a disk controller in the primary domain to
     a disk controller in the secondary domain with a different name.
  2) No, each disk controller in the primary domain has a corresponding
     disk controller in the secondary domain with the same name.

Choice (1-2): 1
```

Ideally, you should be able to select option `2` to continue. However, it is possible that disk controller names might not correspond correctly. In this case, you should select option `1` and redefine the mapping as follows:

```
Enter the mapping for c0 [c0]: c1

FC disk controllers in the primary domain and corresponding
controller in the secondary domain:

    Primary    Secondary
    -------    ---------
    c0         c1

Is the mapping correct?

  1) Yes, the mapping is correct.
  2) No, the mapping is not correct, redo the mapping.

Choice (1-2): 1
```

Note that you are prompted for the mapping for each FC disk controller in the primary domain. If you enter a blank line, the existing default mapping is used. If you need to change a mapping, you must specify the FC disk controller name in the secondary domain that is connected to the same FC disk listed in the primary domain.

When you have finished redefining the mappings, you should select option `1` to continue to the next step in the configuration process.

Saving the Oracle VM Agent Configuration for the Secondary Service Domain

The Oracle VM Agent uses a configuration file to access and configure itself for resources in the secondary service domain. In this step of the configuration process, the configuration file is created and saved to disk within the primary control domain:

```
Creating configuration file
Saving configuration ovm-shadow on the service processor

The secondary service domain is configured. Continuing with
the configuration of the Oracle VM Agent.

This command can not be run while the ovs-agent is online.

Do you want to disable the ovs-agent service?
  1) Yes, disable the ovs-agent service
  2) No, exit the ovs-agent-setup tool

Choice (1-2): 1
```

Reconfiguring the Oracle VM Agent

Finally, the Oracle VM Agent is automatically reconfigured to use the secondary service domain and the Oracle VM Agent is enabled:

```
Network Configuration
Network Configuration OK
Storage Configuration
Storage Configuration OK
OVS Agent Configuration
OVS Agent Configuration OK
Cluster Configuration
Cluster Configuration OK
LDoms Manager Configuration
LDoms Manager Configuration OK
Virtual I/O Services Configuration
Virtual I/O Services Configuration OK
LDoms Configuration
LDoms Configuration OK

Enabling Oracle VM Agent Services
```

The configuration values that are used for this process map onto the values that you entered for the configuration steps when you first configured Oracle VM Agent for your primary control domain, as described in the *Oracle VM Installation and Upgrade Guide* .

When the process is complete, the Oracle VM Agent is enabled and your environment is configured to use both a primary and a secondary service domain.

Chapter 2 Configuring Oracle VM Manager After Installation

After you successfully install *Oracle VM Manager*, you can perform several configuration tasks to customize your environment. These configuration tasks include setting the session timeout period for the Oracle VM Manager Web Interface, and configuring SSL.

2.1 Configuring Oracle VM Manager Web Interface Session Timeout

You can change the amount of time that an Oracle VM Manager session can remain inactive before a timeout occurs through the Oracle WebLogic Server Administration Console.

To configure Oracle VM Manager Web Interface session timeout, do the following:

1. Open the Oracle WebLogic Server Administration Console at:

   `https://hostname:7002/console`

   Where *`hostname`* is the Oracle VM Manager hostname or IP address.

2. Log in as the `weblogic` user.

3. Locate the **Domain Structure** pane on the left and then click **Deployments**.

4. Click **Next** in the list of deployed applications until you locate the **ovm_console** application.

5. Click "**+**" to expand the **ovm_console** application and then click **ovm/console**.

   The settings for ovm_console are displayed.

6. Click the **Configuration** tab and then click **Lock and Edit** in the **Change Center** pane to modify the settings.

7. From the **Configuration** tab, locate the **General** subtab and then edit the value of the **Session Timeout** field. The default setting is half an hour (1800 seconds).

   When you are finished with your changes, click **Save**.

   > **Note**
   >
   > If you receive a permissions related error, you might need to change the permissions or ownership on the file located at `/u01/app/oracle/ovm-manager-3/weblogic/deploy/ovm_console/plan/plan.xml`. Use the following command:
   >
   > ```
   > chown oracle:dba /u01/app/oracle/ovm-manager-3/weblogic/deploy/ovm_console/plan/plan.x
   > ```
   >
   > After you change permissions on the file, you must edit and save the value of the **Session Timeout** field again.

8. Click **Deployments** in the **Domain Structure** pane on the left to return to the list of deployed applications.

9. Locate and select the **ovm_console** check box and then click **Update** to redeploy the application.

10. Change the source and deployment plan paths as appropriate and then click **Finish**.

11. To activate the changes, click **Activate Changes** in the **Change Center**.

Important

Due to the nature of some pages served within Oracle VM Manager, the client browser auto-refreshes regularly to poll for changes. This is particularly apparent on the Health page. Since the client is constantly refreshing, UI timeout may not behave as expected. Therefore, a configuration parameter for ADF has been set to timeout automatic polling after a default period of 20 minutes where there has been no mouse or keyboard interaction within Oracle VM Manager. This means that for these pages, the UI timeout value only becomes effective after the polling timeout has been effected.

Changes to the polling timeout are not directly configurable. If you require this facility to be modified contact Oracle Support.

2.2 Setting Up SSL

By default, Oracle VM Manager provides its own SSL certificates stored in a custom keystore. The certificates that are provided are signed using an internal Certificate Authority (CA). Oracle VM uses SSL certificates:

- For the authentication of Oracle VM Manager to each Oracle VM Server that it has discovered and for the encryption of communications between Oracle VM Manager and the Oracle VM Agent running on each Oracle VM Server.
- For the authentication and encryption of some tools that make use of the Oracle VM Manager web-services API.
- For the encryption of communications between a web-browser and the Oracle VM Manager web-based user interface .

Certificates are generated automatically during the installation of Oracle VM Manager. To avoid SSL validation issues in client web-browsers, you can obtain the internal CA certificate used by Oracle VM Manager and install it into each web-browser that is used to access the Oracle VM Manager web user interface. See Section 2.2.4, “Exporting the CA Certificate”.

Alternatively, if you already have an SSL certificate that is signed by an external CA, you can change the SSL certificate that is used for the encryption of communications between the web-browser and the Oracle VM Manager web-based user interface. See Section 2.2.6, “Changing the Default SSL Certificate”.

Finally, if you need to generate a new SSL key that is signed by the internal CA, you can follow the instructions provided in Section 2.2.5, “Generating a New SSL Key”.

Important

Changing the Oracle VM Manager CA certificate impacts authentication between Oracle VM Manager and various internal components. Changing the CA certificate also impacts authentication between Oracle VM Manager and each Oracle VM Server instance and other external applications such as web browsers.

If you plan to change the CA certificate, you should do so before you begin any other Oracle VM Manager configuration to avoid authentication and communication issues between components.

Oracle VM Manager uses the following 2048-bit keystores instead of the default Oracle WebLogic Server keystore:

- /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmca.jks
- /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmclient.jks
- /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks
- /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmtrust.jks

The passwords for these keystores are randomized at installation. If you need to update a keystore, such as the CA keystore, to add mutually trusted CAs to the keystore, you may need to change the keystore password using the Oracle VM Key Tool. For instructions on changing the keystore password, see Section 2.2.7, “Changing the Keystore Password”.

2.2.1 Oracle VM Key Tool

Oracle VM Manager includes a key management utility to help manage SSL certificates. You use the Oracle VM Key Tool in conjunction with the Java keytool in the Java Development Kit (JDK) that is installed on the Oracle VM Manager host. These utilities are located on the Oracle VM Manager host as follows:

- Oracle VM Key Tool: /u01/app/oracle/ovm-manager-3/ovm_upgrade/bin/ovmkeytool.sh
- Java keytool: /u01/app/oracle/java/bin/keytool

> **Important**
>
> Using key management utilities incorrectly can cause authentication issues between Oracle VM Manager, internal components, and external systems and applications. In some cases, authentication issues can result in complete loss of service. You should carefully plan any changes before using a key management utility and consider the impact those changes have on your environment.

Syntax

ovmkeytool.sh [--help] [--overwrite] [--quiet] [--verbose] [--propertyFile *filename*] [-D *property=value*] [--noWebLogic] { [{ show } | { check } | { setup } | { setupWebLogic } | { gencakey } | { setcakey } | { gensslkey } | { setsslkey } | { changepass } | { exportca }] }

Options

The following table shows the available options for this tool.

Option	Description
--help	Display the ovmkeytool.sh command parameters and options.
--overwrite	Allow existing keystores to be overwritten if user interaction is disabled.
--quiet	Run with no user interaction using property values exclusively.
--verbose	Output extra information while running.
--propertyFile *filename*	The specified file can be used to provide properties to the tool.
-D *property=value*	Sets a property to a given value.
--noWebLogic	Do not attempt to configure Oracle WebLogic Server or verify Oracle WebLogic Server settings.

Commands

The following table shows the available commands for this tool. Only one command can be run at a time.

Option	Description
`show`	Displays SSL configuration details such as the CA and SSL keystore files, certificate key aliases, and certificate details.
`check`	Validates the current SSL configuration and provides information if any errors exist.
`setup`	Sets up all of the keystore files and configures Oracle WebLogic Server.
`setupWebLogic`	Configures existing keystore settings in Oracle WebLogic Server.
`gencakey`	Generates a new certificate authority (CA) key and puts the key into the trust store. You should not run this command if you have already configured Oracle VM Manager
`setcakey`	Sets the CA key to use an existing key from an existing keystore file. You should not run this command if you have already configured Oracle VM Manager
`gensslkey`	Generates a new SSL key.
`setsslkey`	Sets the SSL key to use an existing key from an existing keystore file.
`changepass`	Allows the passwords for existing keystore files and keys to be configured or changed.
`exportca`	Exports the CA certificate in PEM format.

Command Prompts

Depending on the command you run, the Oracle VM Key Tool prompts you for the following information:

- Oracle WebLogic Server `Middleware` directory

  You can set the `MW_HOME` environment variable to point to the location of the Oracle WebLogic Server `Middleware` directory. Otherwise you must specify the default directory each time you run the Oracle VM key tool. The default directory is `/u01/app/oracle/Middleware`.

  Run the following command to set the `MW_HOME` environment variable:

  ```
  # export MW_HOME=/u01/app/oracle/Middleware
  ```

- Oracle WebLogic Server domain directory

  The default directory is `/u01/app/oracle/ovm-manager-3/domains/ovm_domain`.

- Oracle WebLogic Server name

  The default server name is `AdminServer`.

- Oracle WebLogic Server credentials

Use the default `weblogic` username and the one-time password that you set during Oracle VM Manager installation.

2.2.2 Showing the Certificate Configuration

Use the `show` command to view details about the current Certificate Authority (CA) and SSL configuration.

The following is an example of the `show` command:

```
# ./ovmkeytool.sh show

time_stamp oracle.security.jps.JpsStartup start
INFO: Jps initializing.
time_stamp oracle.security.jps.JpsStartup start
INFO: Jps started.
CA Keystore File: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmca.jks
CA Key Alias: ca
Certificate details:
  Algorithm: SHA256withRSA
  Subject: CN=OVM CA 0004fb00000100007c08b684bd203388, OU=Oracle VM Manager,
    O=Oracle Corporation, L=Redwood City, ST=California, C=US
  Issuer: CN=OVM CA 0004fb00000100007c08b684bd203388, OU=Oracle VM Manager,
    O=Oracle Corporation, L=Redwood City, ST=California, C=US
  Serial number: 83605355556470155808720780398068469367331216940З
  Valid from day mm dd hh:mm:ss CET yyyy to day mm dd hh:mm:ss CET yyyy
  SHA256 Fingerprint: b4:6b:00:cd:d3:e1:69:d6:f2:10:80:cf:a8:ef:89:c9:b3
  This is a valid Certificate to be used as a CA.
Full Certificate:
-----BEGIN CERTIFICATE-----
MIID/DCCAuSgAwIBAgIVAJJx8CLgw6WudhsYXsY70zxLaq27MA0GCSqGSIb3DQEB
CwUAMIGkMQswCQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEVMBMGA1UE
BxMMUmVkd29vZCBDaXR5MRswGQYDVQQKExJPcmFjbGUgQ29ycG9yYXRpb24xGjAY
BgNVBAsTEU9yYWNsZSBWTSBNYW5hZ2VyMTAwLgYDVQQDEydPVk0gQ0EgMDAwNGZi
MDAwMDAxMDAwMDdjMDhiNjg0YmQyMDMzODgwHhcNMTUwMzIyMTYxMTI1WhcNMjUw
MzIzMTYxMTI1WjCBpDELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWEx
FTATBgNVBAcTDFJlZHdvb2QgQ2l0eTEbMBkGA1UEChMST3JhY2xlIENvcnBvcmF0
aW9uMRowGAYDVQQLExFPcmFjbGUgVk0gTWFuYWdlcjEwMC4GA1UEAxMnT1ZNIENB
IDAwMDRmYjAwMDAwMTAwMDA3YzA4YjY4NGJkMjAzMzg4MIIBIjANBgkqhkiG9w0B
AQEFAAOCAQ8AMIIBCgKCAQEAprkZ3RYvS433ZOx+3RKK7iK7E52znhNpLPzM6b9s
O0fIdCiTBB16h6SU+GQlMMpQbqDh5V9OgWGd7BqCEnKhCU0O3L+xY45sXWGQ0S9R
DvQMH/68VgDwoSsI6BFL5gJHspWWr9wdqkpVcTpau9IN9nDGD38XnTd0KOtVvt+d
32lK3hBzQiXf/W2vX6vNA/RFMlfFBncnYIO4POvtQDsVSDzbfPq4CPqAxn/io1Gk
lycRrVbzemsrWuvusFCOpUkGpmaqwXneg/ozfN8ObUr+bh/PKhLniOo6gJsY2Y9l
ZjD6XiSUEd/Xb4s89SO6yHsNr65RC+wCCHpWjArr/3oVfQIDAQABoyMwITAPBgNV
HRMBAf8EBTADAQH/MA4GA1UdDwEB/wQEAwIABTANBgkqhkiG9w0BAQsFAAOCAQEA
Iv4g3Djf3KFwWLdQ/Tw1vK4kmzGIxcd9SS1YQUPYnddYeU22dwkqgIn9ap8dVK1u
lmkYdYZ4BDte4Y+Lptxqhf149S3nX1lBKfpg4eLsfUIZ+DTnxcuTCiFp/UNKp4Xk
yn43GMpUtyz8D//QX7T3FOtKq786Rl4i522i9xnWizyEXjTsSZ1T0b0y8lK7a+C4
mFOC53Ah1Ihmjl+1Q/zrcf+iFFFInCFywXDrpslE1R8H3Luse4EO42+xhEbxGY6h
xdVkG3vVCYqExBX3XWFfkVPF78+6bmzZbKZzam+NT49dVJRole4mssyOWa1AdWmB
RXXs6j6MR1mcveQVRFhPjg==
-----END CERTIFICATE-----


SSL Keystore File: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks
SSL Key Alias: ovmm
Certificate details:
  Algorithm: SHA256withRSA
  Subject: CN=ovmm.virtlab.info, OU=Oracle VM Manager,
    O=Oracle Corporation, L=Redwood City, ST=California, C=US
  Issuer: CN=OVM CA 0004fb00000100007c08b684bd203388, OU=Oracle VM Manager,
    O=Oracle Corporation, L=Redwood City, ST=California, C=US
  Serial number: 942935368201955864664901535859030776122723582749
  Valid from day mm dd hh:mm:ss CET yyyy to day mm dd hh:mm:ss CET yyyy
  SHA256 Fingerprint: 83:16:23:e1:2e:f5:7e:ff:3a:d5:72:1b:0b:d9:80:5b:d3:d6:b3
  Subject Alternative Names:
    Hostnames:
      myserver.example.com
      myovmm
Full Certificate:
-----BEGIN CERTIFICATE-----
MIID6TCCAtGgAwIBAgIVAKUqryxHow/khR23pDPGttbIbMMdMA0GCSqGSIb3DQEB
CwUAMIGkMQswCQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEVMBMGA1UE
BxMMUmVkd29vZCBDaXR5MRswGQYDVQQKExJPcmFjbGUgQ29ycG9yYXRpb24xGjAY
BgNVBAsTEU9yYWNsZSBWTSBNYW5hZ2VyMTAwLgYDVQQDEydPVk0gQ0EgMDAwNGZi
MDAwMDAxMDAwMDdjMDhiNjg0YmQyMDMzODgwHhcNMTUwMzIyMTYxMTM5WhcNMjUw
MzIzMTYxMTM5WjCBjjELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWEx
FTATBgNVBAcTDFJlZHdvb2QgQ2l0eTEbMBkGA1UEChMST3JhY2xlIENvcnBvcmF0
aW9uMRowGAYDVQQLExFPcmFjbGUgVk0gTWFuYWdlcjEaMBgGA1UEAxMRb3ZtbS52
aXJ0bGFiLmluZm8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCZEiTS
iY6sgh/23myW2l2PyO02ajGohnRxeDYFHRcfmdw/5C9XZXKY7rEpTx1WdPlRTjG0
DFD1dFjCjhJyIJo4DemyulniDoAKJG7dUoiB1sLb0HwVLyjGr3xQ/TbDyw04nppc
mdCGzhS/7ivzm2haMSSxiAoDFVZbeL/CmJSeN59fJmuUvZW01be/6TUNZVoMoOy0
GZm4D6cGWVcXIOuJSjfXep1mzkLIr4zsBTJQLV5uzRDXWjUANPSoN/XeLeHhYLYY
hBuDkDUMYGt0MsGomgQ4jbWchEid5/zQ3Th6FIKZ9PHVsVJPaeYSjObjNEUKkcIz
360d17bUqzQPXMK3AgMBAAGjJjAkMCIGA1UdEQQbMBmCEW92bW0udmlydGxhYi5p
bmZvggRvdm1tMA0GCSqGSIb3DQEBCwUAA4IBAQAeQfaXBGqfoQFisguthG/yPY4G
CLhp+78qSItCdMYPRrfXpUeeIVwrE6GQvuVflXZk/PPBZQGdDR3n/+hDfD9lccv0
MHFS8akON471tiDoku8tjm8a/EMir2/fEHU4PbgH57qUU9bj3lqzDZVI880qmPEx
IvSHwZy0KbrtPf+KkqHn75O/JlN46J+8AgRwuB/6e5ch7wAL2hupO3WeZV7O/icB
FJieePjxvMV5oXqxkFMHuidvVyAKN0MJK26w2lOWwTJtEmnBJ6UF1btNRQdgujUL
anJoGhJsLHyoosIrXbj3M+SmezwV+2kAPLDd8C/aNnXzZC4m55cwEB/GphYd
-----END CERTIFICATE-----


SSL Trust Keystore File: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmtrust.jks
Trusted certificates:
  CN=OVM CA 0004fb00000100007c08b684bd203388, OU=Oracle VM Manager,
    O=Oracle Corporation, L=Redwood City, ST=California, C=US
CA certificiate found in SSL Trust-Store

Oracle MiddleWare Home (MW_HOME): [home/u01/app/oracle/Middleware]
WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
WebLogic server name: [AdminServer]
WebLogic username: [weblogic]
WebLogic password: [********]
WLST session logged at: /tmp/wlst-session178461015146984067.log

WebLogic SSL Keystore File: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks
WebLogic SSL Key Alias: ovmm
WebLogic SSL Trust Keystore File: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmtrust.jks
```

2.2.3 Validating the Certificate Configuration

Use the check command to verify the current CA and SSL configuration. If any issues exist with the configuration, the command displays information to help you resolve them.

The following is an example of the check command:

```
# ./ovmkeytool.sh check
time_stamp oracle.security.jps.JpsStartup start
INFO: Jps initializing.
time_stamp oracle.security.jps.JpsStartup start
INFO: Jps started.
 Oracle MiddleWare Home (MW_HOME): [/u01/app/oracle/Middleware]
 WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
 Oracle WebLogic Server name: [AdminServer]
 WebLogic username: [weblogic]
 WebLogic password: [********]
 WLST session logged at: /tmp/wlst-session178461015146984067.log
```

```
The Oracle VM Manager CA and SSL configuration appears to be valid.
```

2.2.4 Exporting the CA Certificate

Oracle VM Manager contains an internal CA that performs certificate-based authentication and signs the default SSL certificate. Use the `exportca` command to export the CA certificate in PEM format. You can then add it as a trusted CA in a browser or use as required.

The following is an example of the `exportca` command:

```
# ./ovmkeytool.sh exportca

time_stamp oracle.security.jps.JpsStartup start
INFO: Jps initializing.
time_stamp oracle.security.jps.JpsStartup start
INFO: Jps started.
----BEGIN CERTIFICATE-----
MIID+zCCAuOgAwIBAgIUamdPKrCAl4O1yD8QlywkYhmh0l8wDQYJKoZIhvcNAQEL
BQAwgaQxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlhMRUwEwYDVQQH
EwxSZWR3b29kIENpdHkxGzAZBgNVBAoTEk9yYWNsZSBDb3Jwb3JhdGlvbjEaMBgG
A1UECxMRT3JhY2xlIFZNIE1hbmFnZXIxMDAuBgNVBAMTJ09WTSBDQSAwMDA0ZmIw
MDAwMDEwMDAwN2RiZmM3M2UyYTFkNjY3ZTAeFw0xMzExMDYxNDU5MjBaFw0yMzEx
MDcxNDU5MjBaMIGkMQswCQYDVQQGEwJVUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEV
MBMGA1UEBxMMUmVkd29vZCBDaXR5MRswGQYDVQQKExJPcmFjbGUgQ29ycG9yYXRp
b24xGjAYBgNVBAsTEU9yYWNsZSBWTSBNYW5hZ2VyMTAwLgYDVQQDEydPVk0gQ0Eg
MDAwNGZiMDAwMDAxMDAwMDdkYmZjNzNlMmExZDY2N2UwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQCKEekWsegMBt6aAPLAq+riDX8TS6ssr6NNjdDNy0mQ
32NZRyoR8K85T0O0KoFJ9lkgJOH8Ll4Q4219S2xey0obnqMqt5byW/XhXjiDLgpF
ESg/p2IGic8MubElhOQI3V71SeIcMHGk2b6sdS12T583uZD+FxvzCZoSTod4l4Pw
KvmAWV0FJQHaeOlGxj2tUaAWyVGbw66IzXZM4WlmNFH/2SNdx7XK4lXtPD/QiMVB
7bXaP/wCTc1vQlXgP550idwRQi5ol2ly7IO2fbflfX5wdnkuJWFOKzJfnkclsMHo
DW1FX5FEj34dEd/97wXvfAfYXRtC1DIq91mrF4vxD3lzAgMBAAGjIzAhMA8GA1Ud
EwEB/wQFMAMBAf8wDgYDVR0PAQH/BAQDAgAFMA0GCSqGSIb3DQEBCwUAA4IBAQAe
JK82gdNA/7tftEAgON7GlzJ0BSgu/3e1Luali+xOt2FFGAvrDtTdLxJjEEWM0OU4
Bhoc/6kjQ71nFs9Q/xxP9qC3YQPXa447Q1i9RZql5g2S5aQBr18ZHqeXp6HannLo
iwLBfSpbACgAhZwpzo7ZS38yENir6u1LKAnFAP/6D55Jgx7/UnbHNcFTSXc2u4cI
N3MHJ+0p8umz4+HrqqhFChNYZF2XhmuPawgL8TmRB2FNlQUcbmH19Nwb4UeOxEuD
isAf90p/GlTdtwzbNbm6Mv3rPEK2GtIL5YcIwLyKYKZ07P5VW6tGuzJTMipN0cLo
ij8FtceX5tmLGxlGQoKN
-----END CERTIFICATE-----
```

2.2.5 Generating a New SSL Key

By default, Oracle VM Manager generates and signs an SSL certificate that is valid for ten years from the date of installation. If necessary, you can use the `genssl` command to generate a new SSL certificate that is signed by the Oracle VM Manager internal CA.

The following is an example of the `genssl` command:

```
# ./ovmkeytool.sh gensslkey
Path for SSL keystore: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks]
The hostname should be the fully qualified hostname of the system
(this is the hostname you'd use to access this system from outside the
local domain).  Depending on your machine setup the value below may not be
correct.
Fully qualified hostname: [myserver.example.com]
Validity in months: [120]
Key distinguished name is "CN=myserver.example.com, OU=Oracle VM Manager,
  O=Oracle Corporation, L=Redwood City, ST=California, C=US".  Use these values? [yes]
Alternate hostnames (separated by commas): [myserver.example.com,myserver]
You may either specify passwords or use random passwords.
If you choose to use a random password, only WebLogic, the Oracle VM Manager,
and this application will have access to the information stored in this
keystore.
```

```
Use random passwords? [yes]
Generating SSL key and certificate and persisting them to the keystore...
Updating keystore information in WebLogic
Oracle MiddleWare Home (MW_HOME): [/u01/app/oracle/Middleware]
WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
Oracle WebLogic Server name: [AdminServer]
WebLogic username: [weblogic]
WebLogic password: [********]
WLST session logged at: /tmp/wlst-session178461015146984067.log
```

Note that the command prompts you to provide the values for various steps through the procedure as the new SSL certificate is generated. Notably, you must enter a valid fully qualified domain name for the server. This value is used for the hostname in the SSL certificate and must match the hostname that is used to access the Oracle VM Manager web-based user interface.

2.2.6 Changing the Default SSL Certificate

You can change the default SSL certificate that Oracle VM Manager serves for authentication. For example, you can configure Oracle VM Manager to use an SSL certificate that has been signed by a third-party CA.

This section describes how to use the Java keytool and the Oracle VM Key Tool to change the default SSL certificate.

Note

You should modify the example commands for the Java keytool to suit your business needs. Refer to the appropriate Java keytool documentation for more information.

Creating a Keystore on Oracle VM Manager

If you do not already have a third-party CA certificate and SSL certificate, you can create a new keystore on Oracle VM Manager. The keystore you create contains one entry for a private key. After you create the keystore, you generate a certificate signing request (CSR) for that private key and submit the CSR to a third-party CA. The third-party CA then signs the CSR and returns a signed SSL certificate and a copy of the CA certificate. You then import the CA certificate and SSL certificate into the keystore and configure Oracle VM Manager to use it as the SSL keystore.

1. Create a new keystore.

   ```
   # keytool -genkeypair -alias alias -keyalg RSA -keysize key_size \
   -dname distinguished_name -keypass private_key_password \
   -storetype jks -keystore keystore.jks -storepass keystore_password
   ```

2. Generate a certificate signing request (CSR).

   ```
   # keytool -certreq -alias alias -file certreq.csr -keypass private_key_password \
    -storetype jks -keystore keystore.jks -storepass keystore_password
   ```

3. Submit the CSR file to the relevant third-party CA for signing.

4. Import the CA certificate into the keystore.

   ```
   # keytool -importcert -trustcacerts -noprompt -alias alias -file ca_cert_file \
     -storetype jks -keystore keystore.jks -storepass keystore_password
   ```

5. Import the SSL certificate into the keystore.

   ```
   # keytool -importcert -trustcacerts -noprompt -alias alias -file ssl_cert_file \
   ```

```
  -keypass private_key_password -storetype jks -keystore keystore.jks \
  -storepass keystore_password
```

6. Use the `setsslkey` command to configure Oracle VM Manager to use the new keystore.

   ```
   # ./ovmkeytool.sh setsslkey
   Path for SSL keystore: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks]
     /path/to/keystore.jks
   Keystore password:
   Alias of key to use as SSL key: alias
   Key password:
   Updating keystore information in WebLogic
   Oracle MiddleWare Home (MW_HOME): [/u01/app/oracle/Middleware]
   WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
   Oracle WebLogic Server name: [AdminServer]
   WebLogic username: [weblogic]
   WebLogic password: [********]
   WLST session logged at: /tmp/wlst-session5820685079094897641.log
   ```

7. As the root user, configure the client certificate login.

   ```
   # su -c "/u01/app/oracle/ovm-manager-3/bin/configure_client_cert_login.sh /path/to/cacert"
   ```

   Where `/path/to/cacert` is the absolute path to the CA certificate. You must provide the path to the CA certificate if you used a CA other than the default Oracle VM Manager CA to sign the SSL certificate.

Importing a Keystore into Oracle VM Manager

If you already have a CA certificate and SSL certificate, use the SSL certificate to create a keystore. You can then import that keystore into Oracle VM Manager and configure it as the SSL keystore.

1. Import the keystore into Oracle VM Manager.

   ```
   keytool -importkeystore -noprompt -srckeystore source_keystore \
     -srcstoretype source_format -srcstorepass source_keystore_password \
     -destkeystore destination_keystore.jks -deststoretype JKS \
     -deststorepass destination_keystore_password
   ```

2. Use the `setsslkey` command to configure Oracle VM Manager to use the new keystore.

   ```
   # ./ovmkeytool.sh setsslkey
   Path for SSL keystore: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmssl.jks]
     /path/to/keystore.jks
   Keystore password:
   Alias of key to use as SSL key: alias
   Key password:
   Updating keystore information in WebLogic
   Oracle MiddleWare Home (MW_HOME): [/u01/app/oracle/Middleware]
   WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
   Oracle WebLogic Server name: [AdminServer]
   WebLogic username: [weblogic]
   WebLogic password: [********]
   WLST session logged at: /tmp/wlst-session5820685079094897641.log
   ```

3. As the root user, configure the client certificate login.

   ```
   # su -c "/u01/app/oracle/ovm-manager-3/bin/configure_client_cert_login.sh /path/to/cacert"
   ```

   Where `/path/to/cacert` is the absolute path to the CA certificate. You must provide the path to the CA certificate if you used a CA other than the default Oracle VM Manager CA to sign the SSL certificate.

2.2.7 Changing the Keystore Password

In some scenarios, you may also want to configure Oracle WebLogic Server's SSL truststore to provide additional trusted CAs. To do this you may use the `changepass` command to change the truststore password, since the default password for the keystore is randomized and it would not be possible to modify the keystore without the correct password. Once you have reset the password, you can modify the keystore using the Java `keytool`, as required. It is imperative that the existing internal Oracle VM Manager CA certificate is not removed from the keystore.

An example of setting the keystore password and then accessing trust information using the Java keytool command is shown below:

```
# ./ovmkeytool.sh changepass
You may either specify passwords or use random passwords.
If you choose to use a random password, only WebLogic, the Oracle VM Manager,
and this application will have access to the information stored in this
keystore.
Use random passwords? [yes] no
Change CA Keystore and Key passwords? [yes] no

Change SSL Keystore and Key passwords? [yes] no

Change SSL Trustore password? [yes]
SSL Trust Keystore password:
Verify SSL Trust Keystore password:
Updating trust-store information in WebLogic
Oracle MiddleWare Home (MW_HOME): [/u01/app/oracle/Middleware]
WebLogic domain directory: [/u01/app/oracle/ovm-manager-3/domains/ovm_domain]
Oracle WebLogic Server name: [WLS1] AdminServer
WebLogic username: [weblogic]
WebLogic password: [********]
WLST session logged at: /tmp/wlst-session6297528751781822860.log
```

```
# keytool -list -keystore /u01/app/oracle/ovm-manager-3/domains/ovm_domain/security/ovmtrust.jks
Enter keystore password:

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 1 entry

ovmmgr_ca_key_entry, Nov 7, 2013, trustedCertEntry,
Certificate fingerprint (MD5): 65:31:9C:17:35:59:6C:A7:A3:93:C8:93:F0:A7:81:6A
```

Chapter 3 Administering Oracle VM Manager

Administering Oracle VM Manager involves creating, deleting, and working with user accounts, modifying database schema, rotating log files, and capturing diagnostic information for troubleshooting.

3.1 Oracle VM Manager Administrator Tool (ovm_admin)

The Oracle VM Manager Administrator Tool, which can be invoked on the command line using the `ovm_admin` command, is used to perform administrative actions specific to Oracle VM Manager. These actions allow you to manage users that have access to the Oracle VM Manager data store, and control log rotation for the `AdminServer.log` file. To perform any action using the Oracle VM Manager Administrator Tool, you must use the password that is configured for the *weblogic* user.

The Oracle VM Manager Administrator Tool provides you with the ability to perform various user management functions directly from the command line. By default, the Oracle VM Manager installation process only creates and configures a single Oracle VM Manager administrative user. While this is often sufficient for many customers, creating separate administrative user accounts may be useful for security and auditing purposes.

The Oracle VM Manager Administrator Tool is installed as part of the default Oracle VM Manager installation process. The full path to the Oracle VM Manager Administrator Tool is:

```
/u01/app/oracle/ovm-manager-3/bin/ovm_admin
```

Syntax

```
ovm_admin [ --help ] [ --createuser ] [ --deleteuser admin ] [ --listusers ] [ --
modifyuser ] [ --modifyds ] [ --listds ] [ --lockusers tries ] [ --unlockuser
admin ] [ --listconfig ] [ --rotatelogsdaily HH:MM ] [ --rotatelogsbysize KB ] [ --
updatemysqlroot ]
```

Options

The following table shows the available options for this command.

Option	Description
`--help`	Display the `ovm_admin` command parameters and options.
`--listconfig`	Displays Oracle VM Manager configuration details.
`--listusers`	List the Oracle VM Manager users. For an example of how to list users, see Section 3.1.1, “Listing Users”.
`--createuser`	Create new Oracle VM Manager admin user. For an example of how to create a user, see Section 3.1.2, “Creating Users”.
`--deleteuser`	Delete an Oracle VM Manager admin user. For an example of how to delete a user, see Section 3.1.3, “Deleting Users”.
`--modifyuser`	Modify an Oracle VM Manager user password.

Option	Description
	For an example of how to change a user's password, see Section 3.1.4, “Changing User Passwords”.
`--lockusers tries`	Set the maximum login tries before locking accounts. This setting is global. For an example of how to change account locking, see Section 3.1.5, “Configure Account Locking”.
`--unlockuser admin`	Unlock a user account. For an example of how to unlock a user account, see Section 3.1.6, “Unlocking User Accounts”.
`--listds`	List Oracle VM Manager data sources. For an example of how to list data sources, see Section 3.1.7, “Listing Data Sources”.
`--modifyds`	Modify an Oracle VM Manager database schema. Typically used if the password for the MySQL database has been changed directly within MySQL. For an example of how to modify database schema, see Section 3.1.8, “Modifying the Oracle VM Manager Database Schema”.
`--rotatelogsdaily HH:MM`	Rotate the Oracle VM Manager application logs daily (HH:MM). For examples of rotating log files, see Section 3.1.9, “Rotating Log Files”.
`--rotatelogsbysize KB`	Rotate the Oracle VM Manager application logs by size (KB). For examples of rotating log files, see Section 3.1.9, “Rotating Log Files”.
`--updatemysqlroot`	Change the password for the MySQL root user. The Oracle VM Manager Administrator Tool connects to the MySQL database as the root user. This option changes the password that the Oracle VM Manager Administrator Tool uses for the root user but does not change the password in the database itself. For this reason, you must first change the password with the Oracle VM Manager Administrator Tool and then manually change the password in the database. You should review the best practices and considerations for this option before you change the password, see Section 3.1.10, “Changing the Password for the MySQL Root User”.

3.1.1 Listing Users

Obtain a list of users that have access to Oracle VM Manager with the following command:

```
# ./ovm_admin --listusers
```

The tool prompts you for the Oracle WebLogic Server password and returns output similar to the following:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001
Please enter the password for weblogic:

Location changed to serverRuntime tree. This is a read-only tree with DomainMBean as the root.
For more help, use help('domainConfig')

weblogic, admin, ovmuser
```

Some users stored within Oracle WebLogic Server are critical to your Oracle VM Manager environment, such as the following:

- *OracleSystemUser:* Used by Oracle Web Services Manager (OWSM). OWSM is part of the standard Oracle Fusion Middleware (FMW) Infrastructure, that includes ADF.
- *weblogic:* The default Oracle WebLogic Server administrative user.

The default *admin* user account is also typically listed. Any other user accounts listed, such as the **`ovmuser`** account, have been added to the system after installation.

For more information about default user accounts, see Section 4.1, “Default Oracle VM Manager Users”.

3.1.2 Creating Users

Create new Oracle VM Manager users with the following command:

```
# ./ovm_admin --createuser
```

The tool returns the following output:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001

Please enter the password for weblogic:
```

At this point you must enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

```
Please enter the username: ovmuser
Please enter a new password for ovmuser, this password
must be at least 8 characters long and must contain at least one non-alphabetic character:
Please re-enter the password:
```

```
Location changed to serverRuntime tree. This is a read-only tree with DomainMBean as the root.
For more help, use help('domainConfig')

date_time [main] INFO  ovm.wlst.domainbuilder.Domain - Created a user named ovmuser
```

Note

The must conform to the password requirements suggested by the Oracle VM Manager Administrator Tool or the creation of the user fails in the final step.

3.1.3 Deleting Users

Delete Oracle VM Manager administrative users with the following command:

```
# ./ovm_admin --deleteuser ovmuser
```

You are prompted for the Oracle WebLogic Server password. This is the password for the Oracle WebLogic Server as it was set up during installation. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password. Typical output is presented below:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001
Please enter the password for weblogic:

Location changed to serverRuntime tree. This is a read-only tree with DomainMBean as the root.
For more help, use help('domainConfig')


date_time [main] INFO  ovm.wlst.domainbuilder.Domain - Deleted the user named ovmuser
```

Important

Some users stored within Oracle WebLogic Server are critical to your Oracle VM Manager environment. Do not attempt to delete either of the following users:

- *OracleSystemUser*
- *weblogic*

You should also keep the default *admin* user account so that there is always at least one administrative account that can access Oracle VM Manager.

3.1.4 Changing User Passwords

Change any Oracle VM Manager administrative user's password with the following command:

```
# ./ovm_admin --modifyuser
```

The tool returns the following output:

```
Oracle VM Manager Release version Admin tool
```

```
/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands
date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001

Please enter the password for weblogic:
```

At this point you must enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

```
Please enter the username: ovmuser
Please enter the password for ovmuser:
```

Note

You must provide the user's current password to modify the user account.

If you need to reset an account due to a lost password, you should first delete the user account and then create a new account.

```
Please enter a new password for ovmuser, this password
must be at least 8 characters long and must contain at least one non-alphabetic character:
Please re-enter the password:
```

Note

The password must conform to the password requirements suggested by the Oracle VM Manager Administrator Tool or the creation of the user fails in the final step.

```
Location changed to serverRuntime tree. This is a read-only tree with DomainMBean as the root.
For more help, use help('domainConfig')

date_time [main] INFO  ovm.wlst.domainbuilder.Domain - Changed ovmuser's password
```

3.1.5 Configure Account Locking

To protect unauthorized access to Oracle VM Manager you can configure an account locking facility that is triggered after a number of failed attempts to log in.

Configure the account locking facility with the following command:

```
# ./ovm_admin --lockusers [3]
```

Note

Account locking is enabled by default according to the base Oracle WebLogic Server configuration. After you exceed the maximum number of invalid login attempts, the account is locked for 30 minutes before it is automatically unlocked again.

To change the lock period, you must edit the Oracle WebLogic Server configuration. For more information on configuring the Oracle WebLogic Server lockout parameters, refer to the appropriate Oracle WebLogic Server documentation.

Important

This is a global parameter that applies to all users. Setting this parameter on an instance of Oracle VM Manager that makes use of a single administrator account can result in this account being locked for 30 minutes before anybody can use it again. To recover from this is it is possible to unlock the account. See Section 3.1.6, “Unlocking User Accounts”.

You are prompted to enter the Oracle WebLogic Server password in order to apply this setting. Typical output from the command follows:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001
Please enter the password for weblogic:

Location changed to edit tree. This is a writable tree with
DomainMBean as the root. To make changes you will need to start
an edit session via startEdit().

For more help, use help('edit')


Starting an edit session ...
Started edit session, please be sure to save and activate your
changes once you are done.
date_time [main] INFO  ovm.wlst.domainbuilder.Domain - Set lockout threshold to 3 tries
Saving all your changes ...
Saved all your changes successfully.
Activating all your changes, this may take a while ...
The edit lock associated with this edit session is released
once the activation is completed.

The following non-dynamic attribute(s) have been changed on MBeans
that require server re-start:
MBean Changed : Security:Name=myrealmUserLockoutManager
Attributes changed : LockoutThreshold

Activation completed
```

You must restart Oracle VM Manager for the changes to the account locking facility to take effect, as follows:

```
# service ovmm restart
```

3.1.6 Unlocking User Accounts

When account locking is enabled (see Section 3.1.5, “Configure Account Locking”), it is possible for Oracle VM Manager user accounts to become locked for up to 30 minutes if a user fails to authenticate after the number of attempts that has been configured for this facility. When a user's account has become locked and the user enters the correct username and password combination, an error appears when the user attempts to authenticate:

```
Unexpected error during login (javax.security.auth.login.LoginException),
please consult logs for details.
```

An investigation of the `AdminServer.log` reveals:

```
>BEA-090078< >User ovmuser in security realm myrealm
has had 3 invalid login attempts, locking account for 30 minutes.<
```

You can override the 30 minute lock on an account with the following command:

```
# ./ovm_admin --unlockuser ovmuser
```

You are prompted for the Oracle WebLogic Server account password in order to complete the operation.

3.1.7 Listing Data Sources

Use this command option to check data sources before using the `--modifyds` option or to validate the result of a `--modifyds` operation.

Obtain a list of data sources that Oracle VM Manager uses with the following command:

```
# ./ovm_admin --listds

Oracle VM Manager Release version Admin tool

//u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001
Please enter the password for weblogic:
```

At this point you must enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

The tool prompts you to enter the MySQL user that should be used to query the database and then provides output similar to the following:

```
Please enter the name of a MySQL user: [appfw, ovs] ovs

Listing Oracle VM Manager Data Source 'ovm-jpa-ds'...
DriverName                              com.mysql.jdbc.Driver
Url                                     jdbc:mysql://localhost:49500/ovs
DatabaseName                            ovs
Listing Oracle VM Manager Data Source 'ovm-jpa-ds' successfully

Listing Oracle VM Manager Data Source 'ovm-odof-ds'...
DriverName                              com.mysql.jdbc.Driver
Url                                     jdbc:mysql://localhost:49500/ovs
DatabaseName                            ovs
Listing Oracle VM Manager Data Source 'ovm-odof-ds' successfully
```

3.1.8 Modifying the Oracle VM Manager Database Schema

You can use the Oracle VM Manager Administrator Tool to handle database schema changes within MySQL. The most typical use case for this is where the password for the Oracle VM Manager database has been changed directly within MySQL, without using any of the tools provided with Oracle VM. An alternative use case would be where the Oracle VM Manager database has been renamed within MySQL.

The `--modifyds` option is used to update Oracle VM Manager for changes made directly to the MySQL database:

```
# ./ovm_admin --modifyds
```

The tool prompts you for the Oracle VM Manager database schema password and the Oracle WebLogic Server password, and returns output similar to the following:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001

Please enter the password for weblogic:
```

At this point you must enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

```
Please enter the name of a MySQL user: [appfw, ovs] ovs
Please enter the password for MySQL user ovs:
Please enter the new password for ovs user:
Please re-enter the password:
Location changed to edit tree. This is a writable tree with
DomainMBean as the root. To make changes you will need to start
an edit session via startEdit().

For more help, use help('edit')

Starting an edit session ...
Started edit session, please be sure to save and activate your
changes once you are done.

Saving all your changes ...
Saved all your changes successfully.
Activating all your changes, this may take a while ...

......
The following non-dynamic attribute(s) have been changed on MBeans
that require server re-start:
MBean Changed : com.bea:Name=ovm-odof-ds,
Type=weblogic.j2ee.descriptor.wl.JDBCDriverParamsBean,Parent=[ovm_domain]
/JDBCSystemResources[ovm-odof-ds],Path=JDBCResource[ovm-odof-ds]/JDBCDriverParams
Attributes changed : PasswordEncrypted

Activation completed
```

Note that there is a second database schema, usually named *appfw*, that is also used by Oracle VM Manager. If the password for this database has also been changed, then the same command must be run again, as follows:

```
Oracle VM Manager Release version Admin tool

/u01/app/oracle/ovm-manager-3/ovm_wlst
```

```
Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001
Please enter the password for weblogic:
```

At this point you must enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

```
Please enter the name of a MySQL user: [appfw, ovs] appfw
Please enter the password for MySQL user appfw:
Please enter the new password for appfw user:
Please re-enter the password:
Location changed to edit tree. This is a writable tree with
DomainMBean as the root. To make changes you will need to start
an edit session via startEdit().

For more help, use help('edit')

Starting an edit session ...
Started edit session, please be sure to save and activate your
changes once you are done.

Saving all your changes ...
Saved all your changes successfully.
Activating all your changes, this may take a while ...

......
The following non-dynamic attribute(s) have been changed on MBeans
that require server re-start:
MBean Changed : com.bea:Name=ovm-qrtz-ds,
Type=weblogic.j2ee.descriptor.wl.JDBCDriverParamsBean,Parent=[ovm_domain]
/JDBCSystemResources[ovm-qrtz-ds],Path=JDBCResource[ovm-qrtz-ds]/JDBCDriverParams
Attributes changed : PasswordEncrypted

Activation completed
```

When you have finished running this command, you must restart Oracle VM Manager as follows:

```
# service ovmm restart
# service ovmcli restart
```

3.1.9 Rotating Log Files

The Oracle VM Manager Administrator Tool allows you to control how and when log files are rotated. There are two options available:

- `--rotatelogsdaily`: Set the logs to be rotated on a daily basis at an allocated time.
- `--rotatelogsbysize`: Set the logs to be rotated when they reach a specified size.

In both cases, you are prompted for the Oracle WebLogic Server password to update the configuration.

Rotating Oracle VM Manager logs daily

To set the logs to rotate daily at an allocated time, run the Oracle VM Manager Administrator Tool as follows:

```
# ./ovm_admin --rotatelogsdaily [00:30]
```

The time provided is specified in the format `HH:MM`.

Typical output from the command follows:

```
Oracle VM Manager Release version Admin tool

Please enter the password for weblogic :

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

Connecting to Oracle WebLogic Server ...

Connected ...
Configure log rotation setting to rotate daily at [00:30] ...
Modified log rotation setting successfully ...
Exiting...
```

Rotating Oracle VM Manager logs by size

To set the logs to rotate when they reach a specified size, run the Oracle VM Manager Administrator Tool as follows:

```
# ./ovm_admin --rotatelogsbysize [1024]
```

The size provided is specified according to the number of kilobytes before rotation.

Typical output from the command follows:

```
Oracle VM Manager Release version Admin tool

Please enter the password for weblogic :

Initializing WebLogic Scripting Tool (WLST) ...

Welcome to Oracle WebLogic Server Administration Scripting Shell

Type help() for help on available commands

Connecting to Oracle WebLogic Server ...

Connected ...
Configure log rotation setting to rotate the logs based on size ([1024] KB) ...
Modified log rotation setting successfully ...
Exiting...
```

3.1.10 Changing the Password for the MySQL Root User

You can change the password for the MySQL root user that the Oracle VM Manager Administrator Tool uses to connect to the MySQL database instance.

⚠ **Important**

The Oracle VM Manager Administrator Tool connects to the MySQL server as the root user. This option changes the password that the Oracle VM Manager Administrator Tool uses for the root user but does not change the password in the database itself. For this reason, you must first change the password with the Oracle

> VM Manager Administrator Tool and then manually change the password in the database.

Change the password for the MySQL root user as follows:

1. Run the Oracle VM Manager Administrator Tool with the `--updatemysqlroot` option.

   ```
   # ./ovm_admin --updatemysqlroot
   ```

   The tool returns the following output:

   ```
   Oracle VM Manager Release version Admin tool

   Initializing WebLogic Scripting Tool (WLST) ...

   Welcome to Oracle WebLogic Server Administration Scripting Shell

   Type help() for help on available commands
   date_time [main] INFO  ovm.wlst.commands - Connecting using URL t3://localhost:7001

   Please enter the password for weblogic:
   ```

2. Enter the password for the Oracle WebLogic Server. If you have not changed the Oracle VM Manager admin user's password, this password is usually the same as your default Oracle VM Manager admin user's password.

   The Oracle VM Manager Administrator Tool prompts you with the following:

   ```
   Please enter the current password for MySQL user root:
   ```

3. Enter the current password for the MySQL root user.

   The Oracle VM Manager Administrator Tool prompts you with the following:

   ```
   Please enter the new password for MySQL user root:
   Please re-enter the password:
   ```

4. Enter the new password for the MySQL root user and then confirm the password.

   The command provides the following output:

   ```
   date_time [main] INFO  ovm.wlst.domainbuilder.Domain -
   Updated MySQL root password successfully in WebLogic!
   Please note that you must separately update the password in the database
   ```

5. Stop Oracle VM Manager.

   ```
   # /etc/init.d/ovmm stop
   ```

6. Manually change the password in the database so that it matches the password that you set with the Oracle VM Manager Administrator Tool, as follows:

   a. Connect to the MySQL server.

      ```
      # mysql -S /u01/app/oracle/mysql/data/mysqld.sock -u root -p
      ```

   b. When prompted, enter the previous password for the root user, not the new password that you set with the Oracle VM Manager Administrator Tool.

   c. Ensure you are using the MySQL database.

      ```
      $ mysql> use mysql;
      ```

d. Change the password for the root user.

   ```
   $ mysql> alter user 'root'@'localhost' identified by new_password;
   ```

e. Flush privileges.

   ```
   $ mysql> flush privileges;
   ```

f. Disconnect from the MySQL server.

   ```
   $ mysql> quit
   ```

7. Restart the MySQL service for Oracle VM Manager.

   ```
   # /etc/init.d/ovmm_mysql restart
   ```

8. Start Oracle VM Manager.

   ```
   # /etc/init.d/ovmm start
   ```

3.2 Working with the MySQL Instance

Oracle VM Manager uses an instance of MySQL Enterprise Edition for storing configuration and other data. Database files reside at `/u01/app/oracle/mysql/data`.

Starting, Stopping, and Checking Status of the MySQL Server

Note

Oracle VM Manager depends on a running instance of the MySQL server. MySQL should never be stopped, even for troubleshooting, nor for configuring MySQL server, while Oracle VM Manager is running. The Oracle VM Manager should be stopped before touching MySQL. Irrespective of Oracle Linux 6 or Oracle Linux 7, the `service` command should be used to start and stop the `ovmm_mysql` service.

To start, stop, restart and obtain the status of the MySQL server, you can use the `/etc/init.d/ovmm_mysql` init script as follows:

```
# /etc/init.d/ovmm_mysql restart
```

Alternatively, you can use the `service` command as follows:

```
# service ovmm_mysql start
```

MySQL Configuration and Event Logs

Configuration for the Oracle VM Manager MySQL server is contained in: `/u01/app/oracle/mysql/data/my.cnf`.

Warning

Editing the configuration file might break your Oracle VM Manager installation. Do not edit the configuration file unless an Oracle Support representative instructs you to do so.

MySQL server events are logged in: `/u01/app/oracle/mysql/data/mysqld.err`.

Chapter 4 Managing Oracle VM Manager Authentication

Managing Oracle VM Manager authentication includes changing the password for the administration user, restricting user authentication to specific groups, and configuring LDAP and Active Directory authentication providers.

4.1 Default Oracle VM Manager Users

During installation, several user accounts are created in Oracle WebLogic Server. These users allow you to log in to Oracle VM Manager or to perform various administration tasks. Some user accounts are created to enable internal functions within Oracle VM Manager.

User	Description
`admin`	Oracle VM Manager user The password for this user is specified during installation of Oracle VM Manager. To change the password, use the Oracle VM Manager Administrator Tool.
`weblogic`	Oracle WebLogic Server administration user The password for this user is specified during installation of Oracle VM Manager. To change the password, use the Oracle VM Manager Administrator Tool.
`OracleSystemUser`	Internal user account that is part of the Oracle Fusion Middleware (FMW) infrastructure. The password for this user is specified during installation of Oracle VM Manager. To change the password, use the Oracle VM Manager Administrator Tool.
`ovs`	Internal user account that connects to the Oracle MySQL database user instance for Oracle VM Manager. The password for this user is specified during installation of Oracle VM Manager. To change the password, use the Oracle VM Manager Administrator Tool.
`appframework`	Internal user account that establishes connection between the Oracle VM Manager Web Interface and the Oracle VM Web Services API. This user is created with a randomly generated 128 character password that consists of mixed case letters, digits, and special characters. The password is used only once to register an SSL client certificate that the Oracle VM Manager Web Interface uses to connect to the Oracle VM Web Services API. To change the password for this user account, use the Oracle WebLogic Server Administration Console. **Note** • During an upgrade of Oracle VM Manager this user is replaced with a new account with a new randomly generated password.

User	Description
	• You must not delete this user account. Deleting the user account breaks the internal connection between the Oracle VM Manager Web Interface and the Oracle VM Web Services API and requires you to re-install Oracle VM Manager.

Related Information.

- For details about the Oracle VM Manager Administrator Tool, see Section 3.1, “Oracle VM Manager Administrator Tool (ovm_admin)”.
- For an example of how to change passwords with the Oracle VM Manager Administrator Tool, see Section 3.1.4, “Changing User Passwords”.
- For instructions on changing the Oracle VM Manager user password, see Section 3.1.4, “Changing User Passwords”.
- For details about changing user passwords in the Oracle WebLogic Server Administration Console, navigate to the *Modify users* topic in the Oracle WebLogic Server online help.

4.2 Changing the Oracle VM Manager User Password

The Oracle VM Manager user lets you log in to the Oracle VM Manager Web Interface. The default username is `admin`. You set the password for this user when you install Oracle VM Manager. By default, the Oracle VM Manager user has the same password as the `OracleSystemUser` and the `weblogic` users. To secure your environment, you should change the password for the Oracle VM Manager user if you intend to share the Oracle VM Manager user credentials.

To change the password for the Oracle VM Manager user, do the following:

1. Start an ssh session to the Oracle VM Manager *host computer* as the *root* user.
2. Change to the following directory: `/u01/app/oracle/ovm-manager-3/bin`
3. Run the following command: `# ./ovm_admin --modifyuser`
4. Follow the prompts to change the user password.

This procedure involves using the Oracle VM Manager Administrator Tool to modify the user password. Refer to Section 3.1, “Oracle VM Manager Administrator Tool (ovm_admin)” for more information about the Administrator Tool. For an example of changing user passwords, see Section 3.1.4, “Changing User Passwords”.

4.3 Restricting User Authentication to Oracle WebLogic Server Groups

Configure Oracle VM Manager to restrict authentication to users in specific Oracle WebLogic Server groups, such as administrative groups, as follows:

1. Start an ssh session to the Oracle VM Manager *host computer*.
2. Open the following file for editing: `/etc/sysconfig/ovmm`.

3. Specify the Oracle WebLogic Server user group that can authenticate to Oracle VM Manager as the value for the `AUTHORIZED_GROUPS` entry.

   Enclose the value in double quotes and use a comma to separate multiple values, for example:

   ```
   AUTHORIZED_GROUPS="group1,group2,group3"
   ```

4. Save and close `/etc/sysconfig/ovmm`.

5. Restart Oracle VM Manager to apply the changes.

Only users who belong to the groups that you specify can authenticate to Oracle VM Manager. If the `AUTHORIZED_GROUPS` entry does not exist, or has no value, then all Oracle WebLogic Server users can authenticate to Oracle VM Manager.

For more information about working with users and groups, navigate to the *Manage users and groups* topic in the Oracle WebLogic Server online help.

4.4 Enabling LDAP and Active Directory Authentication

Oracle VM Manager is an application that runs on Oracle WebLogic Server. For this reason, Oracle VM Manager supports any authentication providers that Oracle WebLogic Server supports.

To configure Oracle VM Manager to authenticate against an LDAP or Active Directory service, you must add the directory service as an authentication provider in Oracle WebLogic Server, as follows:

> **Note**
>
> The Oracle VM Manager upgrade process does not save and restore any configurations you create for external authentication providers. If you enable LDAP or Active Directory authentication and then upgrade Oracle VM Manager, you must complete the following steps after the upgrade to re-enable authentication.

1. Open the Oracle WebLogic Server Administration Console at:

   `https://hostname:7002/console`

   Where *`hostname`* is the Oracle VM Manager hostname or IP address.

2. Log in as the `weblogic` user.

3. Click **Lock & Edit** to modify the domain.

4. From the **Domain Structure** pane, select **Security Realms**, and then select **myrealm**.

   The settings page for the security realm displays.

5. Select the **Providers** tab and locate the **Authentication Providers** table.

6. Click **New** to create an authentication provider.

7. Specify a name for the authentication provider, select **LDAPAuthenticator** as the type of authentication provider, and then click **OK**.

   The new authentication provider displays in the **Authentication Providers** table.

8. Change the authentication sequence so that the LDAP authentication provider takes priority over other authentication providers.

   a. Click **Reorder** from the **Authentication Providers** table.

   b. Move the LDAP authentication provider to the top of the list and then click **OK**.

9. Select the LDAP authentication provider you created from the **Authentication Providers** table.

   The settings page displays.

10. On the **Common** tab, select **SUFFICIENT** as the value for **Control Flag** and then click **Save**.

11. Select the **Provider Specific** tab, configure the authentication provider as appropriate, and then click **Save**.

12. Click **Activate Changes** to apply your changes.

13. Restart the Oracle VM Manager service as root:

    ```
    # service ovmm restart
    ```

Verify that the LDAP authenticator is configured and that the LDAP users and groups are populated in Oracle WebLogic Server, as follows:

1. Log in to the Oracle WebLogic Server Administration Console.

2. From the **Domain Structure** pane, select **Security Realms**, and then select **myrealm**.

3. Select the **Users and Groups** tab.

4. Verify that the LDAP users and groups are populated as appropriate.

Chapter 5 Monitoring Oracle VM Server with SNMP

Oracle VM Server provides support for Simple Network Management Protocol (SNMP) monitoring. Find out what SNMP applications are installed by default and learn how to get started with SNMP.

5.1 Installed SNMP Packages

The default installation of Oracle VM Server includes the NET-SNMP suite of applications. These applications allow you use the SNMP protocol to monitor Oracle VM Server.

The following SNMP packages are installed by default:

Package	Description
`net-snmp`	SNMP agent daemon and documentation.
`net-snmp-libs`	Runtime libraries and management information bases (MIBs).
`net-snmp-utils`	Network management utilities such as snmpget and snmpwalk.
`ovs-snmp`	SNMP shared object module that lets you monitor Oracle VM Server. The Oracle VM Server management information base (MIB) defines several objects that provide information about the server and the virtual machines running on the server. The base OID for this MIB is `1.3.6.1.4.1.111.57.1`. You can find more information about the objects that this MIB defines in the following file on Oracle VM Server: `/usr/share/snmp/mibs/ORACLE-OVS-MIB.txt`. **Note** • This package is installed by default. However, you must manually add the object module to `snmpd.conf`, see Section 5.2, “Adding the Oracle VM Server Object Module”. • The Oracle VM Server object module applies to x86 systems only. This object module does not apply to Oracle VM Server for SPARC.

Check the installed packages as follows:

```
# rpm -qa | grep snmp
net-snmp-5.5-xx.x.x.xxx_x.x.x86_64
net-snmp-libs-5.5-xx.x.x.xxx_x.x.x86_64
net-snmp-utils-5.5-xx.x.x.xxx_x.x.x86_64
ovs-snmp-x.x-x.el6.x86_64
```

See the *NET-SNMP* documentation for more information.

5.2 Adding the Oracle VM Server Object Module

To monitor Oracle VM Server configuration with the `ovs-snmp` shared object module, you must add the following line to `/etc/snmp/snmpd.conf`:

```
dlmod ovs /usr/lib64/ovs-snmp/ovs.so
```

Note

The default snmpd.conf on Oracle VM Server provides an example configuration that you should modify to suit your business needs. It is beyond the scope of this documentation to describe a complete configuration for snmpd.conf. Refer to the snmpd.conf man page for more information.

From a high level, the steps to add the ovs-snmp shared object module to snmpd.conf are as follows:

1. Connect to the appropriate instance of Oracle VM Server.

2. Open /etc/snmp/snmpd.conf for editing.

3. Add the following line:

   ```
   dlmod ovs /usr/lib64/ovs-snmp/ovs.so
   ```

4. Ensure that you have read access rights in snmpd.conf.

   Tip

   You can temporarily add rocommunity public to the start of snmpd.conf to allow read access from all computers on the network.

5. Save and close /etc/snmp/snmpd.conf.

6. Restart the SNMP service, if it is running.

   ```
   # service snmpd restart
   ```

5.3 Enabling the SNMP Service

The SNMP daemons are disabled by default, and should be enabled if you intend to use SNMP to monitor an Oracle VM Server. You can check your configuration to determine whether the service has been enabled as follows:

```
# chkconfig --list |grep snmp
snmpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmptrapd       0:off   1:off   2:off   3:off   4:off   5:off   6:off
```

To enable the SNMP service, you can start it manually by issuing the following command:

```
# service snmpd start
```

To enable the SNMP service permanently, you can issue the following command:

```
# chkconfig --level 2345 snmpd on
```

Note

The Oracle VM Server MIB does not define any SNMP traps. You should start the snmptrapd service only if you require it for other purposes.

5.4 How to Retrieve MIB Objects

When the SNMP service is running, you can use NET-SNMP applications to retrieve MIB objects directly from the command line on Oracle VM Server. You can also use other applications, such as Oracle Enterprise Manager, to retrieve MIB objects.

This section provides examples for demonstration purposes only. You should refer to the manpages for NET-SNMP applications or the appropriate documentation for your NMS to determine how you should retrieve MIB objects to suit your business needs.

Note

The examples in this section:

- Assume that you have configured the `public` community for read access.
- Use a lower security level, SNMP v2c, for access. You should configure SNMP v3 to ensure that you restrict access control to authorized users. Refer to the appropriate documentation for information on access control and security levels as well as instructions on configuring SNMP v3.

Tip

If the last line of the output contains `No more variables left in this MIB View (It is past the end of the MIB tree)`, then you might not have read access rights in `snmpd.conf`. To resolve this issue, you can temporarily add `rocommunity public` to the start of `/etc/snmp/snmpd.conf` to allow read access from all computers on the network.

The following example uses the `snmpwalk` application to return values for all objects in the MIB tree:

```
# snmpwalk -v2c -c public localhost
SNMPv2-MIB::sysDescr.0 = STRING: Linux FQDN 3.8.13-68.2.2.el6uek.x86_64
 #2 SMP time_stamp x86_64
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (2654) 0:00:26.54
SNMPv2-MIB::sysContact.0 = STRING: Root <root@localhost>
 (configure /etc/snmp/snmp.local.conf)
SNMPv2-MIB::sysName.0 = STRING: FQDN
SNMPv2-MIB::sysLocation.0 = STRING: Unknown (edit /etc/snmp/snmpd.conf)
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (18) 0:00:00.18
SNMPv2-MIB::sysORID.1 = OID: SNMP-MPD-MIB::snmpMPDMIBObjects.3.1.1
SNMPv2-MIB::sysORID.2 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORID.3 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.4 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.5 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.6 = OID: IP-MIB::ip
SNMPv2-MIB::sysORID.7 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.8 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORDescr.1 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.3 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for SNMPv2 entities
SNMPv2-MIB::sysORDescr.5 = STRING: The MIB module for managing TCP implementations
SNMPv2-MIB::sysORDescr.6 = STRING: The MIB module for managing IP and ICMP implementations
SNMPv2-MIB::sysORDescr.7 = STRING: The MIB module for managing UDP implementations
SNMPv2-MIB::sysORDescr.8 = STRING: View-based Access Control Model for SNMP.
SNMPv2-MIB::sysORUpTime.1 = Timeticks: (17) 0:00:00.17
SNMPv2-MIB::sysORUpTime.2 = Timeticks: (17) 0:00:00.17
SNMPv2-MIB::sysORUpTime.3 = Timeticks: (17) 0:00:00.17
SNMPv2-MIB::sysORUpTime.4 = Timeticks: (17) 0:00:00.17
SNMPv2-MIB::sysORUpTime.5 = Timeticks: (18) 0:00:00.18
SNMPv2-MIB::sysORUpTime.6 = Timeticks: (18) 0:00:00.18
SNMPv2-MIB::sysORUpTime.7 = Timeticks: (18) 0:00:00.18
SNMPv2-MIB::sysORUpTime.8 = Timeticks: (18) 0:00:00.18
....
```

The following example uses the `snmpwalk` application to load `ORACLE-OVS-MIB` and return values for objects in the Oracle VM Server MIB tree:

```
# snmpwalk -v2c -c public localhost -OQ -m +ORACLE-OVS-MIB .1.3.6.1.4.1.111.57.1
ORACLE-OVS-MIB::ovsType.0 = Oracle VM Server
ORACLE-OVS-MIB::ovsVersion.0 = version
ORACLE-OVS-MIB::ovsClusterState.0 = DLM_Ready
ORACLE-OVS-MIB::ovsClusterType.0 = nfs
ORACLE-OVS-MIB::ovsClusterStorage.0 = hostname:/nfs/clusterfs/path
ORACLE-OVS-MIB::ovsManagerUUID.0 = 0004fb0000010000af78ad71a2719608
ORACLE-OVS-MIB::ovsServerpoolName.0 = my-pool
ORACLE-OVS-MIB::ovsAgentState.0 = Running
ORACLE-OVS-MIB::ovsFreeMemory.0 = 12766
ORACLE-OVS-MIB::vmIndex.0 = 0
ORACLE-OVS-MIB::vmIndex.1 = 1
ORACLE-OVS-MIB::vmIndex.2 = 2
ORACLE-OVS-MIB::vmType.0 = 0004fb00000600002eb4165c672efe28
ORACLE-OVS-MIB::vmType.1 = 0004fb0000060000959d078c46ec4268
ORACLE-OVS-MIB::vmType.2 = Domain-0
```

The following example uses the `snmptable` application to retrieve the running virtual machines on Oracle VM Server from the `ORACLE-OVS-MIB::vmTable` SNMP table:

Note

The `ORACLE-OVS-MIB::vmTable` SNMP table contains a cached value that expires after 60 seconds.

```
# snmptable -v 2c -c public localhost ORACLE-OVS-MIB::vmTable
SNMP table: ORACLE-OVS-MIB::vmTable

 vmIndex                           vmType
       0 0004fb00000600002eb4165c672efe28
       1 0004fb0000060000959d078c46ec4268
       2                         Domain-0
```

Chapter 6 Updating Oracle VM Server with Oracle Ksplice

Oracle VM Premier Support customers can use Oracle Ksplice to update Oracle VM Server with kernel, Xen, and user space patches. Oracle Ksplice allows you to apply security updates and bug fixes for your systems' kernel, the hypervisor, and certain user space libraries without having to schedule downtime and reboot machines.

Ksplice is freely available for Oracle customers who subscribe to Oracle Linux Premier Support, and to Oracle Cloud Infrastructure services. If you are an Oracle Linux Basic, Basic Limited, or Network Support subscriber, contact your sales representatives to discuss a potential upgrade of your subscription to a Premier Support plan.

Regular updates to the Oracle VM Servers are performed through Oracle VM Manager. However, the execution of Ksplice operations has not been integrated into Oracle VM Manager. Part of the Ksplice setup, such as configuring Server Update Repositories, does occur through Oracle VM Manager. Please review and understand the steps below before proceeding.

This chapter explains how to access the appropriate channels on ULN (Unbreakable Linux Network), set up the Ksplice client, and install Ksplice updates and patches on your Oracle VM Servers.

6.1 What Is Oracle Ksplice?

Linux kernels typically receive security updates and bug fixes on a regular basis. Applying these critical updates to your systems normally requires scheduled downtime, which is a costly operation because hosted services and applications become temporarily unavailable.

With Oracle Ksplice, however, systems can be updated without rebooting. As a result, access to services and applications is not interrupted, and your systems remain secure and up to date. Oracle Ksplice downloads patch updates from Oracle and applies these patches directly to running processes in memory. This provides an update to the running process without requiring it to restart. After this has been done, the package containing the binary update can be updated using the Unbreakable Linux Network (ULN) so that if the process is restarted later, the update is already applied.

Oracle Ksplice is available to customers with an Oracle Premier Support account, who have registered their systems with ULN. For additional information, refer to the *Oracle Linux Ksplice User's Guide*.

> **Warning**
>
> Use the *Oracle Linux Ksplice User's Guide* only as a reference for comprehensive background information about what Ksplice is and how it works. The instructions apply to generic Oracle Linux installations, while the procedures described in this *Oracle VM Administrator's Guide* contain different steps specific to Oracle VM Server. Use only the procedures in this *Oracle VM Administrator's Guide* to install and configure Ksplice for Oracle VM Server."

Ksplice updates are applied using a client application. Because Oracle VM Server takes advantage of user space updates as well as kernel updates, it requires the Ksplice Enhanced Client. Do not use the Ksplice Uptrack Client on Oracle VM Server. For additional information, refer to Working With the Ksplice Enhanced Client in the *Oracle Linux Ksplice User's Guide*.

6.2 Prerequisites for Ksplice on Oracle VM Server

Important

Ksplice is available for Oracle Premier Support customers. You require a valid Customer Support Identifier (CSI) to be able to install and use Ksplice for Oracle VM Server.

Starting with release 3.4.5 (build 1919), Oracle VM Server can use Ksplice to update the Xen hypervisor, the kernel and any user space packages. If your servers are running an earlier version of Oracle VM Server, you must upgrade and reboot first. To qualify for hypervisor patching, the servers must be running Xen 4.4.4-196 or newer.

To make sure that your servers meet the prerequisites for Ksplice, run the following checks:

1. Log in to the server.
2. From the Oracle Linux command line, check the current version of Oracle VM Server. It must be release 3.4.5 build 1919 or newer.

```
# cat /etc/ovs-info | head -n 4
OVS summary
release: 3.4.5
date: 201805301526
build: 1919
```

3. From the Oracle Linux command line, check the current version of Xen. It must be version 4.4.4-196 or newer.

```
# rpm -qa | grep "xen.*4.4.4"
xen-4.4.4-196.el6.x86_64
xen-tools-4.4.4-196.el6.x86_64
```

4. If the current versions of Oracle VM Server or Xen do not meet the minimum requirements, upgrade to the latest version of Oracle VM 3.4 before proceeding.
5. Repeat these steps on all Oracle VM Servers with which you intend to use Ksplice.

6.3 Accessing ULN Channels

The packages that are required to use Ksplice are hosted on ULN. However, because Oracle VM Server does not provide the tools that are required for ULN registration, it cannot connect to ULN by using Internet access. To install the required packages you must set up a local ULN Mirror that Oracle VM Servers can use to obtain the packages.

Note

Because Ksplice updates are cumulative, you can configure your local ULN mirror to store only the latest packages, which can improve synchronization time and storage requirements dramatically. To ensure that your local ULN mirror stores only the latest packages, edit the `/etc/sysconfig/uln-yum-mirror` file and set the `ALL_PKGS` parameter to 0 (`ALL_PKGS=0`). For more information, refer to ULN Mirror Configuration in the *Oracle Linux Ksplice User's Guide*.

The ULN Mirror can also be used to provide access to the most recent package updates so that patches can also be applied to on-disk binaries, as well as the processes that are patched in-memory by Ksplice.

The system hosting the mirrored ULN channels requires ULN registration and should be installed on a standard Oracle Linux 6 or Oracle Linux 7 host. You can install the ULN mirror on the same host as Oracle VM Manager.

Generic instructions to set up a local ULN mirror, and also configure it as a Ksplice mirror, are available in the following documentation resources:

- Creating and Using a Local ULN Mirror
- Configuring a Local ULN Mirror to Act as a Ksplice Mirror

When registering the system to be configured as ULN mirror, you must subscribe to the channels listed below, and make sure that the mirror provides the Oracle VM Servers access to those mirrored channels.

- Oracle VM 3.4 Latest (x86_64): `ovm34_x86_64_latest`
- Ksplice client and user space updates for Oracle VM (x86_64): `ovm34_x86_64_ksplice`
- Ksplice for Oracle Linux 6 (x86_64): `ol6_x86_64_ksplice`

6.4 Configuring Yum for Oracle Ksplice

To install the Ksplice Enhanced Client, and obtain the updates and patches for your environment, you need to configure Yum through Oracle VM Manager to enable access to the appropriate server update repositories. Provided you use the global server update settings, all discovered Oracle VM Servers take over the Yum repository configuration defined through the Oracle VM Manager.

The instructions in this section are based on the Oracle VM Manager Command Line Interface. For background information, usage instructions, and the command reference, consult the *Oracle VM Manager Command Line Interface User's Guide*.

Set up the server update repositories in Oracle VM Manager as follows:

1. Log in to the Oracle VM Manager Command Line Interface.

```
$ ssh -l admin ovmmgr.example.com -p 10000
admin@ovmmgr.example.com's password:
OVM>
```

2. Verify that you are running Oracle VM Manager release 3.4.5 (build 1919) or newer.

```
OVM> showversion
3.4.5.1919
```

3. Discover any Oracle VM Servers that have not yet been discovered with which you intend to use Ksplice, and take ownership of them.

   The example shows the CLI command and output using `ovmsvr01`. Execute this command for each of your servers, using either the fully qualified domain name or IP address.

   Note

   The *password* parameter is the Oracle VM Agent password on the Oracle VM Server in question. It is a required parameter for the `discoverServer` command.

```
OVM> discoverServer ipAddress=ovmsvr01.example.com password=******** takeOwnership=yes
Command: discoverServer ipAddress=ovmsvr01.example.com password=***** takeOwnership=yes
Status: Success
Time: 2018-06-03 19:46:20,633 PDT
```

```
JobId: 1528080367491
```

4. Add these server update repositories to the Yum configuration:

   - `uln_mirror_ovm34_x86_64_latest`
   - `uln_mirror_ovm34_x86_64_ksplice`
   - `uln_mirror_ol6_x86_64_ksplice`

   Note

   For the `create serverupdaterepository` command, the following parameters are required:

   - the package signature key (GPG key)
   - the path to they GPG key file
   - the Yum repository URL

```
OVM> create serverupdaterepository repositoryname=uln_mirror_ovm34_x86_64_latest enabled=yes pkgsignaturety
pkgsignaturekey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY name=uln_mirror_ovm34_x86_64_latest \
url=http://ulnmirror.example.com/yum/OracleVM/OVM34/latest/$basearch/ \
on serverupdategroup name=GlobalX86ServerUpdateConfiguration

Command: create serverupdaterepository repositoryname=uln_mirror_ovm34_x86_64_latest enabled=yes [...]
Status: Success
Time: 2018-06-03 19:47:43,093 PDT
JobId: 1528080450197
Data:
  id:0004fb0000310000e12a4dfe28933022  name:uln_mirror_ovm34_x86_64_latest

OVM> create serverupdaterepository repositoryname=uln_mirror_ovm34_x86_64_ksplice enabled=yes pkgsignaturet
pkgsignaturekey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY name=uln_mirror_ovm34_x86_64_ksplice \
url=http://ulnmirror.example.com/yum/OracleVM/OVM34/ksplice/$basearch/ \
on serverupdategroup name=GlobalX86ServerUpdateConfiguration

Command: create serverupdaterepository repositoryname=uln_mirror_ovm34_x86_64_ksplice enabled=yes [...]
Status: Success
Time: 2018-06-03 19:47:52,880 PDT
JobId: 1528080465622
Data:
  id:0004fb0000310000f522e7d0ee1911dc  name:uln_mirror_ovm34_x86_64_ksplice

OVM> create serverupdaterepository repositoryname=uln_mirror_ol6_x86_64_ksplice enabled=yes pkgsignaturetyp
pkgsignaturekey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY name=uln_mirror_ol6_x86_64_ksplice \
url=http://ulnmirror.example.com/yum/OracleLinux/OL6/ksplice/x86_64/ \
on serverupdategroup name=GlobalX86ServerUpdateConfiguration

Command: create serverupdaterepository repositoryname=uln_mirror_ol6_x86_64_ksplice enabled=yes [...]
Status: Success
Time: 2018-06-13 23:36:54,637 PDT
JobId: 1528958205006
Data:
  id:0004fb00003100003c4d3d4c252ee126  name:uln_mirror_ol6_x86_64_ksplice
```

5. Verify the Yum repository configuration on the Oracle VM Servers.

   a. Log in to the server.

   b. From the Oracle Linux command line, check the content of the directory /etc/yum.repos.d.

   ```
   # cat /etc/yum.repos.d/ovm.repo

   [uln_mirror_ovm34_x86_64_latest]
   gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY
   gpgcheck = 1
   baseurl = http://ulnmirror.example.com/yum/OracleVM/OVM34/latest/$basearch/
   name = uln_mirror_ovm34_x86_64_latest
   enabled = 1

   [uln_mirror_ovm34_x86_64_ksplice]
   gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY
   gpgcheck = 1
   baseurl = http://ulnmirror.example.com/yum/OracleVM/OVM34/ksplice/$basearch/
   name = uln_mirror_ovm34_x86_64_ksplice
   enabled = 1


   [uln_mirror_ol6_x86_64_ksplice]
   gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY
   gpgcheck = 1
   baseurl = http://ulnmirror.example.com/yum/OracleLinux/OL6/ksplice/x86_64/
   name = uln_mirror_ol6_x86_64_ksplice
   enabled = 1
   ```

   c. Run this check on all Oracle VM Servers with which you intend to use Ksplice.

6.5 Installing the Ksplice Enhanced Client

To update Oracle VM Server using Ksplice, you must use the Ksplice Enhanced Client. It is capable of updating certain shared libraries for user space processes – such as glibc, openssl and xen-tools – in addition to the Xen hypervisor and kernel updates.

The Ksplice Enhanced Client requires a direct connection to the Internet, so that it is able to connect to the Oracle Ksplice update server at https://updates-ksplice.oracle.com/uptrack. If your Oracle VM Servers are unable to connect directly to the Internet or your security policy restricts access, you can consider using an offline version of the client to apply updates from your locally configured ULN mirror. See Using the Ksplice Offline Enhanced Client for more information.

Since the Ksplice Enhanced Client package and its dependencies are only available on ULN and ULN registration is not possible on current versions of Oracle VM Server, the Oracle VM Servers on which you install the Ksplice Enhanced Client must have access to a host that is running a local ULN mirror, and to the ULN channels residing on that mirror. See Section 6.3, "Accessing ULN Channels" and Section 6.4, "Configuring Yum for Oracle Ksplice" for more information.

If you have configured a ULN Mirror and set up your yum configuration within Oracle VM Manager according to the instructions provided at Section 6.4, "Configuring Yum for Oracle Ksplice", the Ksplice Enhanced Client and Ksplice-aware user space packages for Oracle VM Server are available in the ULN channel uln_mirror_ovm34_x86_64_ksplice.

To install the Ksplice Enhanced Client, proceed as follows:

1. Log in to the Oracle VM Server as root.

2. Revert all prelinked binaries and dependent libraries to their original state, then use the yum command to remove the prelink package.

```
# prelink -au
# yum remove -y prelink
```

3. Install the `ksplice` package.

```
# yum install -y ksplice
Setting up Install Process
[...]
Dependencies Resolved

===========================================================================================================
 Package                       Arch        Version               Repository                               S
===========================================================================================================
Installing:
 ksplice                       x86_64      1.0.32-1.el6          uln_mirror_ovm34_x86_64_ksplice         5.
Installing for dependencies:
 boost-filesystem              x86_64      1.41.0-28.el6         uln_mirror_ovm34_x86_64_latest           4
 boost-python                  x86_64      1.41.0-28.el6         uln_mirror_ovm34_x86_64_latest          12
 boost-regex                   x86_64      1.41.0-28.el6         uln_mirror_ovm34_x86_64_latest          47
 ksplice-core0                 x86_64      1.0.32-1.el6          uln_mirror_ovm34_x86_64_ksplice         25
 ksplice-tools                 x86_64      1.0.32-1.el6          uln_mirror_ovm34_x86_64_ksplice         10
 uptrack                       noarch      1.2.47-0.el6          uln_mirror_ovm34_x86_64_ksplice         50
 uptrack-PyYAML                x86_64      3.08-4.el6            uln_mirror_ovm34_x86_64_ksplice         14
 uptrack-libyaml               x86_64      0.1.3-1.el6           uln_mirror_ovm34_x86_64_ksplice          4

Transaction Summary
===========================================================================================================
Install       9 Package(s)

Total download size: 1.7 M
Installed size: 6.0 M
Downloading Packages:
[...]
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY
Importing GPG key 0xEC551F03:
 Userid : customer_user_id <customer@example.com>
 Package: ovs-release-3.4-7.773.23.el6.x86_64 (@anaconda-OracleVMServer-201805301600.x86_64/3.4.5)
 From   : /etc/pki/rpm-gpg/RPM-GPG-KEY
[...]
Installed:
  ksplice.x86_64 0:1.0.32-1.el6

Dependency Installed:
  boost-filesystem.x86_64 0:1.41.0-28.el6       boost-python.x86_64 0:1.41.0-28.el6
  boost-regex.x86_64 0:1.41.0-28.el6            ksplice-core0.x86_64 0:1.0.32-1.el6
  ksplice-tools.x86_64 0:1.0.30-1.el6           uptrack.noarch 0:1.2.47-0.el6
  uptrack-PyYAML.x86_64 0:3.08-4.el6            uptrack-libyaml.x86_64 0:0.1.3-1.el6

Complete!
```

4. Edit `/etc/uptrack/uptrack.conf` to provide the client with the label of the local user space channel. If you followed the instructions in Section 6.4, “Configuring Yum for Oracle Ksplice” the channel should have the label `uln_mirror_ovm34_x86_64_ksplice`. Edit the file to include the lines:

```
[User]
```

```
yum_userspace_ksplice_repo_name = uln_mirror_ovm34_x86_64_ksplice
```

Also edit to add any other required configuration options. For example, to enable automatic installation updates, change the `autoinstall` option from `no` to `yes`:

```
autoinstall = yes
```

For more information on these options, see Configuring a Ksplice Client in the *Oracle Linux Ksplice User's Guide* at https://docs.oracle.com/cd/E37670_01/E39380/html/ol_ksplice_config.html.

Note that some options, such as `upgrade_on_reboot` may not apply to user space packages.

5. Update the system to install the Ksplice-aware versions of user space libraries. For example:

```
# yum update
```

6. Reboot the system so that it uses the new user space libraries.

```
# reboot
```

Using the Ksplice Offline Enhanced Client

If your Oracle VM environment resides in a highly secure data center where it is not possible to maintain a permanent Internet connection for the Oracle VM Servers that you wish to patch for security updates, you can use the Ksplice Offline Enhanced Client (`ksplice-offline`) as an alternative. Ksplice kernel, user space and Xen updates are bundled into RPM packages, specific for a particular version, and are updated within 48 hours after a new Ksplice patch becomes available. These updates are made available on ULN.

At regular intervals, you download the latest Ksplice update packages for your systems, and update your local ULN mirror. Once the Ksplice Offline Enhanced Client is installed on your Oracle VM Servers, they can connect to the local ULN mirror to retrieve updates.

The disadvantages of using the offline client include the delay after a patch becomes available and the requirement to manage and refresh the ULN mirror. However, since the ULN mirror must be maintained to apply on-disk binary updates for the Ksplice packages and other patch updates, this may be the preferred approach. Although there is a delay in patching, this delay also offers some further assurance of stability and further testing.

For installation, configuration, and usage instructions, refer to Installing and Configuring the Ksplice Offline Enhanced Client in the *Oracle Linux Ksplice User's Guide*. Note that the instructions provided in the *Oracle Linux Ksplice User's Guide* are generic Oracle Linux instructions.

⚠

Caution

If you want to switch between the online and offline version of the Ksplice Enhanced Client, you must first remove the installed Ksplice client software, and then install the new Ksplice client version. For example, to switch from the online client to the offline client, run the following commands:

```
# yum remove ksplice
# yum install ksplice-offline
```

To install and configure for Oracle VM Server:

1. Configure the ULN mirror as described in Section 6.3, “Accessing ULN Channels”

2. Configure the required yum repositories within Oracle VM Manager, as described in Section 6.4, “Configuring Yum for Oracle Ksplice”.

3. On each Oracle VM Server, revert prelinked binaries and remove prelink:

   ```
   # prelink -au
   # yum remove prelink
   ```

4. On each Oracle VM Server, install the Ksplice Offline Enhanced Client:

   ```
   #  yum install ksplice-offline
   ```

5. On each Oracle VM Server, edit the `/etc/uptrack/uptrack.conf` file to provide the client with the label of the local user space channel. If you followed the instructions in Section 6.4, “Configuring Yum for Oracle Ksplice” the channel should have the label `uln_mirror_ovm34_x86_64_ksplice`. Edit the file to include the lines:

   ```
   [User]
   yum_userspace_ksplice_repo_name = uln_mirror_ovm34_x86_64_ksplice
   ```

   Also edit to add any other required configuration options. For example, to enable automatic installation updates, change the `autoinstall` option from `no` to `yes`:

   ```
   autoinstall = yes
   ```

   For more information on these options, see Configuring a Ksplice Client in the *Oracle Linux Ksplice User's Guide* at https://docs.oracle.com/cd/E37670_01/E39380/html/ol_ksplice_config.html.

   Note that some options, such as `upgrade_on_reboot` may not apply to user space packages.

6. To install offline update packages, you must install the relevant packages for your system. When installing offline update packages you must specify the release in the command. For example, if you are installing the offline updates package for the Xen hypervisor, specify the release in the command as follows.

   ```
   # yum install ksplice-updates-xen-$RELEASE
   ```

   where *$RELEASE* is the update package that corresponds to the version of the hypervisor that is currently running, as shown in this example:

   ```
   # yum install ksplice-updates-xen-4.4.4-196.0.10.el6
   ```

7. From this point, the Ksplice Offline Enhanced Client behaves similarly to the standard online version of the Ksplice Enhanced Client.

6.6 Installing Oracle Ksplice Updates on Oracle VM Server

The Ksplice Enhanced Client can be used to apply updates and patches to the running Xen hypervisor, the running kernel, as well as certain Ksplice-aware user space libraries, including `xen-tools`. To install and manage the Ksplice patches on your system, use the `ksplice` command on the Oracle Linux command line.

To display the running user space, kernel and xen processes that the client can patch, use the `ksplice all list-targets` command. For each Ksplice-aware library, the command reports the running processes that would be affected by an update.

```
# ksplice all list-targets
User-space targets:

glibc-libpthread-2.12.1.209.0.3.ksplice1.el6_9.2:
 - multipathd (1582)
 - auditd (2077)
 - rsyslogd (2111)
```

```
 - rpcbind (2155)
[...]

glibc-libutil-2.12.1.209.0.3.ksplice1.el6_9.2:
 - sshd (2447)
 - xenconsoled (2533)
 - qemu-system-i38 (2537)
 - xend (2844)
[...]

xen-tools-tools_xenstore_xenstored-4.4.4.155.0.27.ksplice1.el6:
 - xenstored (2526)

xen-tools-tools_console_xenconsoled-4.4.4.155.0.27.ksplice1.el6:
 - xenconsoled (2533)

glibc-libm-2.12.1.209.0.3.ksplice1.el6_9.2:
 - multipathd (1582)
 - auditd (2077)
 - irqbalance (2126)
 - cupsd (2323)
 - ntpd (2460)
[...]

glibc-libnss_dns-2.12.1.209.0.3.ksplice1.el6_9.2:
 - ntpd (2460)
 - sshd (3975)
 - sshd (11292)

xen-tools-tools_libxc_libxenctrl.so.4.4.0-4.4.4.155.0.27.ksplice1.el6:
 - xenstored (2526)
 - xenconsoled (2533)
 - qemu-system-i38 (2537)
 - xend (2844)
[...]

openssl-libssl-1.0.1e.57.0.1.ksplice1.el6:
 - qemu-system-i38 (2537)
 - master (2706)
 - pickup (2736)
 - qmgr (2737)
 - xend (2844)
[...]

xen-tools-tools_xenstore_libxenstore.so.3.0.3-4.4.4.155.0.27.ksplice1.el6:
 - xenconsoled (2533)
 - qemu-system-i38 (2537)
 - xend (2844)
[...]

Kernel version: Linux/x86_64/4.1.12-124.14.5.el6uek.x86_64/#2 SMP Fri May 4 15:36:12 PDT 2018
xen/x86_64/4.4.4OVM/Fri May 11 20:21:05 PDT 2018
```

To display the updates that are available for installation, use the `ksplice all show --available` command. If you want to restrict the scope of the command to a particular category, use these alternatives instead:

- To display the available user space updates: `ksplice -n user upgrade`
- To display the available kernel updates: `ksplice -n kernel upgrade`
- To display the available xen updates: `ksplice -n xen upgrade`

```
#  ksplice -n xen upgrade
The following steps will be taken:
Install [d71xqwov]: update.
Install [ion5usqz]: update 3.
Install [0323dejx]: update 2.
```

To install the available Ksplice updates, use the `ksplice -y all upgrade` command. If you want to restrict the scope of the command to a particular category, use these alternatives instead:

- To install all user space updates: `ksplice -y user upgrade`
- To install all kernel updates: `ksplice -y kernel upgrade`
- To install all xen updates: `ksplice -y xen upgrade`

```
#  ksplice -y xen upgrade
The following steps will be taken:
Install [d71xqwov]: update.
Install [ion5usqz]: update 3.
Install [0323dejx]: update 2.
100% |###########################################################################################|
Done!
```

```
# ksplice -y user upgrade
Updating on-disk packages for new processes
Setting up Update Process
uln_mirror_ovm34_x86_64_ksplice                                              | 2.5 kB     00:00
Resolving Dependencies
[...]

==============================================================================================================
 Package          Arch         Version                                  Repository                           S
==============================================================================================================
Updating:
 glibc            i686         2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice      4
 glibc            x86_64       2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice      3
 glibc-common     x86_64       2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice
 glibc-devel      x86_64       2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice      9
 glibc-headers    x86_64       2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice      6
 nscd             x86_64       2:2.12-1.209.0.3.ksplice1.el6_9.2        uln_mirror_ovm34_x86_64_ksplice      2
 openssl          x86_64       2:1.0.1e-57.0.1.ksplice1.el6             uln_mirror_ovm34_x86_64_ksplice      1
 xen-tools        x86_64       2:4.4.4-155.0.27.ksplice1.el6            uln_mirror_ovm34_x86_64_ksplice      8
Installing for dependencies:
 ksplice-helper   x86_64       1.0.32-1.el6                             uln_mirror_ovm34_x86_64_ksplice

Transaction Summary
==============================================================================================================
Install       1 Package(s)
Upgrade       8 Package(s)

Total download size: 34 M
Downloading Packages:
(1/9): glibc-2.12-1.209.0.3.ksplice1.el6_9.2.i686.rpm                                            | 4.4 MB
(2/9): glibc-2.12-1.209.0.3.ksplice1.el6_9.2.x86_64.rpm                                          | 3.8 MB
(3/9): glibc-common-2.12-1.209.0.3.ksplice1.el6_9.2.x86_64.rpm                                   |  14 MB
(4/9): glibc-devel-2.12-1.209.0.3.ksplice1.el6_9.2.x86_64.rpm                                    | 992 kB
(5/9): glibc-headers-2.12-1.209.0.3.ksplice1.el6_9.2.x86_64.rpm                                  | 619 kB
(6/9): ksplice-helper-1.0.32-1.el6.x86_64.rpm                                                    |  17 kB
(7/9): nscd-2.12-1.209.0.3.ksplice1.el6_9.2.x86_64.rpm                                           | 232 kB
(8/9): openssl-1.0.1e-57.0.1.ksplice1.el6.x86_64.rpm                                             | 1.5 MB
(9/9): xen-tools-4.4.4-155.0.27.ksplice1.el6.x86_64.rpm                                          | 8.7 MB
--------------------------------------------------------------------------------------------------------------
Total                                                                                       MB/s |  34 MB
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
[...]
Complete!
```

To display the updates that have been applied to the system, use the `ksplice all show` command. If you want to restrict the scope of the command to a particular category, specify `user`, `kernel` or `xen` instead of `all`.

```
# ksplice all show

Ksplice user-space updates:
No Ksplice user-space updates installed

Ksplice kernel updates:
Installed updates:
None

Effective kernel version is 4.1.12-124.14.5.el6uek

Ksplice xen updates installed:
[ion5usqz]: update 3.
[0323dejx]: update 2.
[d71xqwov]: update.
```

Note

After Ksplice has applied updates to a running kernel, the kernel has an effective version that is different from the original boot version displayed by the `uname -a` command. Use the `ksplice kernel uname -r` command to display the effective version of the kernel.

To remove Ksplice updates from the system, use the `remove` subcommand. You can choose to remove Ksplice updates for the category `user`, `kernel` or `xen`.

```
# ksplice -y xen remove --all
The following steps will be taken:
Remove [d71xqwov]: update.
Remove [ion5usqz]: update 3.
Remove [0323dejx]: update 2.
100% |############################################################################################|
Done!
```

For more information about using the `ksplice` command, see the `ksplice(8)` manual page.

Chapter 7 Provisioning ISO Files for PVM Guest Installations

Important

As of Oracle VM Release 3.4.6, support for PVM guests is removed. For more information, Section 1.5, “Disabling Paravirtualized Guests on Oracle VM Server”.

When you create a PVM guest from an ISO file, you cannot use an ISO file from a repository to install the operating system. Oracle VM requires that the ISO file is mounted so that its internal file system contents are available during the installation of the operating system for a PVM guest. The mounted ISO file can be made available via an NFS, HTTP or FTP server. The following examples show how to create and use mounted ISO files on an NFS share, and on an HTTP server.

Example 7.1 Creating an installation tree on an NFS share

This example creates an installation tree for *paravirtualized guest* by mounting an ISO file. The installation tree is made available via an NFS share. On the NFS server, enter

```
# mkdir -p /isos/EL5u6-x86_64
# mount -o ro,loop /path/Enterprise-R5-U6-Server-x86_64-dvd.iso /isos/EL5u6-x86_64
# exportfs *:/isos/EL5u6-x86_64/
```

When you create the *virtual machine* using the Oracle VM Manager Web Interface, enter the installation location in the **Network Boot Path** field in the **Create Virtual Machine** wizard as:

```
nfs:example.com:/isos/EL5u6-x86_64
```

Example 7.2 Creating an installation tree on an HTTP server

This example creates an installation tree from an ISO file that can be accessed via HTTP. On the HTTP server, enter

```
# cd /var/www/html
# mkdir EL5u6-x86_64
# mount -o ro,loop /path/Enterprise-R5-U6-Server-x86_64-dvd.iso EL5u6-x86_64
```

When you create the virtual machine using the Oracle VM Manager Web Interface, enter the installation location in the **Network Boot Path** field in the **Create Virtual Machine** wizard as:

```
http://example.com/EL5u6-x86_64
```

Tip

If you have multiple ISO files, you can mount each ISO file and copy the contents into a single directory. All the ISO files are then available from the same location.

Chapter 8 Installing and Using the Oracle VM Guest Additions

The Oracle VM Guest Additions allow bi-directional communication between *Oracle VM Manager* and the *guest* operating system of a *virtual machine* running in the Oracle VM environment. The Oracle VM Guest Additions provide fine-grained control over the configuration and behavior of components running within the virtual machine directly from Oracle VM Manager!

8.1 Features of the Oracle VM Guest Additions

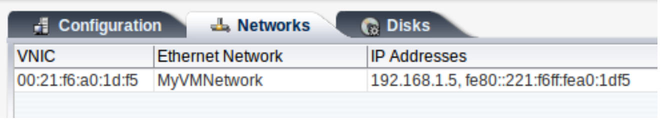
The Oracle VM Guest Additions allow direct integration between guest software and the virtualization layer, to assist in orchestration and automation of complex, multi-virtual machine deployments. This integration can be used between Oracle Solaris, Microsoft Windows, and Oracle Linux running the Oracle Unbreakable Enterprise Kernel (UEK).

The Oracle VM Guest Additions allow you to:

- Send key-value pairs to a *virtual machine*, or *guest*, and to retrieve messages from the guest.
- More easily get information about virtual machines within Oracle VM Manager, such as reporting on IP addressing.
- Use the template configuration facility to automatically configure virtual machines as they are first started.
- Trigger programmed events by sending messages directly to a virtual machine from Oracle VM Manager.
- Query virtual machines to obtain information pertaining to previous messages.
- Interact with the Oracle VM Utilities `ovm_vmmessage` command.

Virtual Machine IP Address

When you install the Oracle VM Guest Additions, the IP address of the *virtual machine* displays in the **Networks** subtab of the **Virtual Machines** perspective in the Oracle VM Manager Web Interface.



VNIC	Ethernet Network	IP Addresses
00:21:f6:a0:1d:f5	MyVMNetwork	192.168.1.5, fe80::221:f6ff:fea0:1df5

8.2 Oracle VM Guest Additions Packages

To install the Oracle VM Guest Additions, you first download a set of packages from a yum repository to the virtual machine. You can download the packages from the Unbreakable Linux Network (ULN) or the Oracle Linux Yum Server. After you download the required packages, you use the `yum install` to install them.

The following table lists the packages for the Oracle VM Guest Additions:

Package	Description	Required or Optional
`libovmapi`	Library that adds support for the Oracle VM API.	Required
`libovmapi-devel`	Library that adds developer support for the Oracle VM API.	Optional

Package	Description	Required or Optional
ovmd	Daemon that handles configuration events and enables sending and receiving of messages between the virtual machine and Oracle VM Manager.	Required
xenstoreprovider	Library that communicates with the Oracle VM API kernel infrastructure.	Required
ovm-template-config	Basic operating system configuration scripts.	Required
ovm-template-config-authentication	Script for configuring virtual machine authentication.	Optional
ovm-template-config-datetime	Script for configuring virtual machine datetime settings.	Optional
ovm-template-config-firewall	Script for configuring virtual machine firewall.	Optional
ovm-template-config-network	Script for configuring virtual machine network settings.	Optional
ovm-template-config-selinux	Script for configuring virtual machine selinux settings.	Optional
ovm-template-config-ssh	Script for configuring virtual machine ssh settings.	Optional
ovm-template-config-system	Script for configuring virtual machine system settings.	Optional
ovm-template-config-user	Script for configuring virtual machine user settings.	Optional

Microsoft Windows Guests

For Microsoft Windows guests, the Oracle VM Guest Additions are included in the Oracle VM Paravirtual Drivers. The implementation of the guest additions consists of these components:

- vmapi.dll: a dynamic link library that exposes callable interfaces
- ovmsvc: the service that receives and sends events and messages
- xenpci: the driver that interacts with the hypervisor
- ovmcmd: the command used to invoke the guest additions

vmapi.dll

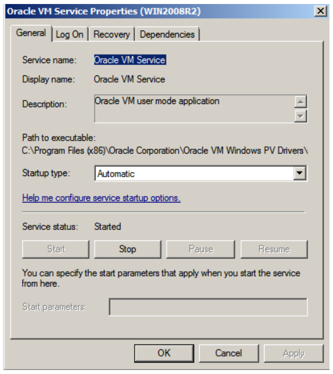
The vmapi.dll exposes interfaces to functions for communication between Windows applications and the xenpci driver in the Windows guest operating system. Applications load this library and call the exported functions to access xenpci. Normally device I/O controls (IOCTLs) are used to communicate with the xenpci driver. The following table shows function names and their descriptions:

Function	Description
OVMAPI_Register	initializes OVMAPI and optionally registers a callback
OVMAPI_ParamGetValue	retrieves name-value pairs stored within the OVMAPI engine
OVMAPI_ParamSetValue	creates or modifies a name-value pair
OVMAPI_Subscribe	receives particular events in the application registered event handler

Function	Description
OVMAPI_UnSubscribe	blocks particular events in the application registered event handler
OVMAPI_EventComplete	completes the event with a finalizing status code at a later time
OVMAPI_ParamGetValueSize	retrieves the size, in bytes, of a parameter value
OVMAPI_GetSessionFileDescriptors	retrieves the internal file descriptor for use in making special calls to the driver
OVMAPI_UserEventPublish	sends application-specific events to the management server and to other OVMAPI-enabled applications running on the same VM
OVMAPI_ParamGetAllNames	browses for existing parameters that may be of interest to the application
OVMAPI_ParamGetCount	retrieves the number of existing parameters
OVMAPI_UnRegister	terminates OVMAPI usage

ovmsvc

The ovmsvc service starts automatically by default when the Windows VM is booted. The command to install or uninstall this service is ovmsvc install or ovmsvc uninstall respectively. The ovmsvc service monitors messages sent from Oracle VM Manager, and sends Windows VM specific information, such as the operating system version and the VM's IP address. It uses the vmapi.dll to implement communication between the Windows guest and Oracle VM Manager. The following figure shows ovmsvc running inside a Windows VM.



Oracle VM Service Properties (WIN2008R2)

Service name: Oracle VM Service

Display name: Oracle VM Service

Description: Oracle VM user mode application

Path to executable:
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers\

Startup type: Automatic

Help me configure service startup options.

Service status: Started

You can specify the start parameters that apply when you start the service from here.

Start parameters:

When ovmsvc is started, it opens the xenpci device and registers a callback function where messages are processed. A thread is created to monitor events corresponding with vmapi messages. The current callback function of ovmsvc processes VM shutdown messages; other messages, for example "snapshot", could be implemented as additional features.

xenpci

Applications exchanging messages with Oracle VM Manager use the xenpci driver, which implements several IOCTLs to process messages. The xenpci driver directly accesses xenstore, a memory area shared by Xen domains. The ovmsvc calls API functions to send or read VM messages to or from xenstore.

The xenpci driver uses the xenbus_watch function to monitor incoming messages. The xenpci driver initialization registers a watch function on the xenstore key "control/oracle-vmapi/to-guest/last-write". Messages sent by Oracle VM Manager are set in this xenstore key. The watch function of xenpci is triggered when this xenstore key changes. The watch function checks and sends out events, a monitor thread acknowledges events, and a callback function processes API messages.

The `ovmsvc` sends Windows VM messages during its startup process, and uses an IOCTL in `xenpci` to write the xenstore key "`control/oracle-vmapi/from-guest/%d/%s`". The `ovmsvc` updates the message periodically to synchronize the information, such as guest IP address, from the guest to Oracle VM Manager.

Oracle Solaris

The Oracle VM Guest Additions are also available for Oracle Solaris, both on SPARC and x86. They can be optionally installed from a Solaris IPS repository: `pkg://solaris/system/management/ovm-guest-additions`.

8.3 Downloading the Oracle VM Guest Additions Packages

The Oracle VM Guest Additions packages are available for download from the *addons* channel for the *guest* operating system on the Unbreakable Linux Network (ULN) or the Oracle Linux Yum Server. The appropriate channels for the Oracle Linux guest operating systems on the Oracle Linux yum server are:

- Oracle Linux 8 Add ons (x86_64) – *Oracle Linux 8 (x86_64) Addons*
- Oracle Linux 7 Add ons (x86_64) – *Oracle Linux 7 (x86_64) Addons*
- Oracle Linux 6 Add ons (x86_64) – *Oracle Linux 6 (x86_64) Addons*

Downloading from ULN

To download the Oracle VM Guest Additions packages from ULN, subscribe your system to the *addons* channel.

Alternatively, you can create a yum server that acts as a local mirror of the ULN *addons* channel.

For Oracle Linux Release 8, see the Oracle Linux documentation for more information at *Registering With the Unbreakable Linux Network*

For Oracle Linux Release 7: or Oracle Linux Release 6, see the following Oracle Linux documentation for more information:

Oracle Linux Release 7:

- *Managing ULN Channel Subscription by Using the ULN Web Interface*
- *Creating and Using a Local ULN Mirror*

Oracle Linux Release 6:

- *Managing ULN Channel Subscription by Using the ULN Web Interface*
- *Creating and Using a Local ULN Mirror*

Downloading from the Oracle Linux Yum Server

By default, the yum repository configuration file on Oracle Linux contains a section that defines the *addons* channel on the Oracle Linux Yum Server.

To download the Oracle VM Guest Additions packages from the public yum repository, you need to enable the *addons* channel in the yum configuration file.

For Oracle Linux Release 8, see the Oracle Linux documentation for more information at *Obtaining Errata and Updates From the Oracle Linux Yum Server*

For Oracle Linux Release 7: or Oracle Linux Release 6, see the following Oracle Linux documentation for more information:

Oracle Linux Release 7:

- *Yum Repository Configuration*

Oracle Linux Release 6:

- *Yum Repository Configuration*

Microsoft Windows Guests

For Microsoft Windows guests, the Oracle VM Guest Additions are included in the Oracle VM Paravirtual Drivers. Refer to the Oracle VM Paravirtual Drivers for Microsoft Windows documentation library. Follow the download instructions for the selected release of the paravirtual drivers.

Oracle Solaris

For Oracle Solaris, both on SPARC and x86, the Oracle VM Guest Additions can be downloaded from a Solaris IPS repository at https://pkg.oracle.com/solaris/release/en/index.shtml or through https://support.oracle.com/portal/ with your support contract access.

8.4 Installing the Oracle VM Guest Additions

To install the Oracle VM Guest Additions, do the following:

1. Download the required packages for the Oracle VM Guest Additions.

2. Run the `yum install` command to install the packages; for example,

```
# yum install libovmapi xenstoreprovider ovmd xenstoreprovider
```

Microsoft Windows Guests

For Microsoft Windows guests, the Oracle VM Guest Additions are included in the Oracle VM Paravirtual Drivers. Refer to the Oracle VM Paravirtual Drivers for Microsoft Windows documentation library. Follow the installation instructions for the selected release of the paravirtual drivers.

Oracle Solaris

To install the Oracle VM Guest Additions on Oracle Solaris, do the following:

1. Ensure that an appropriate 'solaris' repository publisher is configured.

2. Run the IPS command to install the package; for example,

```
# pkg install ovm-guest-additions
```

8.5 Upgrading the Oracle VM Guest Additions

You should update the Oracle VM Guest Additions packages regularly to ensure they function correctly. You should update the packages after you create a new *virtual machine* from an Oracle VM template on which the Oracle VM Guest Additions are installed.

To update the Oracle VM Guest Additions, do the following:

1. Ensure your virtual machine is connected to a yum repository that contains the packages for the Oracle VM Guest Additions.

2. Run the `yum update` command to install the packages; for example,

   ```
   # yum update libovmapi xenstoreprovider ovmd xenstoreprovider
   ```

Microsoft Windows Guests

For Microsoft Windows guests, the Oracle VM Guest Additions are included in the Oracle VM Paravirtual Drivers. Refer to the Oracle VM Paravirtual Drivers for Microsoft Windows documentation library. Follow the upgrade instructions for the selected release of the paravirtual drivers. Silent upgrade is available in case you need to upgrade a large number of Windows guests.

Oracle Solaris

To upgrade the Oracle VM Guest Additions on Oracle Solaris, do the following:

1. Ensure that an appropriate 'solaris' repository publisher is configured.

2. Run the IPS command to update the package; for example,

   ```
   # pkg update ovm-guest-additions
   ```

8.6 Using the Oracle VM Guest Additions (ovmd)

The Oracle VM Guest Additions daemon, `ovmd`, facilitates a bi-directional *messaging* channel between *Oracle VM Manager* and the *guest*. It allows first-boot installation configuration, and is capable of sending and receiving messages consisting of key-value pairs.

In previous releases you could use the `ovmd` utility to send key/value messages to a virtual machine. This feature is now included directly in Oracle VM Manager. Although this section mentions the options available to send messages to a virtual machine using `ovmd`, you should instead use the Oracle VM Manager Web Interface or Oracle VM Manager Command Line Interface to send key/value messages. The virtual machine must have the Oracle VM Guest Additions daemon installed and running. See *sendVmMessage* in the *Oracle VM Manager Command Line Interface User's Guide* , or the *Send VM Messages* section in the *Oracle VM Manager User's Guide* for more information.

Used in conjunction with the `ovm-template-config` script, the `ovmd` utility can be used to remotely configure system and application configuration parameters within a *virtual machine* as it boots. See Section 8.8, “The Oracle VM Template Configuration Script and Modules” for more information on this facility.

Oracle VM Manager makes use of `ovmd` in order to obtain IP addressing information from the guest to include in the Oracle VM Manager Web Interface when displaying detailed virtual machine information. See Virtual Machine IP Address.

You can run `ovmd` directly from the command line to perform actions outside of `ovmd`'s function as a daemon or system service. Running `ovmd` using the `--help` parameter provides you with a breakdown of the options supported when run directly from the command line.

> **Note**
>
> On Oracle Solaris, the command options and script names may differ from those presented for Oracle Linux. For details, please refer to the ovmd(1M) man page,

> which is installed along with the ovm-guest-additions package on Oracle Solaris.

Syntax

ovmd [{ -p | --set-param= } *param*] [{ -g | --get-param= } *key*] [{ -r | --delete-param= } *key*] [{ -x | --delete-params }] [{ -l | --list-params }] [{ -e | --event= } *event*] [{ -s | --script= } *script*] [{ -d | --debug= }{ 0 | 1 | 2 }] [{ -f | --pid-file= } *filename*] [{ -t | --time-period= } *seconds*] [{ -v | --version }] [{ -h | --help }]

Options

The following table shows the available options for this command:

Option	Description
{ -p | --set-param= } *param*	Set a parameter in the format of *key*=*value*.
{ -g | --get-param= } *key*	Get the value of the parameter by key name.
{ -r | --delete-param= } *key*	Delete the parameter by key name.
{ -x | --delete-params }	Delete all parameters.
{ -l | --list-params }	List all parameters.
{ -e | --event= } *event*	Inject an event.
{ -s | --script= } *script*	Run a script on the virtual machine.
{ -d | --debug= }{ 0 | 1 | 2 }	Set the debug level. 0 is DEBUG_OFF, 1 is DEBUG_STDERR, and 2 is DEBUG_SYSLOG. The default is 2.
{ -f | --pid-file= } *filename*	Set the path name of the process ID (PID) file.
{ -t | --time-period= } *seconds*	Set the period for daemon mode. The default is 10 seconds.
{ -v | --version }	Show the ovmd script version number and exit.
{ -h | --help }	Show help on the ovmd command options.

Examples

Example 8.1 Showing the ovmd script version

```
# ovmd -v
```

Example 8.2 Running a script on a virtual machine

```
# ovmd --script=/scripts/cleanup
```

Example 8.3 Sending a message from a virtual machine to Oracle VM Manager

```
# ovmd -p key1=value1
```

See Section 8.6.2, “Sending Messages to Virtual Machines” for more information on sending and receiving messages using the ovmd script.

Example 8.4 Listing messages sent from Oracle VM Manager on a virtual machine

```
# ovmd -\-list
{"key1":"value1"}
```

```
{"key2":"value2"}
```

Example 8.5 Deleting a message on a virtual machine

```
# ovmd -r key1
```

8.6.1 Using the Oracle VM Guest Additions Daemon to Enable First-Boot Configuration

If you are configuring a virtual machine to act as a template, or if you intend to *clone* it, you can enable a first-boot configuration. This configuration causes the virtual machine to behave as if it is booting for the first time each time it boots. As a result, the virtual machine prompts for configuration input either by the VM API or on the virtual machine console. This can be achieved by running the following commands as *root* within the virtual machine:

```
# ovmd -s cleanup
# service ovmd enable-initial-config
# shutdown -h now
```

On next boot, the virtual machine acts as if it is performing a first-time boot.

If you have configured `ovmd` to run as a service, you can configure it remotely using the *messaging* facility and the `ovm-template-config` script.

See the example Section 8.6.2, “Sending Messages to Virtual Machines” for more information on using the messaging channel. Also see Section 8.8, “The Oracle VM Template Configuration Script and Modules” for information about the `ovm-template-config` script.

8.6.2 Sending Messages to Virtual Machines

This section gives an example of a message exchange between Oracle VM Manager and a running Oracle Linux virtual machine with Oracle VM Guest Additions installed.

Example 8.6 Sending a message from the guest to Oracle VM Manager

Using `ovmd`, you send information from within the virtual machine to your Oracle VM Manager using the following syntax:

```
# ovmd -p key1=value1
```

The message appears in the Oracle VM Manager user interface, as a *Virtual Machine API Incoming Message* event for the virtual machine in question. When you expand the event, the description shows the key-value pair and the date and time when the information exchange took place.

Example 8.7 Sending a message from Oracle VM Manager to a virtual machine

Using `ovmd` from within the guest, you can retrieve the message sent from Oracle VM Manager using the following syntax:

```
# ovmd -\-list
{"key1":"value1"}
{"key2":"value2"}
```

The `ovmd -\-list` command retrieves all messages, both sent and received. You can identify the specific message you are looking for by its key. To remove obsolete messages, use the following syntax:

```
# ovmd -r key1
# ovmd -\-list
{"key2":"value2"}
```

8.6.3 Configuring the Oracle VM Guest Additions Daemon to Run as a Service

To enable `ovmd` to run as a service on Oracle Linux, run the `chkconfig` command as *root*:

```
# chkconfig ovmd on
```

To start the `ovmd` service, run the following as *root*:

```
# service ovmd start
```

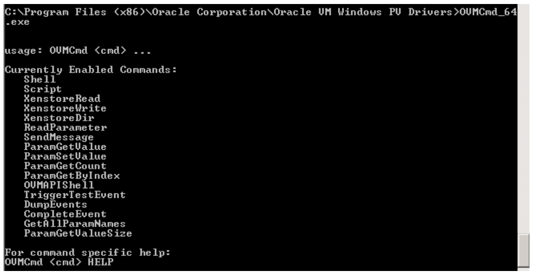
When configured as a service, `ovmd` listens for message requests sent from Oracle VM Manager.

8.7 Using the Oracle VM Guest Additions with Microsoft Windows (ovmcmd)

The Oracle VM Guest Additions provide a command line tool named `ovmcmd` for interacting with their functions, similar to the `ovmd` command in Linux. The command line tool is distributed during installation of the Oracle VM Paravirtual Drivers, and is located in the folder: `C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers`.

The Shell

The executable binary file `ovmcmd` is used with 32-bit Windows, while `ovmcmd_64` is used with 64-bit Windows. You can run `ovmcmd` directly by using the Windows command line. The `ovmcmd` command provides its own shell mode, which is convenient for setting multiple variables. Enter `ovmcmd` without any parameter to display the list of supported interfaces.

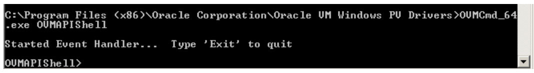


```
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers>OVMCmd_64
.exe


usage: OVMCmd <cmd> ...

Currently Enabled Commands:
   Shell
   Script
   XenstoreRead
   XenstoreWrite
   XenstoreDir
   ReadParameter
   SendMessage
   ParamGetValue
   ParamSetValue
   ParamGetCount
   ParamGetByIndex
   OVMAPIShell
   TriggerTestEvent
   DumpEvents
   CompleteEvent
   GetAllParamNames
   ParamGetValueSize

For command specific help:
OVMCmd <cmd> HELP
```

Enter one of these commands to run ovmcmd in shell mode:

- 32-bit Windows: `ovmcmd ovmapishell`
- 64-bit Windows: `ovmcmd_x64 ovmapishell`

This example shows the ovmcmd shell on a 64-bit Windows:



```
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers>OVMCmd_64
.exe OVMAPIShell

Started Event Handler...  Type 'Exit' to quit

OVMAPIShell>
```

The Commands

In shell mode, `ovmcmd` sets up a session and connects an event handler. The user can enter commands at the `OVMAPIShell` shell prompt without having to re-issue `ovmcmd`. This is optional, as individual commands can be specified on the Windows command line executing `ovmcmd`. Shell mode can be used to send and receive messages containing parameter values, or to display the contents of the xenstore.

Commands for sending and receiving messages are illustrated under *Conversation Between Guest and Oracle VM Manager*. Command names in `ovmcmd` are not case sensitive.

The `Xenstore` commands should only be used by people with substantial knowledge of the internals of Xen. Those commands are:

- `XenstoreRead`: reads the value of a xenstore key. The parameter following the subcommand is the xenstore key to read.

  ```
  OVMAPIShell>XenstoreRead /local/domain/708/device/vif/0/state
  4
  OVMAPIShell>
  ```

- `XenstoreWrite`: writes a value to a xenstore key or creates a new xenstore key. The parameters are the xenstore key and the value to be written. If the xenstore key is read-only, an error message appears.

  ```
  OVMAPIShell>XenstoreWrite /local/domain/708/control/test 1
  Operation Succeeded
  OVMAPIShell>XenstoreWrite /local/domain/708/test 2
  Operation Failed
  OVMAPIShell>
  ```

- `XenstoreDir`: lists a xenstore directory. The parameter is the xenstore directory to display.

  ```
  OVMAPIShell>XenstoreDir /local/domain/0
  vm
  device
  control
  error
  memory
  guest
  hvmpv
  data
  cpu
    1
      availability
    2
      availability
  [...]
  OVMAPIShell>
  ```

The `SendMessage` command is used to send messages. However, messages sent this way are not saved.

```
OVMAPIShell>SendMessage value 1
Operation Succeeded
OVMAPIShell>
```

To send a message and at the same time save the parameter key/value pair, use the `ParamSetValue` command.

```
OVMAPIShell>ParamSetValue guestparam1 guestval 1
ParamSetValue returned 0
OVMAPIShell>OVMCmdEventHandler: Received Event type 64, size 12, ID 6
ParamSetValue returned 0
```

To retrieve the value of a specified parameter, use the `ParamGetValue` command.

```
OVMAPIShell>ParamGetValue guestparam1 guestval
ParamGetValue returned 0, guestparam1 = guestval (len = 9)
```

Conversation Between Guest and Oracle VM Manager

Conversations between the guest VM and Oracle VM Manager are based on sending and receiving messages.

From the Windows VM side of the conversation: use the `ovmcmd` command `ParamSetValue` to set a parameter and send a message from the guest VM to Oracle VM Manager. For example `ParamSetValue mykey myval` sets the value of key "mykey" to "myval", and sends a message to Oracle VM Manager.

```
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers>OVMCmd_64 ParamSetValue mykey myval
ParamSetValue returned 0
```

The `SendMessage` command can also be used.

```
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers>OVMCmd_64 SendMessage foo bar
Operation Succeeded
```

Use the ReadParameter command as shown below to read messages from Oracle VM Manager. Output can optionally be piped to a file for later processing.

```
C:\Program Files (x86)\Oracle Corporation\Oracle VM Windows PV Drivers>OVMCmd_64 ReadParameter sesame
Success: sesame = street (7)
```

For the Oracle VM Manager side of the conversation: to display the message from the Oracle VM Manager user interface, select the VM in the Servers and VMs tab, right click, and select Display Events.

Alternatively, send a message to the guest with the Oracle VM CLI sendVmMessage command.

```
OVM> sendvmmessage vm name=VMNAME key=managerparam message=managerval log=no
```

See *sendVmMessage* in the *Oracle VM Manager Command Line Interface User's Guide* for more information.

8.8 The Oracle VM Template Configuration Script and Modules

The Oracle VM Guest Additions include a set of packages that can help with the automatic configuration of *virtual machine* as they are created from a template and booted for the first time. The master package for this facility is known as `ovm-template-config`. The Oracle VM template configuration script can be used to configure a virtual machine remotely using the Oracle VM *messaging* facility via `ovmd`.

8.8.1 Template Configuration Script (`ovm-template-config`)

The Oracle VM Template Configuration Script, `ovm-template-config` works in conjunction with a set of modular configuration scripts that function in a manner very similar to the standard Linux System V, `init.d` and `chkconfig`, script model. Control over how configuration modules are run is handled within the `/etc/template.d` directory on the *guest* virtual machine. The configuration module scripts are stored in `/etc/template.d/scripts`.

The `ovm-template-config` script is the master script that is used to control all enabled modules. Running `ovm-template-config` with the `--help` parameter provides a usage breakdown.

For remote configuration, `ovm-template-config` is used in conjunction with `ovmd` to capture configuration parameters that have been sent to the guest using the Oracle VM messaging facility. When this is the case, `ovm-template-config` targets are presented to `ovmd` as `--script` parameters:

```
# ovmd -s cleanup
# ovmd -s configure
```

Important

When performing remote configuration, using `ovmd` to process messages containing configuration keys, the *authentication* module must be enabled. Processing of messages can only be completed if the final message contains the root user password. See Section 8.8.2, “Enabling and Disabling Configuration Modules (ovm-chkconfig)” for more information on this module.

See Section 8.8.4, “Triggering Configuration Changes” for more information on calling this script directly.

Syntax

`ovm-template-config` [`-e` | `--enumerate`] [`--human-readable`] [{ `-i` | `--input=` } *`input`*] [{ `-o` | `--output=` } *`output`*] [`--stdin`] [`--console-input`] [`--ovf-transport-iso`] [{ `-s` | `--script=` } *`script`*] [`--logfile=` *`logfile`*] [`--loglevel=` *`loglevel`*] [`--version`] [{ `-h` | `--help` }] *`target`*

Where *`target`* is:

{ `configure` | `unconfigure` | `reconfigure` | `cleanup` | `suspend` | `resume` | `migrate` | `shutdown` }

Options

The following table shows the available options for this command.

Option	Description
[`-e` | `--enumerate`]	Enumerate the parameters for *`target`*.
`--human-readable`	Print in human readable format when enumerate parameters.
{ `-i` | `--input=` } *`input`*	Input parameters from this file descriptor.
{ `-o` | `--output=` } *`output`*	Output parameters to this file descriptor.
`--stdin`	Build parameters from `stdin`.
`--console-input`	Build parameters from the console input.
`--ovf-transport-iso`	Build parameters from the OVF transport ISO.
{ `-s` | `--script=` } *`script`*	Specify a script.
`--logfile=` *`logfile`*	Set the name of the log file.
`--loglevel=` *`loglevel`*	Set the logging level.
`--version`	Show the `ovm-template-config` script version number and exit.
{ `-h` | `--help` }	Show help on the `ovm-template-config` command options.

Examples

Example 8.8 Listing the key pairs in configuration modules

```
# ovm-template-config --enumerate configure
```

Example 8.9 Listing the key pairs specific to the network configuration module

```
# ovm-template-config --enumerate --script network configure
```

Example 8.10 Passing configuration information to the script from STDIN

```
# ovm-template-config --stdin configure
```

Example 8.11 Passing configuration information from the command line prompt (prompting for values)

```
# ovm-template-config --console-input configure
```

Example 8.12 Passing configuration information from an OVF transport mounted on a CDROM device

```
# ovm-template-config --ovf-transport-iso configure
```

8.8.2 Enabling and Disabling Configuration Modules (ovm-chkconfig)

When a module is enabled, symlinks to the module script are made to other subdirectories within `/etc/template.d` based on the type of target the module provides, in much the same way that the System V `init` process works. When a module gets added, the header of the module script is read to verify the name, priority and targets and then a symlink is made to the corresponding subdirectories under `/etc/template.d`.

Enabling and disabling targets for any module is handled using the `ovm-chkconfig` script. Usage of this command is outlined using the `--help` parameter.

Syntax

`ovm-chkconfig` [`--list` [*name*]] [`--add` *name*] [`--del` *name*] [`--target=` *target* ... *name* { `on` | `off` }] [`--version`] [{ `-h` | `--help` }]

Where *target* is:

{ `configure` | `unconfigure` | `reconfigure` | `cleanup` | `suspend` | `resume` | `migrate` | `shutdown` }

Options

The following table shows the available options for this command.

Option	Description
[`--list` [*name*]]	Lists the status of the script *name*.
[`--add` *name*]	Add a new script *name*.
[`--del` *name*]	Delete a script *name*.
[`--target=` *target* ... *name* { `on` | `off` }]	Specify the *target*s in a comma separated list, for example: `--target="configure,unconfigure"`
`--version`	Show the `ovm-chkconfig` script version number and exit.
{ `-h` | `--help` }	Show help on the `ovm-chkconfig` command options.

Examples

Example 8.13 Listing available modules and their target runtime status

```
# ovm-chkconfig --list
name            configure unconfigure reconfigure cleanup suspend resume migrate shutdown
authentication  on:90     off         off         off     off     off    off     off
datetime        on:50     off         off         off     off     off    off     off
firewall        on:41     off         off         off     off     off    off     off
network         off       off         off         off     off     off    off     off
selinux         off       off         off         off     off     off    off     off
ssh             off       off         off         off     off     off    off     off
system          off       off         off         off     off     off    off     off
user            off       off         off         off     off     off    off     off
```

Example 8.14 Enabling all targets supported by a module

```
# ovm-chkconfig --add authentication
```

Example 8.15 Disabling all targets supported by a module

```
# ovm-chkconfig --del datetime
```

Example 8.16 Disabling particular targets for a module

```
# ovm-chkconfig --target=cleanup user off
```

8.8.3 Key-Value Pairs Used By Available Configuration Modules

To obtain a full listing of all of the key pairs that are used to trigger configuration changes through the `ovm-template-config` configuration modules, run the following command on the guest system where `ovm-template-config` is installed:

```
# ovm-template-config --human-readable --enumerate configure
```

The output from this command is printed as a Python data structure, that is easy to parse and understand. Content can be limited to the information specific to a configuration module by using the `--script` parameter as presented below:

```
# ovm-template-config --human-readable --enumerate configure --script datetime
        [('50',
  'datetime',
  [{u'description': u'System date and time in format year-month-day-hour-minute-second,
                     e.g., "2011-4-7-9-2-42".',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.datetime'},
   {u'description': u'System time zone, e.g., "America/New_York".',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.timezone'},
   {u'description': u'Whether to keep hardware clock in UTC: True or False.',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.utc'},
   {u'description': u'Whether to enable NTP service: True or False.',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.ntp'},
   {u'description': u'NTP servers separated by comma, e.g.,
                     "time.example.com,0.example.pool.ntp.org".',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.ntp-servers'},
   {u'description': u'Whether to enable NTP local time source: True or False.',
    u'hidden': True,
    u'key': u'com.oracle.linux.datetime.ntp-local-time-source'}])]
```

From the output, it becomes clear as to which configuration modules are triggered at which runlevel and what keys and values they accept. Note that all key names are structured so that they are prefixed with *com.oracle*, to avoid conflicts with any custom modules that you may intend to develop for your own purposes.

Key value pairs are actually passed to `ovm-template-config` in JSON format:

```
{"com.oracle.linux.datetime.ntp":"True"}
{"com.oracle.linux.datetime.ntp-servers":"0.pool.ntp.org,1.pool.ntp.org,2.pool.ntp.org"}
{"com.oracle.linux.root-password":"mysecret"}
```

8.8.4 Triggering Configuration Changes

There are a variety of approaches that can be used to trigger a configuration change using `ovm-template-config`. Most commonly, this is done by setting the `ovmd` service to run using in *enable-initial-config* mode. This causes the virtual machine to wait to be provided with configuration parameters either on via the console, or via the `ovmd` messaging facility, after the next boot. This is the usual approach when configuring a virtual machine to act as a template.

To manually force ovmd to pass messages to the ovm-template-config script, simply run ovmd with the --script parameter set to point to one of the ovm-template-config targets:

```
# ovmd --list
{"com.oracle.linux.datetime.ntp":"True"}
{"com.oracle.linux.datetime.ntp-servers":"0.pool.ntp.org,1.pool.ntp.org,2.pool.ntp.org"}
{"com.oracle.linux.root-password":"password"}
# ovmd -s configure
```

Note

To perform an action using messaging parameters and ovm-template-config, the *authentication* module must be enabled and your final message should include an authentication request in the form of a key-value message:

```
{"com.oracle.linux.root-password":"password"}
```

When running ovmd like this, ovmd prepares two pipes (*infd* and *outfd*) and calls the ovm-template-config script transparently in the background in the following way:

```
# ovm-template-config --input <infd> --output <outfd> configure
```

During testing, it may be useful to simply pass configuration information directly to the script from STDIN on the command line. This can be achieved by calling the script directly with the --stdin parameter:

```
# ovm-template-config --stdin configure <<EOF
> {"com.oracle.linux.selinux.mode": "disabled"}
> {"com.oracle.linux.root-password": "ovsroot"}
> EOF
```

Configuration can also be achieved directly from the console by running the script using the --console-input option. Doing this will prompt you for values for each of the keys that need to be defined for any enabled modules:

```
# ovm-template-config --console-input configure
```

8.8.5 Developing Oracle VM Template Configuration Modules

The provided module scripts are developed in Python. Theoretically, it is possible to develop module scripts in a different language, as long as the input, output and argument handling remains the same. The example provided in this section makes use of the Python programming language.

Each module script consists of two main parts:

1. The script header, which contains information like script name, targets, priorities and description.

2. The actual script, which handles a small set of parameters.

For examples of functional module scripts, refer to the existing modules in the /etc/template.d/scripts directory.

Module Script Header

Module script headers require a very specific comment block in order for ovm-chkconfig to handle enabling and disabling your script functionality. The format for the script header is as follows:

```
### BEGIN PLUGIN INFO
# name: [script name]
# [target]: [priority]
# [target]: [priority]
# description: a description that can
#   cross multiple lines.
### END PLUGIN INFO
```

When developing your own module script, you must include a header following the exact same format. Provide your own script name, which will be used when calling `ovm-chkconfig`, the targets that your script will support, and the priority for your script. The priority will specify in what order the script gets executed. You do not have to implement all targets. If you have a configure target but no cleanup target, this is still acceptable. The configure target gets called when a first boot/initial start of the virtual machine happens. The cleanup target happens when you manually initiate a cleanup in your virtual machine or when you want to restore the virtual machine to its original state. An example of the network module script header is provided below:

```
### BEGIN PLUGIN INFO
# name: network
# configure: 50
# cleanup: 50
# description: Script to configure template network.
### END PLUGIN INFO
```

Module Script Body

The main requirement for the module script body is that it accepts at least one target parameter. Target parameters that might get presented by the `ovm-template-configure` script include:

- `configure`
- `unconfigure`
- `reconfigure`
- `cleanup`
- `suspend`
- `resume`
- `migrate`
- `shutdown`

Your script can handle any other arguments that you require. There is one optional parameter which is useful to implement and this is `-e` or `--enumerate`. `ovm-template-config` uses this to be able to enumerate or list the parameters for a target supported by your script.

A very basic template to use for your script body follows:

```
try:
    import json
except ImportError:
    import simplejson as json
from templateconfig.cli import main


def do_enumerate(target):
    param = []
    if target == 'configure':
        param += []
    elif target == 'cleanup':
        param += []
    return json.dumps(param)
```

```


def do_configure(param):
    param = json.loads(param)
    return json.dumps(param)


def do_cleanup(param):
    param = json.loads(param)
    return json.dumps(param)


if __name__ == '__main__':
    main(do_enumerate, {'configure': do_configure, 'cleanup': do_cleanup})
```

This script supports the configure and cleanup targets.

You can fill out the script with your own code. For instance, for the `do_enumerate` function, you would populate the parameters that are supported for each target in the script. An example from the firewall module is presented below:

```
def do_enumerate(target):
    param = []
    if target == 'configure':
        param += [{'key': 'com.oracle.linux.network.firewall',
                   'description': 'Whether to enable network firewall: True or False.',
                   'hidden': True}]
    return json.dumps(param)
```

Each target function begins by reading the JSON parameters passed to the script, using the `param = json.loads(param)` statement. From this point, code can be written to perform actions based on the values of the keys that the script expects to receive. Once again, the example provided below is from the firewall module:

```
def do_configure(param):
    param = json.loads(param)
    firewall = param.get('com.oracle.linux.network.firewall')
    if firewall == 'True':
        shell_cmd('service iptables start')
        shell_cmd('service ip6tables start')
        shell_cmd('chkconfig --level 2345 iptables on')
        shell_cmd('chkconfig --level 2345 ip6tables on')
    elif firewall == 'False':
        shell_cmd('service iptables stop')
        shell_cmd('service ip6tables stop')
        shell_cmd('chkconfig --level 2345 iptables off')
        shell_cmd('chkconfig --level 2345 ip6tables off')
    return json.dumps(param)
```

Module Script Packaging

Once you have written one or more configuration module scripts, you may want to package them as RPMs that can be deployed on other systems. In order to install and configure template configure scripts, they have to be packaged in an RPM, with a specific naming convention. Package the script as `ovm-template-config-[scriptname]`. Ideally in the post install of the RPM you should add the script automatically by executing `# /usr/sbin/ovm-chkconfig --add scriptname`. When uninstalling a script/RPM, remove it at uninstall time using `# /usr/sbin/ovm-chkconfig --del scriptname`. This is illustrated in the following example of an RPM spec file that can be used:

```
Name: ovm-template-config-example
Version: 3.0
Release: 1%{?dist}
Summary: Oracle VM template example configuration script.
```

```
Group: Applications/System
License: GPL
URL: https://www.oracle.com/virtualization
Source0: %{name}-%{version}.tar.gz
BuildRoot: %(mktemp -ud %{_tmppath}/%{name}-%{version}-%{release}-XXXXXX)
BuildArch: noarch
Requires: ovm-template-config

%description
Oracle VM template example configuration script.

%prep
%setup -q

%install
rm -rf $RPM_BUILD_ROOT
make install DESTDIR=$RPM_BUILD_ROOT

%clean
rm -rf $RPM_BUILD_ROOT

%post
if [ $1 = 1 ]; then
    /usr/sbin/ovm-chkconfig --add example
fi

%preun
if [ $1 = 0 ]; then
    /usr/sbin/ovm-chkconfig --del example
fi

%files
%defattr(-,root,root,-)
%{_sysconfdir}/template.d/scripts/example

%changelog
* Tue Mar 22 2011 John Smith  - 3.0-1
- Initial build.
```

Edit the example spec file to reference your own script name.

In order to create RPMs, you must install `rpmbuild`:

```
# yum install rpm-build
```

The following is an example Makefile that you might assist in automating the build process:

You must edit this Makefile to reference your own script.

```
DESTDIR=
PACKAGE=ovm-template-config-example
VERSION=3.0

help:
@echo 'Commonly used make targets:'
@echo '  install    - install program'
@echo '  dist       - create a source tarball'
@echo '  rpm        - build RPM packages'
@echo '  clean      - remove files created by other targets'

dist: clean
mkdir $(PACKAGE)-$(VERSION)
tar -cSp --to-stdout --exclude .svn --exclude .hg --exclude .hgignore \
--exclude $(PACKAGE)-$(VERSION) * | tar -x -C $(PACKAGE)-$(VERSION)
tar -czSpf $(PACKAGE)-$(VERSION).tar.gz $(PACKAGE)-$(VERSION)
rm -rf $(PACKAGE)-$(VERSION)
```

```
install:
install -D example $(DESTDIR)/etc/template.d/scripts/example

rpm: dist
rpmbuild -ta $(PACKAGE)-$(VERSION).tar.gz

clean:
rm -fr $(PACKAGE)-$(VERSION)
find . -name '*.py[cdo]' -exec rm -f '{}' ';'
rm -f *.tar.gz

.PHONY: dist install rpm clean
```

Create a working directory, copy over your script, the spec file and the Makefile. Run the following command to create a src tarball of your code:

```
# make dist
```

Run the following command to generate an RPM file:

```
# make rpm
```

The preceding command generates an RPM in the `RPMS/noarch` directory within your working directory, for example: `RPMS/noarch/ovm-template-config-test-3.0-1.el6.noarch.rpm`.

Chapter 9 Using the Oracle VM Utilities

The Oracle VM Utilities are a collection of command line scripts that allow you to perform certain tasks in an Oracle VM environment, such as collecting metrics about the Oracle VM Server host on which virtual machines run and configuring CPU pinning, which is also referred to as hard partitioning.

The Oracle VM Utilities communicate directly with Oracle VM Manager using a Oracle VM API. The Oracle VM Utilities connect to the Oracle VM API with an Oracle VM Manager administrative user name and password. Oracle VM Manager listens for the Oracle VM Utilities on port 7002 (HTTPS).

Note

- The Oracle VM Utilities version 2.1 and later work with Oracle VM Manager Release 3.4. Additionally, only the `ovm_vmdisks`, `ovm_vmhostd`, `vm_dump_metrics`, and `ovm_vmcontrol` scripts are intended for use with Oracle VM Manager Release 3.4. All other scripts are either obsolete or deprecated by the Oracle VM Manager Command Line Interface. See the *Oracle VM Manager Command Line Interface User's Guide* for details on how you can perform management tasks in an Oracle VM environment.
- The Oracle VM Utilities are provided as-is and are not supported by Oracle. However, Oracle provides support for the Oracle VM Utilities in the following cases:
  - The `ovm_vmhostd` and `vm_dump_metrics` scripts are supported when running SAP applications on Oracle Linux guests in an Oracle VM environment.
  - The `ovm_vmcontrol` script is supported only when configuring *hard partitioning*, which is also referred to as CPU pinning. For more information, see the *Setting Hard Partitioning for Virtual Machines CPUs* section in the *Oracle VM Concepts Guide* .

Downloading and Installing the Oracle VM Utilities

To install the Oracle VM Utilities, do the following:

1. Download the Oracle VM Utilities as a `.zip` file from the Oracle VM downloads page:

   https://www.oracle.com/technetwork/server-storage/vm/downloads/index.html

2. Extract the contents of the `.zip` file to the system. Refer to the `readme` file for specific instructions.

9.1 Oracle VM Virtual Machine Control (ovm_vmcontrol)

The `ovm_vmcontrol` script lets you configure CPU pinning, which is also referred to as hard partitioning, on *virtual machines*.

Note

- The `ovm_vmcontrol` script supports CPU pinning for virtual machines running on x86-based Oracle VM Servers only. You cannot configure CPU pinning for virtual machines running on Oracle VM Server for SPARC.
- If you are using Oracle VM Release 3.4.1 or Release 3.4.2, after you configure CPU pinning for a virtual machine, you must stop and then start the virtual

machine for the configuration to take effect. Restarting the virtual machine does not load the configuration changes for CPU pinning.

- As of Oracle VM Release 3.4.3, support for dynamic CPU pinning is available. As a result, after you configure CPU pinning for a virtual machine, there is no longer a requirement to stop and then start the virtual machine for the configuration to take effect.

Syntax

`ovm_vmcontrol` { `-u` *`username`* } [`-p` *`password`* | `-E`] { `-h` *`hostname`* } { `-c` *`command`* } { `-v` *`vm_name`* | `-U` *`vm_uuid`* } [`-s` *`cpu_thread_list`* ...]

Where *`command`* is:

{ `setvcpu` | `getvcpu` | `rmvcpu` }

Options

The following table shows the available options for this command.

Option	Description
`-u` *`username`*	Username of an Oracle VM Manager admin user. This option is required.
[`-p` *`password`* | `-E`]	Corresponding password for the Oracle VM Manager admin user. You can specify the password as follows: • Do not set an option for the password in the command line. The Oracle VM Utilities then prompt you to specify it. You should use this method to set the password. • Set the password as an environment variable and then use the `-E` option. It is best practice to remove the environment variable when it is no longer required. You must use the `OVMUTIL_PASS` environment variable with the `-E` option. Set this variable for a single session as follows: `# export OVMUTIL_PASS=password` • Specify the password on the command line with the `-p` option. This option is deprecated as of Oracle VM Release 3.4.
{ `-h` *`hostname`* }	Hostname of the server running Oracle VM Manager.
`-c` { `setvcpu` | `getvcpu` | `rmvcpu` }	Command to execute. This option is required. The `setvcpu` command hard-binds or *pins* virtual CPUs to threads. For example, `-c setvcpu -s 0,1,2` physically binds vcpu0 to thread0, vcpu1 to thread1, vcpu2 to thread2. Use the `getvcpu` command to retrieve information about pinned virtual CPUs for the selected virtual machine. CPU binding immediately takes effect, and will continue on subsequent start ups of the virtual machine.

Option	Description
`-v` *`vm_name`*	Virtual machine name.
`-U` *`vm_uuid`*	Virtual machine UUID. If you do not specify the virtual machine name with the `-v` *`vm_name`* option, you must specify the UUID of the virtual machine.
`-s` *`cpu_thread_list`* ...	List of physical thread numbers to which you can bind virtual CPUs. You can set the value as follows: • One or more numbers separated by a comma. For example, `0,1,2,3,4`. • A range of numbers. For example, `0-4` is equal to `0,1,2,3,4`. • A range of numbers that excludes one number. For example, `0-4,^2` is equal to `0,1,3,4`. • A combination of comma-separated numbers and ranges. For example, `0-2,4,6-8` is equal to `0,1,2,4,6,7,8`. **Note** You must not set a number as the value for the `-s` option that is greater than the physical CPU count for the server. For example, if a server has 8 physical CPUs then you can set 0 to 7. Setting 8 or higher is not valid. You must use this command in combination with: `-c setvcpu`

Examples

Example 9.1 Setting CPU pinning for a virtual machine

This example binds the virtual CPUs of the virtual machine to threads 0, 1, 3, 5, and 7.

```
# ./ovm_vmcontrol -u admin -h localhost -v MyVM01 -c setvcpu -s 0-3,^2,5-7,^6
Oracle VM VM Control utility 2.1.
Connecting to OVM Manager using Web Service.
Connected.
OVM Manager version: version
Command : setvcpu
Pinning vCPU '0-3,^2,5-7,^6' to VM 'MyVM01'
Pin vCPU succeed.
```

Example 9.2 Checking CPU pinning for a virtual machine

This example shows the virtual CPUs of the virtual machine are bound to threads 0, 1, 3, 5, and 7.

```
# ./ovm_vmcontrol -u admin -h localhost -v MyVM01 -c getvcpu
Oracle VM VM Control utility 2.1.
Connecting to OVM Manager using Web Service.
Connected.
OVM Manager version: version
Command : getvcpu
```

```
Getting pinned CPU list...
Current pinned CPU:0-3,^2,5-7,^6
```

9.2 Oracle VM Retrieve Disk (ovm_vmdisks)

The `ovm_vmdisks` script retrieves details about *virtual disks*. This script lists every virtual disk for a given virtual machine, as well as the virtual machine configuration (`vm.cfg`) file. If the virtual disks are physical devices directly attached to the virtual machine, this script lists the device mapper entry on the Oracle VM Server to which the virtual machine is assigned. If the virtual disks are files on an NFS server, the utility lists the NFS server name, mount point, and file name and location.

Syntax

```
ovm_vmdisks{-u username}{-p password|-E }{-h hostname}{-v vm_name|-U vm_uuid
}
```

Options

The following table shows the available options for this command.

Option	Description
`-u` *`username`*	Username of an Oracle VM Manager admin user. This option is required.
[`-p` *`password`*|`-E`]	Corresponding password for the Oracle VM Manager admin user. You can specify the password as follows: • Do not set an option for the password in the command line. The Oracle VM Utilities then prompt you to specify it. You should use this method to set the password. • Set the password as an environment variable and then use the `-E` option. It is best practice to remove the environment variable when it is no longer required. You must use the `OVMUTIL_PASS` environment variable with the `-E` option. Set this variable for a single session as follows: `# export OVMUTIL_PASS=password` • Specify the password on the command line with the `-p` option. This option is deprecated as of Oracle VM Release 3.4.
{`-h` *`hostname`*}	Hostname of the server running Oracle VM Manager.
`-v` *`vm_name`*	Virtual machine name.
`-U` *`vm_uuid`*	Virtual machine UUID. If you do not specify the virtual machine name with the `-v` *`vm_name`* option, you must specify the UUID of the virtual machine.

Examples

Example 9.3 Listing virtual disks for a virtual machine

```
# ./ovm_vmdisks -u admin -p Welcome1 -h localhost -v MyVM01
```

```
Oracle VM Retrieve Disk utility 2.1.
Connecting to OVM Manager using Web Service.
Connected.
Virtual Machine : 'MyVM01'
Assigned Server : OVS_01
Virtual Disk : '0004fb00001200003a3384c82332cce4.img' size : 50GB
    repository='MyRepo'
    Mounted Path=/OVS/Repositories/ \
    0004fb0000030000c96714fda5ef202f/ \
    VirtualDisks/0004fb00001200003a3384c82332cce4.img
    Absolute Path=IP_address:/ \
    nfs/mypath/sp1/VirtualDisks/0004fb00001200003a3384c82332cce4.img
Config File :
   Mounted Path=/OVS/Repositories/ \
   0004fb0000030000c96714fda5ef202f/ \
   VirtualMachines/0004fb000006000097e4d197a07005d9/vm.cfg
   Absolute Path=IP_address:/ \
   nfs/mypath/sp1/VirtualMachines/0004fb000006000097e4d197a07005d9/vm.cfg
```

9.3 Oracle VM Hostd For Metrics Messaging (ovm_vmhostd)

The `ovm_vmhostd` script collects metrics about the Oracle VM Server host on which a virtual machine is running. Every 60 seconds, the script checks that the virtual machine is running and then sends the metrics about the host to the virtual machine itself. In other words, the same virtual machine sends and receives the metrics about the Oracle VM Server host.

The virtual machine message that the `ovm_vmhostd` script sends has a key of `vmhost`. The metrics about the Oracle VM Server host are defined as the value of the virtual machine message.

To retrieve the host metrics from the virtual machine, you can use either the Oracle VM Guest Additions daemon, `ovmd`, or the `vm-dump-metrics` script. See Section 9.4, “Retrieving Host Metrics from Virtual Machines (vm-dump-metrics)”.

> **Note**
>
> The `ovm_vmhostd` script uses the Oracle VM Guest Additions to send and receive the metrics as virtual machine messages. For this reason, the guest virtual machine must be running Oracle Linux with the Oracle VM Guest Additions installed. See Chapter 8, *Installing and Using the Oracle VM Guest Additions*.

Syntax

`ovm_vmhostd` { `-u` *username* } [`-p` *password* | `-E`] { `-h` *hostname* } { `-v` *vm_name* | `-U` *vm_uuid* }

Options

The following table shows the available options for this command.

Option	Description
`-u` *username*	Username of an Oracle VM Manager admin user. This option is required.
[`-p` *password* | `-E`]	Corresponding password for the Oracle VM Manager admin user. You can specify the password as follows: • Do not set an option for the password in the command line. The Oracle VM Utilities then prompt you to specify it. You should use this method to set the password.

Option	Description
	- Set the password as an environment variable and then use the `-E` option. It is best practice to remove the environment variable when it is no longer required. You must use the `OVMUTIL_PASS` environment variable with the `-E` option. Set this variable for a single session as follows: `# export OVMUTIL_PASS=password` - Specify the password on the command line with the `-p` option. This option is deprecated as of Oracle VM Release 3.4.
{`-h` *`hostname`*}	Hostname of the server running Oracle VM Manager.
`-v` *`vm_name`*	Virtual machine name. The virtual machine name is the name you assign during the creation of the virtual machine.
`-U` *`vm_uuid`*	Virtual machine UUID. If you do not specify the virtual machine name with the `-v` *`vm_name`* option, you must specify the UUID of the virtual machine.

Examples

Example 9.4 Sending Oracle VM Server details to a virtual machine

```
# ./ovm_vmhostd -u admin -h localhost -v MyVM01
Oracle VM Hostd 2.1.
Connecting to OVM Manager using Web Service.
Connected.
Processing VM : 'MyVM01'

VM : 'MyVM01' has status :  Running.
Message sent.
Sleeping 60 seconds.
Sleeping 60 seconds.
Sleeping 60 seconds.
```

9.4 Retrieving Host Metrics from Virtual Machines (vm-dump-metrics)

The `vm-dump-metrics` script outputs the metrics about the Oracle VM Server host in XML format that an SAP application running on a virtual machine guest can consume.

To collect the host metrics, do the following:

1. Run the `ovm_vmhostd` script to send metrics about the Oracle VM Server host as a virtual machine message.

2. Copy the `vm-dump-metrics` script to the guest virtual machine that received the virtual machine message.

3. Run the `vm-dump-metrics` script. The `vm-dump-metrics` script does not take any commands or options.

The `vm-dump-metrics` script queries `ovmd` to retrieve a message with the `vmhost` key. The script then does one of the following:

- Parses the message and outputs the XML to standard output (stdout).
- Exits with status `1` if a message with the `vmhost` key does not exist.

The following is an example of XML output from the `vm-dump-metrics` script:

```
<metrics>
  <metric type='real64' context='host'>
    <name>TotalCPUTime</name>
    <value>2694.3596</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>PagedOutMemory</name>
    <value>0</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>PagedInMemory</name>
    <value>0</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>UsedVirtualMemory</name>
    <value>6747</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>FreeVirtualMemory</name>
    <value>9817</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>FreePhysicalMemory</name>
    <value>9817</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>MemoryAllocatedToVirtualServers</name>
    <value>6747</value>
  </metric>
  <metric type='uint32' context='host'>
    <name>NumberOfPhysicalCPUs</name>
    <value>4</value>
  </metric>
  <metric type='string' context='host'>
    <name>HostSystemInfo</name>
    <value>ovm3</value>
  </metric>
  <metric type='string' context='host'>
    <name>VirtProductInfo</name>
    <value>Oracle VM 3</value>
  </metric>
  <metric type='string' context='host'>
    <name>VirtualizationVendor</name>
    <value>Oracle Corporation</value>
  </metric>
  <metric type='uint64' context='host'>
    <name>Time</name>
    <value>1360606566774</value>
  </metric>
  <metric type='string' context='host'>
    <name>HostName</name>
    <value>ovm3</value>
  </metric>
  <metric type='uint64' context='vm' id='0' uuid='0004fb00-0006-0000-d72b-647e20a85939'>
    <name>PhysicalMemoryAllocatedToVirtualSystem</name>
    <value>1024</value>
  </metric>
  <metric type='uint64' context='vm' id='0' uuid='0004fb00-0006-0000-d72b-647e20a85939'>
    <name>ResourceMemoryLimit</name>
    <value>1024</value>
  </metric>
  <metric type='uint32' context='vm' id='0' uuid='0004fb00-0006-0000-d72b-647e20a85939'>
    <name>ResourceProcessorLimit</name>
    <value>1</value>
  </metric>
  <metric type='real64' context='vm' id='0' uuid='0004fb00-0006-0000-d72b-647e20a85939'>
    <name>TotalCPUTime</name>
    <value>2694.3596</value>
  </metric>
</metrics>
```

Querying the Oracle VM Guest Additions Daemon Directly

As an alternative to using the `vm-dump-metrics` script, you can query `ovmd` to view the message that contains the metrics about the Oracle VM Server host, as follows:

```
# ovmd -g vmhost
com.sap.host.VirtualizationVendor=Oracle Corporation;com.sap.host.VirtProductInfo=Oracle VM 3;
com.sap.host.PagedInMemory=0;com.sap.host.PagedOutMemory=0;com.sap.host.PageRates=0;
com.sap.vm.uuid=0004fb0000060000d72b647e20a85939;com.sap.host.HostName=ovm3;
com.sap.host.HostSystemInfo=ovm3;com.sap.host.NumberOfPhysicalCPUs=4;com.sap.host.NumCPUs=4;
com.sap.host.TotalPhyMem=16383;com.sap.host.UsedVirtualMemory=6747;
com.sap.host.MemoryAllocatedToVirtualServers=6747;com.sap.host.FreeVirtualMemory=9817;
com.sap.host.FreePhysicalMemory=9817;com.sap.host.TotalCPUTime=381175.97;
com.sap.host.Time=1360606887997;com.sap.vm.PhysicalMemoryAllocatedToVirtualSystem=1024;
com.sap.vm.ResourceMemoryLimit=1024;com.sap.vm.TotalCPUTime=2696.2214;
com.sap.vm.ResourceProcessorLimit=1;
```

For more information about the Oracle VM Guest Additions daemon, `ovmd`, see Chapter 8, *Installing and Using the Oracle VM Guest Additions*.

Chapter 10 Converting Physical Hosts to Virtual Machines

Converting hosts involves creating *hardware virtualized* *guest* images from existing physical computers. This chapter explains how to use the Physical to Virtual (P2V) conversion utility to convert hosts.

10.1 Introduction to the P2V Utility

The P2V utility lets you convert the operating system and application software on a computer to an Oracle VM *hardware virtualized* *guest* image. The P2V utility is included on the *Oracle VM Server* ISO image.

Requirements for Using the P2V Utility

- The host operating system that you are converting must be supported as a guest operating system by Oracle VM. See the *Oracle VM Release Notes* for a list of the supported guest operating systems.
- The *host computer* must have a CPU that supports PAE (Physical Address Extension).
- The P2V utility included with Oracle VM Release 3.4 applies to 64-bit operating systems only.

Conversion Process Overview

You can run the P2V utility interactively or as an automated process using a kickstart configuration file.

To use the P2V utility, you boot from the *Oracle VM Server* ISO on the physical computer you want to convert to a hardware virtualized guest. The P2V utility then creates a *virtual machine* configuration file (`vm.cfg`) and creates raw *virtual disk* images from the disks on the computer.

The first four virtual disk images are created as IDE disks (hda, hdb, hdc, and hdd) on the guest, using the original disk names. Up to seven additional disks are created as SCSI devices (sda, sdb, sdc, and so on).

The following is an example of disk entries in the `vm.cfg` file:

```
disk = ['file:System-sda.img,hda,w',
'file:System-sdb.img,hdb,w',
'file:System-sdc.img,hdc,w',
'file:System-sdd.img,hdd,w',
'file:System-sde.img,sda,w',
'file:System-sdf.img,sdb,w',
'file:System-sdg.img,sdc,w',
'file:System-sdh.img,sdd,w',
'file:System-sdi.img,sde,w',
'file:System-sdj.img,sdf,w',
'file:System-sdk.img,sdg,w',
]
```

> **Note**
>
> Oracle VM Manager limits virtual machines with the HVM domain type to a maximum of four virtual disks. In this case, if the `vm.cfg` file has more than four disk entries, then Oracle VM Manager imports only the first four virtual disks.

At the end of the conversion process, the P2V utility starts a web server that hosts the `vm.cfg` file and virtual disk file(s). You can then import the hardware virtualized guest into Oracle VM Manager as a virtual machine template.

10.2 Converting Hosts in Interactive Mode

To convert the operating system and applications on a physical computer to a hardware virtualized guest with the P2V utility, complete the following steps:

Converting the Host with the P2V Utility

Before you begin, burn the Oracle VM Server ISO image to bootable physical media, such as a DVD-ROM.

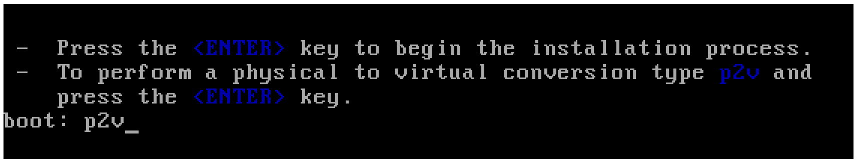
1. Insert the Oracle VM Server installation media into the optical disc drive of the computer you want to image.

2. Change the boot order to start from the optical disc drive in the BIOS or UEFI settings.

3. Boot the computer with the Oracle VM Server installation media.

   The Oracle VM Server boot screen displays.

4. At the `boot:` prompt, enter **`p2v`** and then press **Enter**.

   ```
   -  Press the <ENTER> key to begin the installation process.
   -  To perform a physical to virtual conversion type p2v and
      press the <ENTER> key.
   boot: p2v_
   ```

   The Oracle VM Server installer initializes. The first step in the conversion process is for the network manager to configure the network interface.

   - If the host has one network interface, the network manager automatically configures it.

   - If the host has two or more network interfaces, the network manager prompts you to select one network interface and then automatically configures it.

   - If the network manager cannot successfully configure the network interface, you are prompted to configure TCP/IP settings.

5. If prompted, configure the TCP/IP settings as appropriate.

Setting	Description
Enable IPv4 support	Enables Internet Protocol Version 4 (IPv4) in the network interface.
Dynamic IP configuration (DHCP)	Uses the Dynamic Host Configuration Protocol (DHCP) network service to automatically assign an IPv4 address to the hardware virtualized guest and retrieve DNS settings.
Manual configuration	Lets you specify the following: • IPv4 address for the hardware virtualized guest. • IP address of the gateway server for your network. • IP address of the name server for your network.
Enable IPv6 support	Enables Internet Protocol Version 6 (IPv6) in the network interface.
Automatic	Uses the DHCP network service to automatically assign an IPv6 address to the hardware virtualized guest and retrieve DNS settings.
Automatic, DHCP only	Uses the DHCP network service to automatically assign an IPv6 address for the hardware virtualized guest.
Manual configuration	Lets you specify the following: • IPv6 address for the hardware virtualized guest.

Setting	Description
	• IP address of the gateway server for your network. • IP address of the name server for your network.

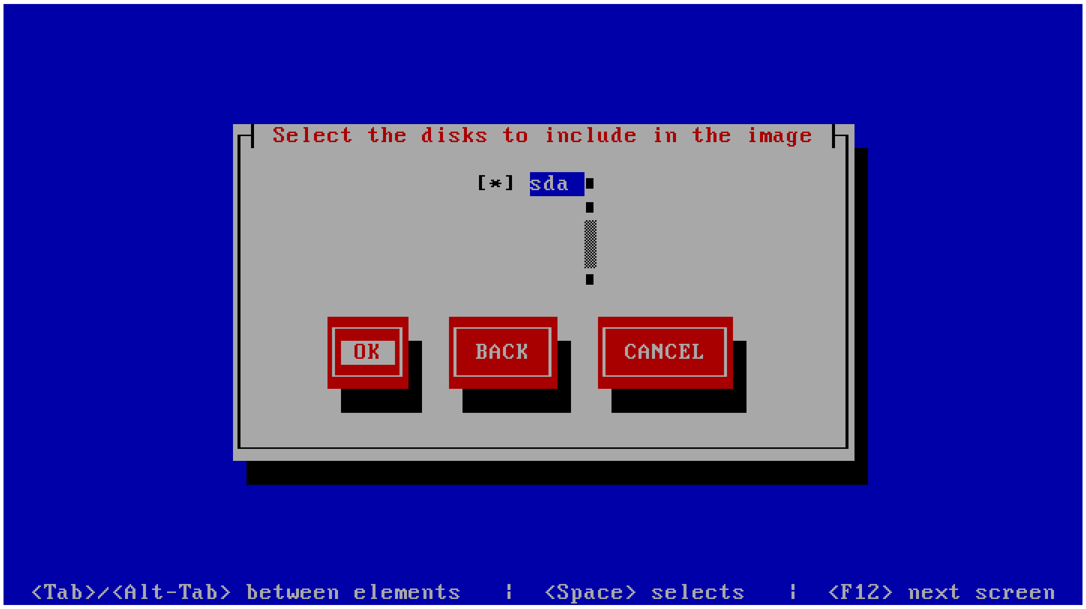
6. After the network interface is configured, you are prompted to test the media for errors. Do one of the following:

   - Select **OK** and then press **Enter** to test the bootable media for errors.

     > **Note**
     >
     > You must reboot after the test is complete.

   - Select **Skip** and press **Enter** to continue without testing the bootable media.

   The disk selection screen displays.

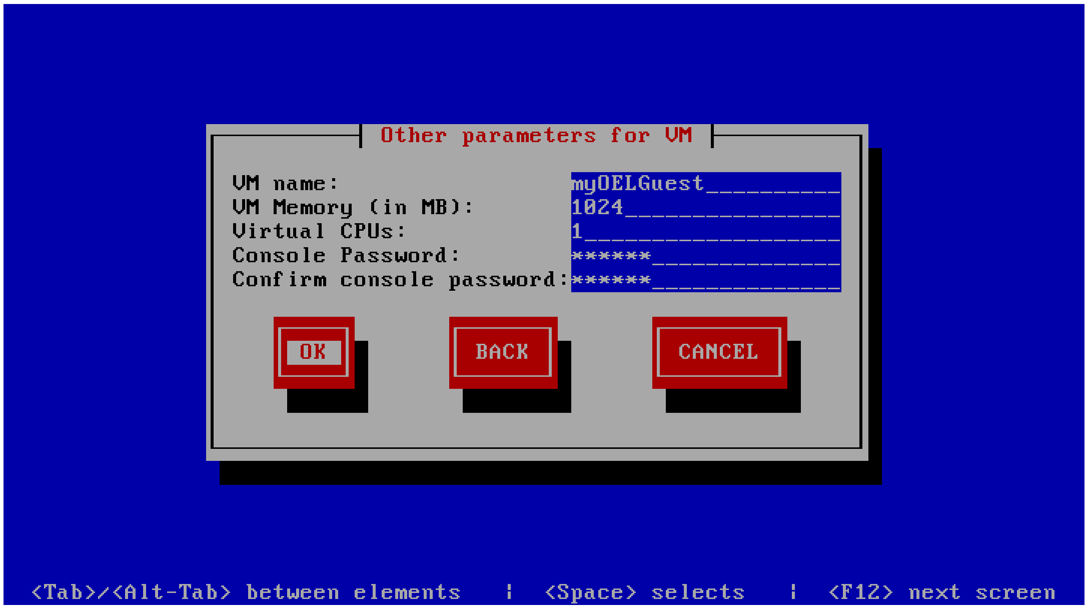


```
Select the disks to include in the image

[*] sda

<Tab>/<Alt-Tab> between elements   |  <Space> selects   |  <F12> next screen
```

7. Select the disk partition(s) on the computer to include in the guest image.

8. Select **OK** and then press **Enter**.

   The virtual machine parameters screen displays.



Other parameters for VM

```
VM name:                  myOELGuest___________
VM Memory (in MB):        1024_________________
Virtual CPUs:             1____________________
Console Password:         ******_______________
Confirm console password: ******_______________
```

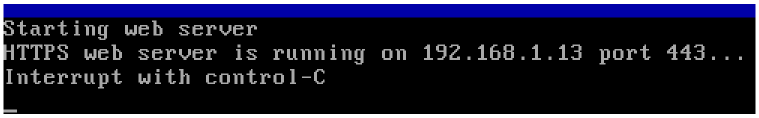
<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen

9. Specify values for each parameter as appropriate.

Parameter	Description
VM name	Specifies a name for the virtual machine.
VM Memory (in MB)	Specifies the amount of RAM, in megabytes, to allocate to the virtual machine when it is running. The amount of RAM that you specify will be requested from the host operating system. For this reason, it must be available or made available as free memory on the host when attempting to start the virtual machine and will not be available to the host while the virtual machine is running.
Virtual CPUs	Specifies the number of CPU cores to allocate to the virtual machine.
Console Password	Specifies a password for the virtual machine console.
Confirm console password	Verifies the console password.

10. Select **OK** and then press **Enter**.

A secure web server (HTTPS) starts. The IP address of the computer and port number where the web server is available displays.



```
Starting web server
HTTPS web server is running on 192.168.1.13 port 443...
Interrupt with control-C
_
```

You can now access the `vm.cfg` and virtual disk(s) that the P2V utility created on the web server. Proceed to Section 10.4, “Importing Guests as Virtual Machine Templates to Oracle VM Manager”.

10.3 Converting Hosts Silently with a Kickstart File

To convert physical hosts to a *guest* image silently, you use a kickstart file to pass parameters to the P2V utility.

The following steps provide a high-level overview of how to convert hosts silently:

1. Create a P2V kickstart file and copy it to your kickstart server.
2. Insert the Oracle VM Server installation media into the optical disc drive of the computer you want to image.
3. Change the boot order to start from the optical disc drive in the BIOS or UEFI settings.
4. Boot the computer with the Oracle VM Server installation media.

   The Oracle VM Server boot screen displays.
5. At the boot prompt, enter the following:

   ```
   # mboot.c32 xen.mb.efi --- vmlinuz p2v ksdevice=9C:B6:54:82:15:AC
   ks=http://example.com/mypath/ks.cfg --- initrd.img
   ```

   Where:

   - ksdevice specifies the network interface to use.
   - ks is the path to the P2V kickstart file.

The P2V utility begins the conversion process. If there are any missing parameters in the kickstart file, you are prompted to enter them.

When the conversion process completes, you can import the hardware virtualized guest into Oracle VM Manager as a virtual machine template. See Section 10.4, “Importing Guests as Virtual Machine Templates to Oracle VM Manager”.

Considerations for Using a Kickstart File

Before you attempt to convert hosts with a kickstart file, you should consider the following:

- The P2V utility converts disks on the computer to *virtual disk* images. The virtual disk images are created as IDE disks (hda, hdb, hdc, hdd, and so on) on the guest, using the original disk names. When you use a P2V kickstart file, up to four disks are automatically deployed in the guest. Any extra disks are converted and added to the guest configuration file (vm.cfg) but are not deployed in the guest.

  To deploy the additional disks in the guest, edit the guest configuration file to remove the comments from the disk entries and map the additional disks to SCSI device names; for example, sda, sdb, and sdc. The boot disk must always be mapped to device hda. Any files on the guest that contain references to these devices must also be changed; for example, the /etc/fstab file might contain references to /dev/hda1, /dev/sda1, and so on.
- At least one network interface must use DHCP so that the computer on which the P2V utility is running can read the kickstart file over the network.
- If you want the P2V utility web server to listen using a network interface other than the one that initiates the kickstart session, specify the network configuration for that network interface in the kickstart file.

P2V Kickstart Parameters

To converting hosts with a kickstart file, you must specify parameters at the boot prompt and in a P2V kickstart file.

At the boot prompt, you must specify parameters as follows:

```
# mboot.c32 xen.mb.efi --- vmlinuz p2v ksdevice=device ks=ksfile --- initrd.img
```

Boot Parameters

Option	Description
`p2v`	Loads the P2V conversion utility.
`ks=`*`ksfile`*	Specifies the name and location of a P2V kickstart file.
`ksdevice=`*`value`*	Specifies the network interface to use. You can set one of the following values: • The device name of the interface, for example: *`eth0`* • The MAC address of the interface, for example: *`9C:B6:54:82:15:AC`* • The keyword `link`, which uses the first active network interface found on the system. • The keyword `bootif`, which uses the interface that the system used to boot from a PXE server. • The keyword `ibft`, which uses the MAC address of the interface specified by the iSCSI Boot Firmware Table (iBFT) in the system BIOS or firmware.

Kickstart File Options

Option	Description
`p2v`	Indicates the kickstart file is intended to automate a P2V conversion. It accepts no parameters.
`target --ovmmanager`	Sets the end destination for the guest image. Sets the P2V utility to operate in HTTPS server mode to transfer the guest image to a running instance of *Oracle VM Manager*.
`diskimage` *`option`* ...	Denotes a disk to be included in the guest image. The P2V utility uses device mapper-based snapshotting to copy the disk as a `system-*.img` file on the target computer. There may be multiple `diskimage` directives in a P2V kickstart file, each resulting in a disk image in the guest image. The `--device` parameter must always be used with the `diskimage` directive to indicate which device should be imaged. The *`option`* parameter is one or more of the following: `--device` *`path`* The device to image. *`path`* must be the full path to the device. For example: `diskimage --device /dev/sda` `--type` { `IDE` | `SCSI` | `LVM` | `MDRAID` } Sets the type of disk. Must be one of `IDE`, `SCSI`, `LVM`, or `MDRAID`. Devices /dev/hda, /dev/hdb, /dev/hdc, and /dev/hdd should be `IDE`. Devices /dev/sd[a-zz] should be `SCSI`. A

Option	Description
	logical volume should be `LVM`. Devices /dev/md[a-zz] should be `MDRAID`. For example: `diskimage --device /dev/hda --type IDE`

`network` *`option`* ...	Configures network information for the computer.
	The *`option`* parameter is one or more of the following:
	`--bootproto` { `dhcp` | `bootp` | `static` }
	Sets the method by which the network configuration is determined. Must be `dhcp`, `bootp`, or `static`. The default is `dhcp`. `bootp` and `dhcp` are treated as the same.
	`dhcp` uses a DHCP server to obtain the networking configuration, for example:
	`network --bootproto dhcp`
	`static` requires all the necessary networking information. As the name implies, this information is static and is used during and after the installation. The entry for static networking is more complex, as you must include all network configuration information on one line. You must specify the IP address, netmask, gateway, and nameserver, for example:
	`network --bootproto static --ip 10.0.2.15`
	`  --netmask 255.255.255.0 --gateway 10.0.2.254`
	`  --nameserver 10.0.2.1`
	The `static` method has the following restrictions:
	• All static networking configuration information must be specified on one line; you cannot wrap lines using a backslash.
	• You can only specify one nameserver.
	`--ip` *`ipaddress`*
	The IP address for the computer.
	`--gateway` *`ipaddress`*
	The IP address for the default gateway.
	`--nameserver` *`ipaddress`*
	The IP address for the primary nameserver.
	`--netmask` *`netmask`*
	The netmask for the computer.

`vm_options` *`option`* ...	Sets the configuration options for the guest. The *`option`* parameter is one or more of the following:

Option	Description

	`--name name`
	The name of the guest.
	`--mem size`
	The memory allocation for the guest in Mb.
	`--vcpus number`
	The number of VCPUs for the guest.
	`--consolepasswd password`
	The console password for the guest. This option is ignored by Oracle VM Manager when the guest is imported.
	For example:
	`vm_options --name MyVM --mem 2048 --vcpus 2`
	`  --consolepasswd mypassword`

P2V Kickstart File Example

The following provides an example of a P2V kickstart file:

```
p2v
target --ovmmanager
network --device eth0 --bootproto dhcp
diskimage --device /dev/sda --type IDE
vm_options --name myVM --mem 2048 --vcpus 2 --consolepasswd password
```

10.4 Importing Guests as Virtual Machine Templates to Oracle VM Manager

After the P2V utility successfully converts a physical host to a hardware virtualized guest, you can import the guest as a virtual machine template into Oracle VM Manager.

You must ensure that the `vm.cfg` file and virtual disk(s) reside at a location where you can import them before you attempt to import the hardware virtualized guest as a virtual machine template,

The web server that hosts the `vm.cfg` and virtual disk(s) uses port 443. You can import the files directly into a Oracle VM repository with Oracle VM Manager. However, you must ensure that the Oracle VM Server that is designated as the administration server for that repository can access the web server on port 443. Alternatively, you can download the `vm.cfg` and virtual disk(s) from the web server and then host them at a location that Oracle VM Server can access.

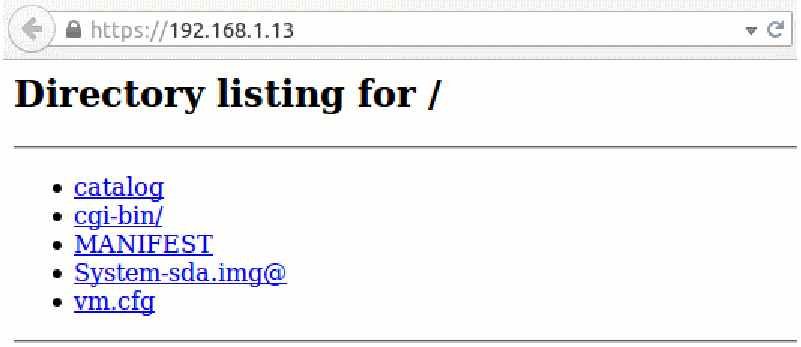
Note

The web server provides a serial HTTP connection. For this reason, multiple requests to import guests as virtual machine templates are issued sequentially. For example, if you start a second download while another download is in progress, the second download does not start until the first download is complete.

1. Open a browser and navigate to the location of the web server that is hosting the `vm.cfg` and virtual disk(s); for example,

https://192.168.1.13/

The browser displays a directory listing of the files that the P2V utility created.



https://192.168.1.13

Directory listing for /

- catalog
- cgi-bin/
- MANIFEST
- System-sda.img@
- vm.cfg

2. Copy the URLs for the `vm.cfg` file and `*.img` file(s).

3. Import the files into Oracle VM Manager as a virtual machine template.

   a. Log in to Oracle VM Manager.

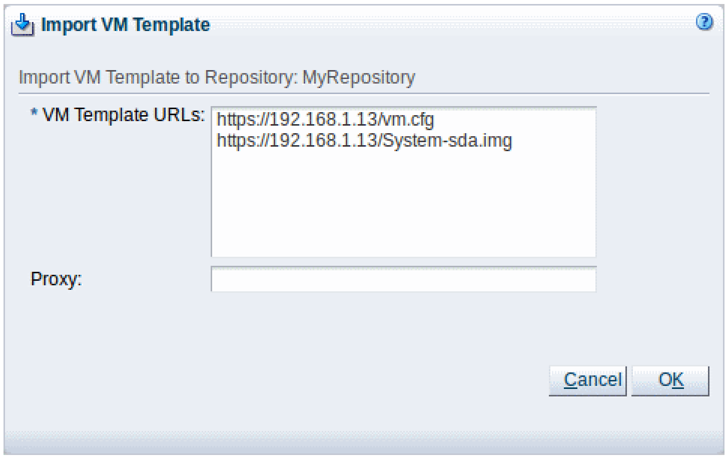
   b. Click the **Repositories** tab and then select the repository in which to store the template.

   c. Select **VM Templates** in the navigation tree.

   d. Select **Import VM Template** from the toolbar in the management pane.

   e. Specify the URLs for the `vm.cfg` file and `*.img` file(s) as separate lines in the **VM Template URLs pane**.

   f. Click **OK** to import the files.



Import VM Template

Import VM Template to Repository: MyRepository

* VM Template URLs:
https://192.168.1.13/vm.cfg
https://192.168.1.13/System-sda.img

Proxy:

Refer to the Oracle VM Manager Online Help for more information about importing virtual machine templates.

The hardware virtualized guest is available in the repository as a virtual machine template.

The hardware virtualized guest must have a unique network configuration. For this reason, you should configure a new network device and ensure that it is detected before you start the virtual machine that you created with the P2V utility. If you use the same network configuration as the original computer, a network clash might occur because two computers on the network have the same IP and MAC addresses.

After you successfully import the hardware virtualized guest as a virtual machine template, press **Control**+**C** to interrupt the P2V utility on the host computer. You can then remove the *Oracle VM Server* bootable media from the drive and restart the computer.

Chapter 11 Installing and Configuring the Oracle VM Exporter Appliance

The Oracle VM Exporter Appliance is a special type of virtual machine used to export another virtual machine from the Oracle VM environment to a tenancy account in Oracle Cloud Infrastructure. Before you can use the Oracle VM Exporter Appliance, in addition to a valid Oracle VM account, you need an active tenancy and user account in Oracle Cloud Infrastructure.

> **Note**
>
> SSH is disabled on the Oracle VM Exporter Appliance for security reasons. To log on to the Oracle VM Exporter Appliance, you have to use the console.

The Oracle VM Exporter Appliance uses Oracle Cloud Infrastructure APIs to perform the export. You need to upload the Oracle VM Exporter Appliance public key to Oracle Cloud Infrastructure to export a virtual machine.

For more information about uploading keys to Oracle Cloud Infrastructure, see https://docs.cloud.oracle.com/en-us/iaas/Content/API/Concepts/apisigningkey.htm#three.

Additionally, before using the Oracle VM Exporter Appliance, you need to know Oracle Cloud Infrastructure:

- Region
- Compartment
- Availability Domain
- Instance Shapes (and their quotas)

For more information about finding the above values, see https://docs.cloud.oracle.com/en-us/iaas/Content/API/Concepts/apisigningkey.htm.

For more information about exporting virtual machines and the parameters needed see the *Exporting Virtual Machines* section of the *Oracle VM Concepts Guide* .

11.1 Considerations

This section contains useful information to consider when using the Oracle VM Exporter Appliance.

Oracle VM Considerations

- You can export hardware virtual machines (HVMs) or paravirtual hardware virtual machines (PVHVM).
- You *cannot* export paravirtual virtual machines (PVMs).
- An Oracle VM usually has *Storage Networks* defined for storage like NFS, iSCSI, and so on. If virtual machines that are to be exported to Oracle Cloud Infrastructure have virtual disks accessed through Storage Networks (or other types of networks), the Oracle VM Storage Network's configuration must be modified to add virtual machine use.
- VNICs tied to Storage Networks are needed for accessing resources that can be accessed via one or more Storage Networks.

- If you need to access resources that can only be reached through one or more Storage Networks, then you must also attach a VNIC to each Storage Network in Oracle VM Exporter Appliance.

- If Oracle VM Exporter Appliance requires static IP address configuration, then you must configure these IP addresses on all VNICs for the virtual machines and storage networks.

- Once all of these parameters have been set successfully in the Oracle VM Exporter Appliance virtual machine, configuration is complete. For information on using the Oracle VM Exporter Appliance, see see the *Export a Virtual Machines* section of the *Oracle VM Manager User's Guide* .

Virtual Machine Specifics

- You can export any virtual machine listed in https://docs.cloud.oracle.com/en-us/iaas/Content/Compute/References/bringyourownimage.htm to Oracle Cloud Infrastructure

- You cannot export a `SOLARIS11U3_X86_PVHVM` virtual machine to Oracle Cloud Infrastructure.

Exporting Windows Virtual Machines

- You cannot export a Windows virtual machine to Oracle Cloud Infrastructure unless it has the latest VirtIO drivers. These drivers can be downloaded from the Oracle Software Delivery Cloud (OSDC) at https://edelivery.oracle.com.

NFS Server Considerations

- The Oracle VM Exporter Appliance needs read-only access to NFS shares of repositories that contain virtual machine resources. Modify the NFS export on the NFS server to export these resources to the Oracle VM Exporter Appliance IP address on the appropriate Storage Network.

- The Oracle VM Exporter Appliance needs an NFS Share with read-write permission to hold temporary, converted disk images. Create an NFS share on the NFS server and export it to the Oracle VM Exporter Appliance IP address on the appropriate Storage Network

LUN or Local Storage Considerations

- For repositories on LUN or local storage physical disks, check the firewall for NFS export from the Oracle VM Server host.

- Add "Repository Export" on one of the Oracle VM Server hosts on which the LUN or Local Disk exists. The repository must be presented to the Oracle VM Server host in order to export the repository to an IP address that reaches the Oracle VM Server host.

- If you have a Repository on a LUN or Local Storage (physical disk), the Oracle VM Exporter Appliance must have export access to that LUIN or Local Storage on the hosts or servers in the server pool that is presented to that LUN or Local Storage repository. To add export access in Oracle VM, highlight the host or server and change the **Perspective** to **Repository Exports**. Click on the **+** symbol to add the IP address of the Oracle VM Exporter Appliance virtual machine to the LUN or Local Storage Repository.

11.2 Installing the Oracle VM Exporter Appliance

Before exporting a virtual machine to Oracle Cloud Infrastructure with the Oracle VM Exporter Appliance, you must download the Oracle VM Exporter Appliance Open Virtualization Appliance (OVA), create the Oracle VM Exporter Appliance virtual machine, and configure the Oracle VM Exporter Appliance virtual machine.

You can download the Oracle VM Exporter Appliance from the following location:

- Oracle Software Delivery Cloud (OSDC) at https://edelivery.oracle.com.

Once there, search for the Oracle VM Exporter Appliance.

You have to agree to the same terms and conditions as the Oracle VM software.

Note

You can only use the Oracle VM Exporter Appliance in a valid Oracle VM environment and with Oracle Cloud Infrastructure parameters listed above.

11.3 Creating the Oracle VM Exporter Appliance Virtual Machine

Once the Oracle VM Exporter Appliance software has been successfully downloaded, you can create the Oracle VM Exporter Appliance virtual machine. The Oracle VM Exporter Appliance virtual machine can be installed as part of an Oracle VM Server Pool or stand-alone as an unassigned virtual machine. However, the Oracle VM Exporter Appliance virtual machine, like any other virtual machine, cannot be run as an unassigned virtual machine.

Follow these steps to create the Oracle VM Exporter Appliance virtual machine:

1. Create the Oracle VM Exporter Appliance virtual machine from the Oracle VM Exporter Appliance OVA.

2. By default, the name of the Oracle VM Exporter Appliance virtual machine is *OVA_name*_**Exporter_Appliance**. Once this virtual machine is created, you should edit it to **Exporter Appliance**.This name is highly recommended, but not strictly required. However, using this name enables the Oracle VM Exporter Appliance wizard to make several user interface steps easier.

   Important

   Using a different name means that other information such as hostname and IP address have to be entered manually when the Oracle VM Exporter Appliance is used.

3. Configure the Oracle VM Exporter Appliance virtual machine's network. For information on creating, editing and removing VNICs on virtual machines, please refer to the *Edit Virtual Machine* and *Create Virtual Machine* sections of the *Oracle VM Manager User's Guide* .

   - The Oracle VM Manager needs to communicate with the Oracle VM Exporter Appliance virtual machine. We recommend tying the virtual machine's Virtual Network Interface Card (VNIC) in slot 0 (eth0 in the virtual machine) to the virtual machine network. The IP address associated with this VNIC is what the user interface for the Oracle VM Exporter Appliance wizard finds.

   - **(Optional)** You can add an additional VNIC to enable the Oracle VM Exporter Appliance virtual machine to access the Network File System (NFS) repository or the repositories where the virtual machines to be exported have their disks.

     Note

     The Oracle VM networks used above must have the "Virtual Machine" role to tie the VNICs to these networks.

   - **(Optional)**You can add an additional VNIC to enable the Oracle VM Exporter Appliance virtual machines to access Oracle Cloud Infrastructure.

4. Start the Oracle VM Exporter Appliance virtual machine.

The virtual machine boots up and uses auto-provisioning to complete configuration.

11.4 Configuring the Oracle VM Exporter Appliance Virtual Machine

After you successfully create Oracle VM Exporter Appliance virtual machine, you can perform configuration tasks to customize the virtual machine. These configuration tasks include setting the system host name, network interfaces, boot protocol, and more. Some parameters are mandatory for the auto-provisioning configuration to complete. If the settings are not configured correctly for the local environment, then Oracle VM Exporter Appliance virtual machine functions incorrectly and provides unexpected results.

The local Oracle VM Exporter Appliance parameters are:

- Local host and local domain name
- Network host name
- Network device
- Activate or deactivate the network interface
- Use DHCP or static configuration of network parameters

The network parameters used by Oracle VM Exporter Appliance can be configured manually (static addressing). You can also use DHCP to supply these parameters.

If you choose static configuration for the network parameters, then you must also configure the following for Oracle VM Exporter Appliance:

- IP address
- Gateway (router) IP address
- DNS server IP address

There are also sharing and security parameters to configure for Oracle VM Exporter Appliance:

- NFS share path
- CA certificate
- Root password

Parameters are sent as VM Messages to the automated virtual machine provisioning process. For more information on sending VM Messages, see the the *Send VM Messages* section of the *Oracle VM Manager User's Guide* .

In the parameter configuration listing, replace the **parameter.n** ending with **parameter.0** to configure the *eth0* interface, and replace the **parameter.n** ending with **parameter.1** to configure the *eth1* interface. You can have more than two interfaces.

The parameters and values used to configure the Oracle VM Exporter Appliance virtual machine are

1. To set the system host name entry:

   ```
   com.oracle.linux.hostname host-name
   ```

   where *host-name* is the host name, such as `example-hostname.domain`.

2. To set the host name entry for the `/etc/hosts` file:

   ```
   com.oracle.linux.network.host.n host-name-info
   ```

where *n* is the interface number and *host-name-info* is the host name, such as `127.0.0.1 example-hostname.domain example-hostname`.

3. To configure the network device:

   ```
   com.oracle.linux.network.device.n network-device
   ```

   where *n* is the interface number and *network-device* is the local network information, such as `eth0`.

4. To activate or deactivate the interface on the system:

   ```
   com.oracle.linux.network.onboot.n yes or no
   ```

   where *n* is the interface number and *yes* or *no* determines if the interface is active or not.

5. To configure the boot protocol:

   ```
   com.oracle.linux.network.bootproto.n dhcp or static
   ```

   where *n* is the interface and *dhcp* or *static* determines if the interface uses DHCP or has a static IP address.

6. To configure the IP address of the interface:

   ```
   com.oracle.linux.network.ipaddr.n n.n.n.n
   ```

   where *n* is the interface and *n.n.n.n* is the IPv4 address in dotted decimal notation.

7. To configure the netmask of the interface:

   ```
   com.oracle.linux.network.netmask.n n.n.n.n
   ```

   where *n* is the interface and *n.n.n.n* is the IPv4 netmask in dotted decimal notation.

8. To configure the IP address of the interface's network gateway (router):

   ```
   com.oracle.linux.network.gateway.n n.n.n.n
   ```

   where *n* is the interface and *n.n.n.n* is the IPv4 address in dotted decimal notation.

9. To configure the IP addresses of the Domain Name System (DNS) servers, separated by commas:

   ```
   com.oracle.linux.network.dns-servers.n n.n.n.n,n.n.n.n,...
   ```

   where *n* is the interface and *n.n.n.n* is the IPv4 address of one or more DNS servers in dotted decimal notation.

10. To configure the NFS share (path) to hold temporary converted disk images:

    ```
    com.oracle.ovm.vmexporter.nfs_share path
    ```

    where *path* is the path to the temporary disk images.

11. To configure Oracle VM Manager's Certificate Authority (CA) certificate:

    ```
    com.oracle.ovm.vmexporter.ca_certificate certificate
    ```

    where *certificate* is the value of the certificate, passed as a string with carriage returns in the certificate replaced with new lines (\n). For more invformation on obtaining and exporting the CA certificate, see Section 2.2.4, “Exporting the CA Certificate” and related sections.

Note

Before setting the Oracle VM CA certificate for the Oracle VM Exporter Appliance, the certificate must be converted from a multi-line string to a single-line string. Use this Linux command to convert carriage-return and any "/" (forward slash) in the certificate to a compatible format:

```
# sed 's-/-//-g;1h;1!H;$!d;x;;s-\n-/\\n-g' ca.crt
```

When selecting the single-line string certificate, include the *"-----BEGIN CERTIFICATE-----"* and *"-----END CERTIFICATE-----"*

12. To configure Oracle VM Manager's root password:

    ```
    com.oracle.linux.network.root-password password
    ```

    where *password* is the root password.

You set these parameters in the Oracle VM Exporter Appliance by sending messages to the virtual machine. If you have named the virtual machine "Exporter Appliance" then you can pass these parameters as shown in this example:

```
sendVmMessage Vm name="Exporter Appliance" key=com.oracle.linux.hostname message=
     example-hostname.domain log=no
sendVmMessage Vm name="Exporter Appliance" key=com.oracle.linux.network.host.0 message=
     127.0.0.1 example-hostname.domain example-hostname log=no
sendVmMessage vm name="Exporter Appliance" key=com.oracle.linux.network.device.0 message=
     eth0 log=no
sendVmMessage vm name="Exporter Appliance" key=com.oracle.linux.network.onboot.0
     message=yes log=no
sendVmMessage vm name="Exporter Appliance" key=com.oracle.linux.network.bootproto.0
     message=dhcp log=no
sendVmMessage Vm name="Exporter Appliance" key=com.oracle.ovm.vmexporter.nfs_share
     message="ca-ovmstor101://export//<user-id>//converteddisks" log=no
sendVmMessage Vm name="Exporter Appliance" key=com.oracle.ovm.vmexporter.ca_certificate
     message="-----BEGIN CERTIFICATE-----/\nMIID+zCCAuOgAwIBAgIUZwL8sCLKIaknyZMNFmPtcc
     Vc86MwDQYJKoZIhvcNAQEL/\nBQAwgaQxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpDYWxpZm9ybmlhMRUw
     EwYDVQQH/\nEwxSZWR3b29kIENpdHkxGzAZBgNVBAoTEk9yYWNsZSBDb3Jwb3JhdGlvbjEaMBgG/\nA1U
     ECxMRT3JhY2xlIFZNIE1hbmFnZXIxMDAuBgNVBAMTJ09WTSBDQSAwMDA0ZmIw/\nMDAwMDEwMDAwMDgwN
     mM5OGJiMWRiMDRkYzAeFw0yMDA0MjMwMDUwMDhaFw0zMDA0/\nMjQwMDUwMDhaMIGkMQswCQYDVQQGEwJ
     VUzETMBEGA1UECBMKQ2FsaWZvcm5pYTEV/\nMBMGA1UEBxMMUmVkd29vZCBDaXR5MRswGQYDVQQKExJPc
     mFjbGUgQ29ycG9yYXRp/\nb24xGjAYBgNVBAsTEU9yYWNsZSBWTSBNYW5hZ2VyMTAwLgVDVQQDEydPVk0
     gQ0Eg/\nMDAwNGZiMDAwMDAxMDAwMDA4MDZjOThiYjFkYjA0ZGMwggEiMA0GCSqGSIb3DQEB/\nAQUAA4
     IBDwAwggEKAoIBAQCSma4RrVJ5//TR7Iz2j0zq3e3cmGL82kFbqhVtG2ufz/\nqbeOkYD2cpObpP0KUAF
     pQ9UHQxVbkWSKKTAhn2UPAPdA6mMjKZ17PMIpBiEmnD/\n0N1zE3a9IBp7wMd3X2zdpBQadVVD3v7P9k+
     rFhprlbqXG4BZxUnc//SYsgBdrDkOj/\nn6RrGJoufpIc5TADOOcpu3HaDaZ4DUj3tASzMesXRPnj45d/
     /F8axJnxxRzC2+L3b/\nJIVPdzyBK5ZSqU2rGPesQixC56yW//iywgj2i1n0+O60Uv2ypBZ4GAjYSGY5A
     I+qE/\ne//DRsrE6FtI+cmoUHckoWRtu0f8QdNBiYzmQzDYqOs+5AgMBAAGjIzAhMA8GA1Ud/\nEwEB//
     wQFMAMBAf8wDgYDVR0PAQH//BAQDAgAFMA0GCSqGSIb3DQEBCwUAA4IBAQBY/\niZosYxc35hIEYbUqxz
     qBes9Bg3fdxZXNIFs1mdxQJUNSd1QSd526lna4kN1hpc70/\nnoAVn8AF3Ct7t8qltc710E8xI9mUO2Us
     ISZtdVvWhUUbeZaLiH0x4SEIZT9BwrxW/\n7ZB//BjCuENCRjTCsZpjj8X9c//nZyB+LqHS7W6VznsDry
     UYhJ6hiHe9//dcnBLlXPH/\nWnqvCiZLHFeletI6c1ahMA6R7+arx9EPp5MjZfXU8slLmzSCrZaYwFd0z
     F//Gy8Xz/\nfEYbjAkO6T3T//OnLBC1Q4fs80jD5n2y13YKpTPSaFgO0An//YgbWecd7JjJE//5BVQ/\n
     g37A8ZcQkMxBZyELRv1C/\n-----END CERTIFICATE-----/\n" log=no
sendVmMessage Vm name="Exporter Appliance" key=com.oracle.linux.root-password
     message=<root-pw> log=no
```

Note

You must also set the date and time correctly. See Section 8.8.2, “Enabling and Disabling Configuration Modules (ovm-chkconfig)” for more information on setting the date and time and related parameters.

Chapter 12 Backing up and Restoring Oracle VM Components

This chapter gives you the basic information required to back up and restore *Oracle VM Manager*, the database repository, and *virtual machines*.

Oracle does not recommend backing up Oracle VM Servers as no critical data is contained on them. Instead of backing up and recovering an Oracle VM Server, in the event of a failure, simply delete it from Oracle VM Manager, reinstall the Oracle VM Server software if required, and rediscover it.

For a more thorough discussion on backing up and restoring the various entities that comprise Oracle VM, see the *Oracle VM 3: Backup and Recovery Best Practices Guide* at:

https://www.oracle.com/technetwork/server-storage/vm/overview/index.html

12.1 Backing up and Restoring Oracle VM Manager

This section contains information about how to manually back up and restore *Oracle VM Manager*, including the database repository.

From a high level, the steps to manually backup Oracle VM Manager are as follows:

1. Make a copy of the Oracle VM Manager configuration file.

   See Section 12.1.1, “Backing up the Oracle VM Manager Configuration File” for more information on backing up the Oracle VM Manager configuration file.

2. Back up the Oracle VM Manager database.

   Note that the Oracle VM Manager database is backed up automatically every 24 hours using the MySQL Enterprise Backup utility, but it is also possible to perform a manual backup.

   See Section 12.1.2, “Backing up the MySQL Database Repository” for more information on Oracle VM Manager MySQL database backup and restore facilities.

12.1.1 Backing up the Oracle VM Manager Configuration File

To back up Oracle VM Manager, you should make a copy of the Oracle VM Manager configuration file located at:

`/u01/app/oracle/ovm-manager-3/.config`

This configuration file contains database connection information, ports, and the Oracle VM Manager UUID. The following is an example of the configuration file structure:

```
DBTYPE=MySQL
DBHOST=localhost
SID=ovs
LSNR=46500
OVSSCHEMA=ovs
APEX=None
WLSADMIN=weblogic
OVSADMIN=admin
COREPORT=54321
UUID=uuid
BUILDID=x.x.x.xxx
```

The following table describes the properties and values in the configuration file:

Name	Description
DBTYPE	Database type. This is a legacy configuration property. The value is always MySQL.
DBHOST	Hostname of database server. This is a legacy configuration property. The value is always localhost.
SID	Oracle System ID (SID). The default value is ovs.
LSNR	Database listener port number. The default value is 49500.
OVSSCHEMA	Oracle VM Manager database name. The default value is ovs.
APEX	This is a legacy configuration property. The default value is None.
WLSADMIN	Oracle WebLogic Server administrator username. The default value is weblogic.
OVSADMIN	Oracle VM Manager administrator username. The default value is admin.
COREPORT	Oracle VM Manager core port number. The default value is 54321.
UUID	Oracle VM Manager universally unique identifier (UUID).
BUILDID	Oracle VM Manager version and build number.

12.1.2 Backing up the MySQL Database Repository

This section describes the *Oracle VM Manager* MySQL backup facility.

As of Oracle VM Manager Release 3.2.1, backups of the local MySQL database are performed automatically. Backups are stored within /u01/app/oracle/mysql/dbbackup by default, and are rotated regularly so that only the most recent backups are stored at any point in time. Backups make use of the MySQL Commercial Backup utility. See https://www.mysql.com/products/enterprise/backup.html for more information on the MySQL Enterprise Backup utility.

The MySQL Enterprise Backup package is installed as a dependency during the installation of Oracle VM Manager. On Oracle Linux systems this is handled by installing mysql-commercial-backup-*version_number*.x86_64.rpm.

On x86 systems, backup configuration options are defined in /etc/sysconfig/ovmm on the Oracle VM Manager host.

To configure the default path used to store MySQL database backup files, locate the following line:

```
DBBACKUP=/u01/app/oracle/mysql/dbbackup
```

Note

This path can be changed to an alternate location if you need to cater to disk space requirements.

The default path for the mysqlbackup binary is specified in the following line:

```
DBBACKUP_CMD=/opt/mysql/meb-x.x/bin/mysqlbackup
```

Warning

This path is made explicit for the purposes of handling future updates to the MySQL Enterprise Backup package. It should not be changed.

Configuration options such as how frequently the automated database backup facility should run and how many backups should be kept through rotations, are stored within the database itself. These parameters can be set using either the Oracle VM Manager Web Interface or the Oracle VM Manager Command Line Interface. For more information on how to set these parameters, please refer to *Prefenences* section in the Oracle VM Manager Online Help and the *setDbBackupConfig* command in the *Oracle VM Manager Command Line Interface User's Guide* .

12.1.2.1 Contents of the Backup Directory

The MySQL database backup directory has the following naming convention: `AutoFullBackup-`*`MMDDYYYY_hhmmss`*.

The backup directory contains the following:

- `AutoBackup.log`, which contains information about the events that took place during the backup process.
- A copy of the MySQL configuration file.
- `datadir` directory that contains the binary log for the database.
- `meta` directory that contains files specific to the MySQL Enterprise Backup process.
- `MBI` image file for the database that is backed up

12.1.2.2 Configuring Backup Interval and Rotation

You can set the frequency of MySQL database backups as well as the number of database backups that Oracle VM Manager retains. See the *Preferences* section in the Oracle VM Manager Online Help for more information.

12.1.2.3 Backing Up the MySQL Database Manually

It is possible to manually initiate a backup. This is usually done when performing an upgrade of Oracle VM Manager. While it is possible to invoke the `mysqlbackup` utility directly, it is recommended that you use the backup script at `/u01/app/oracle/ovm-manager-3/ovm_tools/bin/BackupDatabase`.

The following is an example of this script:

```
# /u01/app/oracle/ovm-manager-3/ovm_tools/bin/BackupDatabase -w
Enter your OVM Manager username: admin
Enter your OVM Manager password:

INFO:  Backup job starting with destination:
        /u01/app/oracle/mysql/dbbackup/ManualBackup-time_stamp

        Job Id   = 'Start Backup to: ManualBackup(ID)
        Uri: https://localhost:7002/ovm/core/wsapi/rest/Job/ID'
        Job Name = 'Start Backup to: ManualBackup'

INFO:  Backup job finished
```

Note

It is important that MySQL root user password is the same on both the Oracle VM Manager and MySQL database. Otherwise, database backups fail.

By default, the backup script stores the backup as a manual backup, to avoid the rotation that takes place for automatic backups.

The preceding example uses the `-w` command-line switch to force the backup script to wait until the backup job is complete. This option is useful if you need to capture potential error messages, but it also causes the script to wait until the job either completes or exits due to an error. If you do not use the `-w` command-line switch, you should check the status of the backup job within the Oracle VM Manager Web Interface or Oracle VM Manager Command Line Interface to determine whether or not the job completes successfully. You can get a full list of supported options for this command with the `-h` command-line switch.

> **Note**
>
> The backup script assumes that you are using a CA-signed SSL certificate within a production environment. Using a self-signed certificate is not recommended and may result in an error when you run the script. It is possible to override SSL verification by using the `--insecure` command-line parameter, however this may compromise the security of the operation and is not recommended. A better approach to resolving any SSL verification error, is to install an SSL certificate signed by a recognized CA, as described in Section 2.2.6, “Changing the Default SSL Certificate”.

For more information on using MySQL Enterprise Backup, see https://dev.mysql.com/doc/mysql-enterprise-backup/en.

12.1.3 Oracle VM Manager Backup and Restore Troubleshooting

If you are experiencing problems with Oracle VM Manager, do the following:

1. Perform a database restore. See Section 12.1.4, “Restoring Oracle VM Manager” for more information.

   Provided that you have a good backup in your archive, you should be able to revert back to it. To check for a recent database backup, on the Oracle VM Manager host, run the following command:

   ```
   ls -ltr /u01/app/oracle/mysql/dbbackup/
   ```

   If a backup exists, you can proceed with the steps documented in Section 12.1.4, “Restoring Oracle VM Manager”.

   > **Note**
   >
   > You can also refer to *Doc ID 2405023.1* in the Oracle Support Knowledge Base for information about restoring the Oracle VM Manager database.

   There is a check completed before each backup is run to determine if the database can be backed up. If the database consistency check fails, automatic backups are no longer generated. For more information, see *Doc ID 2060953.1* in the Oracle Support Knowledge Base.

2. If you do not have a good backup of the database or if you have other database corruption issues, rebuild the database. See Section 12.1.5, “Restoring Oracle VM Manager If You Have No Database Backup”

   Also, see *Doc ID 2038168.1* in the Oracle Support Knowledge Base for additional information about regenerating the Oracle VM Manager database.

12.1.4 Restoring Oracle VM Manager

To restore Oracle VM Manager, and the Oracle VM Manager database schema from a backup, do the following:

1. First, if you need to reinstall or upgrade Oracle VM Manager, use the Oracle VM Manager installation media to perform an install or upgrade of the software on your server. See the Installation and Upgrade Guide.

   You should perform the install using the `runInstaller.sh --uuid` *`uuid`* command and provide the UUID from the previous manager installation you created a backup from. The UUID can be found in the Oracle VM Manager configuration file.

   > **Note**
   >
   > The Oracle VM Manager UUID is also persisted in the `/etc/sysconfig/ovmm` file on Oracle Linux, and in the `/etc/opt/ovmm` file on . If the system disk of the server on which you are installing or restoring Oracle VM Manager was not wiped entirely, the existing UUID is still present and will be detected when running the installer.
   >
   > - The `--uuid` option on Oracle Linux overrides this existing UUID. Oracle Solaris users must use the shortened form of this option: `-u`.
   >
   > - If no UUID is present in `/etc/sysconfig/ovmm`, the `--uuid` option adds the UUID to the file on Oracle Linux. On Oracle Solaris, the `-u` option adds the UUID to `/etc/opt/ovmm` if the UUID is not present in this file.

   An example install command syntax for Oracle Linux is as shown in this example:

   ```
   # ./runInstaller.sh --uuid 0004FB000000100002CB7F2DFFA8D8
   ```

   When the Oracle VM Manager installer prompts for installation information, reuse the same usernames for the database schema, Oracle WebLogic Server and Oracle VM Manager administration user, as set out in the backup of the Oracle VM Manager configuration file.

   If possible, you should reuse the same passwords that existed for Oracle VM Manager prior to reinstallation, to avoid problems restarting the Oracle VM Manager service after Oracle VM Manager has been restored from backup. If you intend to change these passwords, do so after you have completed the restore operation.

   Should you use a new password during reinstallation, you are unable to start the Oracle VM Manager service after the database has been restored. To rectify this situation, you must manually reset the passwords for the **ovs** and **appfw** users in the MySQL database. This can be achieved using the `mysqladmin` tool.

2. After installation, reinstallation or upgrade, stop the Oracle VM Manager Command Line Interface, Oracle VM Manager, and the database before you restore the backup. On Linux:

   ```
   # /sbin/service ovmcli stop
   # /sbin/service ovmm stop
   # /sbin/service ovmm_mysql stop
   ```

3. To initiate the database restore, as the **oracle** user, use the `RestoreDatabase` command located in `/u01/app/oracle/ovm-manager-3/ovm_tools/bin`, for example:

   ```
   # su - oracle
   $ bash /u01/app/oracle/ovm-manager-3/ovm_tools/bin/RestoreDatabase.sh \
        ManualBackup-time_stamp_ID
   ```

   The `RestoreDatabase` script expects the name of the directory for a particular backup directory as described in the Installation and Upgrade Guide. You do not need to specify the full path to the backup directory as this is already specified in the **DBBACKUP** variable.

The RestoreDatabase script prompts you to remove existing database directories and their contents so that the database restore operation can complete successfully. You must confirm that it is safe to delete this data before the script can proceed. If you opt not to delete this data, the script cannot continue until the data has been removed. It is recommended that you allow the script to perform this action rather than attempting to do this manually:

```
Before the database can be restored,
  the following database directories/files must be deleted:
appfw ibdata1 ib_logfile0 ib_logfile1 mysql ovs performance_schema

Are you sure it is safe to delete these directories/files now? [y,n] y
Deleting directory /u01/app/oracle/mysql/data/appfw
Deleting directory /u01/app/oracle/mysql/data/ibdata1
Deleting directory /u01/app/oracle/mysql/data/ib_logfile0
Deleting directory /u01/app/oracle/mysql/data/ib_logfile1
Deleting directory /u01/app/oracle/mysql/data/mysql
Deleting directory /u01/app/oracle/mysql/data/ovs
Deleting directory /u01/app/oracle/mysql/data/performance_schema
INFO: Expanding the backup image...
INFO: Applying logs to the backup snapshot...
INFO: Restoring the backup...
INFO: Restoring OVM keystores and certificates
INFO: Success - Done!
INFO: Log of operations performed is available at:
 /u01/app/oracle/mysql/dbbackup/ManualBackup-time_stamp_ID/Restore.log


IMPORTANT:

      As 'root', please start the OVM Manager database and application using:
            service ovmm_mysql start; service ovmm start; service ovmcli start
```

> **Important**
>
> The RestoreDatabase script performs a version check to ensure that the database version matches the version of the database from which the backup was created. If there is a version mismatch, the script exits with a warning, as this action may render Oracle VM Manager unusable.
>
> It is possible to override this version check by using the --skipversionchecks option when invoking the script. This option should be used with care as version mismatches may have undesirable consequences for Oracle VM Manager.
>
> For example, database backups from an earlier 3.4.x release cannot be used in an Oracle VM Manager deployment at release 3.4.5 or later, due to database schema changes.

4. Restart the database and Oracle VM Manager, and the Oracle VM Manager Command Line Interface. On Oracle Linux:

```
# /sbin/service ovmm_mysql start
# /sbin/service ovmm start
# /sbin/service ovmcli start
```

5. Because the certificates required to authenticate various components, such as the Oracle VM Manager Web Interface and Oracle VM Manager Command Line Interface, are regenerated during the new

installation and the mappings for these are overwritten by the database restore, it is necessary to reconfigure the certificates used to authenticate these components.

Run the following script to reconfigure the Oracle WebLogic Server:

```
# export MW_HOME=/u01/app/oracle/Middleware
# /u01/app/oracle/ovm-manager-3/ovm_upgrade/bin/ovmkeytool.sh setupWebLogic
```

For more information on the `ovmkeytool.sh` script, see Section 2.2.1, “Oracle VM Key Tool”.

6. If you moved Oracle VM Manager to a new host, you must generate a new SSL key as follows:

```
# /u01/app/oracle/ovm-manager-3/ovm_upgrade/bin/ovmkeytool.sh gensslkey
```

For more information on generating a new SSL key, see Section 2.2.5, “Generating a New SSL Key”.

7. Restart Oracle VM Manager and then run the client certificate configuration script, as follows:

```
# /sbin/service ovmm restart
# /u01/app/oracle/ovm-manager-3/bin/configure_client_cert_login.sh /path/to/cacert
```

Where */path/to/cacert* is the absolute path to the CA certificate. You must provide the path to the CA certificate if you used a CA other than the default Oracle VM Manager CA to sign the SSL certificate.

The script requires that Oracle VM Manager is running, and prompts you for the administrator username and password that should be used to access Oracle VM Manager. The script makes changes that may require Oracle VM Manager to be restarted:

```
# /sbin/service ovmm restart
```

8. Within Oracle VM Manager go to the **Servers and VMs** tab and perform a **Refresh All** on your existing *server pools*. Refer to the Oracle VM Manager Online Help for more information on these options.

12.1.5 Restoring Oracle VM Manager If You Have No Database Backup

⚠ **Important**

The instructions provided here should be used as a last resort. You really must ensure that your backup strategy is adequate and that the database backups are available on storage that is suitable for this purpose. Typically, these backups should be stored on some form of network attached storage, preferably with a RAID that provides some form of mirroring. To change the backup path to a suitable location, see Section 12.1.2, “Backing up the MySQL Database Repository”.

If you have reinstalled Oracle VM Manager from scratch, using the `runInstaller.sh --uuid` *`uuid`* command and have provided the UUID from the previous manager installation, but you do not have a database backup, a certain level of recovery is possible based on the information stored on the Oracle VM Servers and in your attached storage. It is important that you follow a set order of actions to ensure that the server pools that your Oracle VM Servers are members of are able to be properly recovered. These steps are outlined as follows:

1. Discover one Oracle VM Server from each server pool.

2. Discover the storage that contains the server pool file system. Present it to the newly discovered Oracle VM Server. Refresh the storage.

3. Refresh the file system or physical disks that contain the server pool file system.

4. Refresh the file systems or physical disks that contain the repositories used by server pool. If you get an error, when refreshing a physical disk, similar to the following:

   ```
   OVMAPI_7281E Cannot perform operation on file system...
   ```

   then take ownership of the repositories and try to perform the physical disk refresh again.

5. Present the repositories to the Oracle VM Server.

6. Refresh the repositories.

7. Discover the remaining Oracle VM Servers in the server pool.

8. Refresh all Oracle VM Servers in the server pool to discover the virtual machines.

12.1.6 Oracle WebLogic Server Backup and Restore

In general, it is not necessary to perform a separate backup of the Oracle WebLogic Server component used by Oracle VM Manager, however in the instance that you have created separate Oracle WebLogic Server users to facilitate a number of different login credentials that can be used to access Oracle VM Manager you may need to perform your own backup of the Oracle WebLogic Server LDAP directory used for authentication. This is particularly important if you intend to upgrade Oracle VM Manager, as there is a possibility that any user credentials that have been manually configured within Oracle WebLogic Server may be lost during an update process.

Full documentation describing the Oracle WebLogic Server LDAP backup process and how to configure Oracle WebLogic Server LDAP backups can be found at:

https://docs.oracle.com/middleware/1221/wls/START/failures.htm#START162

In the Oracle VM context, LDAP data for Oracle WebLogic Server is stored in:

`/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/data/ldap`

Based on the information provided in the Oracle WebLogic Server documentation, you can perform a full backup of this directory on your own schedule, or you can rely on Oracle WebLogic Server's automated backups located in:

`/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/data/ldap/backup`

If you choose to use the Oracle WebLogic Server backup service, you can change default backup parameters. See the *Configure backups for embedded LDAP servers* topic in the Oracle WebLogic Server online help.

12.2 Backing up Virtual Machines

There are a number of options you can use to take a back up of a *virtual machine*. This section discusses some of them and the pros and cons for each.

One of the main points that should be considered when backing up a virtual machine, is whether you can shut down the virtual machine during the back up. Backing up a running virtual machine allows the machine to be available for use, but does not allow for a back up that is consistent or easy to restore. Creating a back up of a running virtual machine is similar to taking a back up of a running database without putting the tablespaces in back up or read-only mode. The first couple of blocks you back up from a the virtual machine are likely to be out of sync with the last blocks of your back up. If and when you try to restore a back up taken from a running virtual machine, you may not be able to rebuild the machine due to disk errors.

You can install back up software in the virtual machine, for example Oracle Secure Backup. This allows you to safely back up a running virtual machine. The ease by which you can restore the virtual machine from the back up depends on the software used.

The following table discusses some virtual machine back up options, with some high level benefits and disadvantages of each method. This is not an exhaustive list of all the options that may be available.

Table 12.1 Virtual Machine Backup Options

Backup Option	**Benefits**	**Disadvantages**
Install back up software in the virtual machine and back up to an external source.	Virtual machine can be running. Fine grained control of files backed up.	
Create a back up of the virtual machine from the storage repository (see *Clone a Virtual Machine or Template* section in the User Guide).	Consistent virtual disk status.	Virtual machine must be stopped.
Create a cold *clone* of the (stopped) virtual machine (see the *Clone a Virtual Machine or Template* section in the User Guide, then back up the clone from the storage repository (see the *Repositories Perspective* section in the User Guide).	Consistent virtual disk status.	Virtual machine must be stopped.
Create a hot clone of the (running) virtual machine (see the *Clone a Virtual Machine or Template* section in the User Guide, then back up the clone from the storage repository (see the *Repositories Perspective* section in the User Guide).	Virtual machine can be running.	Inconsistent virtual disk status. Virtual disks may need to be recovered using a disk repair utility. May cause data loss or corruption. Only available on *OCFS2-based* file systems (iSCSI or fibre channel-based storage). Should not be used for virtual machines running an Oracle Database (instead use the rman utility or similar).
Create a back up of the entire storage repository (see the *Repositories Perspective* section in the User Guide).	Back up all virtual machines at once. Consistent virtual disk status.	Virtual machines must be stopped.

The two recommended strategies for backing up a virtual machine are to:

- Shut down the virtual machine and create a cold clone, then back up the clone files from the storage repository.

- Shut down the virtual machine and back up the virtual machine files from the storage repository.

These two options create a safe back up, with the *virtual disks* in a stable and consistent state. To restore the virtual machine, import the virtual machine into the storage repository (see the *Import Virtual Machine* section of the *Server Pools Folder* in the User Guide).

Chapter 13 Troubleshooting Oracle VM

This chapter describes how to troubleshoot common issues with Oracle VM.

For more information and support, see the following Oracle sites:

- My Oracle Support, https://support.oracle.com
- Oracle Virtualization Community, https://community.oracle.com/community/groundbreakers/server_%26_storage_systems/virtualization
- For additional information on best practices regarding Oracle VM deployments, contact Oracle Support and refer to Document ID 1940756.1.

13.1 Capturing Diagnostic Information for Oracle VM

Oracle Support Services provide a script, `vmpinfo3.sh`, to collect diagnostic information from your Oracle VM environment for troubleshooting purposes. This script is installed with Oracle VM Manager in the following location:

```
/u01/app/oracle/ovm-manager-3/ovm_tools/support/vmpinfo3.sh
```

Syntax

```
./vmpinfo3.sh { --username=admin } [ listservers | servers=server1,server2,server3 ]
```

Options

Option	Description
{ `--username=admin` }	Specifies the user that runs the script. You should not specify a user other than the default Oracle VM Manager user, `admin`.
[`listservers`]	Lists all servers for which the script can collect diagnostic information. If you specify this option, the script displays the list of servers that the Oracle VM Manager owns and then exits. You must then run the script again to collect diagnostic information.
[`servers=server_list`]	Specifies at least one instance of Oracle VM Server for which you want to collect diagnostic information. Use this option to exclude servers in your environment or limit the diagnostic collection to certain servers only. Separate multiple server names with a comma. **Tip** It can take the script several minutes to collect diagnostic information across your Oracle VM environment. To save time, specify this option to include only the instances of Oracle VM Server for which issues might exist, especially if you have a large number of Oracle VM Server deployments.

Examples

To collect diagnostic information for your Oracle VM environment, including Oracle VM Manager and all of the owned Oracle VM Servers in the model, run:

```
# ./vmpinfo3.sh --username=admin
```

To list the servers for which you can collect diagnostic information, run:

```
# ./vmpinfo3.sh --username=admin listservers
```

To collect diagnostic information for specific servers only, run:

```
# ./vmpinfo3.sh --username=admin servers=myserver1.example.com,myserver2.example.com
```

Output

The script creates a tarball in the `/tmp` directory that contains log files, sosreports, and other diagnostic information such as details about the Oracle VM model.

When the script completes successfully, it displays the filename of the tarball, as in the following example:

```
=======================================================================================
 Please send /tmp/vmpinfo3-3.x.y.z-time_stamp.tar.gz to Oracle OVM support
=======================================================================================
```

13.2 Troubleshooting Oracle VM Server

This section describes some problems you may encounter when using *Oracle VM Server*, and explains how to resolve them.

If you need to contact Oracle Support Services, you will be asked to supply the log files mentioned in this section. You may also be required to provide the exact version of each Oracle VM component. To find the version of *Oracle VM Manager*, click the **Help** menu, then **About**. To find the version of Oracle VM Server and *Oracle VM Agent*, see the *Control Domain Perspective* section of the *Oracle VM Manager User's Guide* .

13.2.1 Oracle VM Server Debugging Tools

If *virtual machine* creation fails, check the Oracle VM Server log files and use the command-line tools to help you find the cause of a problem. There are a number of useful command-line tools, important directories, and log files that you should check when troubleshooting problems with Oracle VM Server. This section discusses these tools and log files.

13.2.1.1 Oracle VM Server Directories

The important Oracle VM Server directories you should check when troubleshooting problems with Oracle VM Server are listed in Table 13.1, “Oracle VM Server directories”.

Table 13.1 Oracle VM Server directories

Directory	Purpose
`/etc/xen`	Contains Oracle VM Server configuration files for the Oracle VM Server daemon and virtualized *guests*.
`/etc/xen/scripts`	Contains networking related scripts.
`/var/log`	Contains the following files: . • `ovs-agent.log`, log file for the Oracle VM Agent. • `ovmwatch.log`, logs virtual machine life cycle *events*.

Directory	Purpose
	• ovm-consoled.log, logs remote VNC console access, and all communication with Oracle VM Manager. • messages*, logs all Oracle VM Server messages.
/var/log/xen	Contains Oracle VM Server log files.

13.2.1.2 Oracle VM Server Log Files

The Oracle VM Serverlog files you should check when troubleshooting problems with Oracle VM Server are listed in the following table:

Table 13.2 Oracle VM Server log files

Log File	Directory	Description
xend.log	/var/log/xen/	Contains a log of all the actions of the Oracle VM Server daemon. Actions are normal or error conditions. This log contains the same information as output using the xm log command.
xend-debug.log	/var/log/xen/	Contains more detailed logs of the actions of the Oracle VM Server daemon.
xen-hotplug.log	/var/log/xen/	Contains a log of hotplug events. Hotplug events are logged if a device or network script does not start up or become available.
qemu-dm.*pid*.log	/var/log/xen/	Contains a log for each *hardware virtualized* guest. This log is created by the quemu-dm process. Use the ps command to find the *pid* (process identifier) and replace this in the file name.
ovs-agent.log	/var/log/	Contains a log for Oracle VM Agent.
osc.log	/var/log/	Contains a log for Oracle VM Storage Connect plug-ins.
ovm-consoled.log	/var/log/	Contains a log for the Oracle VM virtual machine console.
ovmwatch.log	/var/log/	Contains a log for the Oracle VM watch daemon.

13.2.1.3 Oracle VM Server Command Line Tools

The following table lists command line tools you can use when troubleshooting problems with Oracle VM Server:

Note

These command line tools are included as part of the Xen environment. You should refer to the appropriate Xen documentation for more information on using them.

Table 13.3 Oracle VM Server command line tools

Command Line Tool	Description
xentop	Displays real-time information about Oracle VM Server and domains.
xm dmesg	Displays log information on the *hypervisor*.
xm log	Displays log information of the Oracle VM Server daemon.

13.2.2 Using DHCP on Oracle VM Servers

It is recommended that you install Oracle VM Server on a computer with a static IP address. If you use DHCP to manage the IP address space in your environment, the DHCP server should be configured to map the server interface MAC addresses to specific IP assignments. This makes sure your host always receives the same IP address. The behavior of the Oracle VM Server host is undefined if used in an environment where your IP address may change due to DHCP lease expiry.

13.2.3 Cannot Use Certain Key Combinations When Connecting to Dom0 Console

Some server models and some client terminals are not ideally compatible with regard to special key combinations. For instance, on some HP servers, such as the HP DL380G4 (BIOS P51) server, the Alt-F2 key combination required to toggle to the login screen does not work for all terminal clients. Some terminal clients provide alternate key mappings, so it is worth checking the documentation of your selected terminal client to determine whether an alternative mapping is available.

If you are using the Windows PuTTY SSH client, you can press **Alt** + the right arrow key and **Alt** + the left arrow key to toggle the login screen, instead of the printed Alt-F2.

13.2.4 Storage Array LUN Remapping on Oracle VM Servers

Storage array LUN remapping is not supported by Oracle VM Servers. An Oracle VM Server must maintain the connections to a storage array's logical drive using the same LUN IDs. If a LUN is remapped, the following error may be printed in the Oracle VM Server's messages log:

```
Warning! Received an indication that the LUN assignments on this target have changed.
The Linux SCSI layer does not automatically remap LUN assignments.
```

To work around this issue:

- For Fibre Channel storage, reboot the Oracle VM Server. The new storage array LUN IDs are used.
- For iSCSI storage, restart the iscsi daemon on the Oracle VM Server to delete and restore all iSCSI target connections, for example:

  ```
  # service iscsi restart
  ```

  Alternatively, on the Oracle VM Server, log out and log in again to the target for which the LUN IDs have changed, for example:

  ```
  # iscsiadm --mode node --logout ip_address iqn.xyz:1535.uuid
  # iscsiadm --mode node --login ip_address iqn.xyz:1535.uuid
  ```

13.2.5 Tuning ISCSI Settings on Oracle VM Servers

In some cases, it is possible to run into limitations or bugs within a particular ISCSI implementation that may require you to tune your ISCSI settings for storage initiators on each of your Oracle VM Servers.

Typically this is required when you experience an IO lock on a LUN and an unexpected change in the kernel workload on the Oracle VM Server. You may also notice a dramatic increase in network traffic between the Oracle VM Server and the storage array. This particular case has been noted to occur on some ZFS appliances running Oracle Solaris 11 and is related to the Sun iSCSI COMSTAR port provider. The problem can usually be resolved by updating package versions, however if this is not an option you may tune your ISCSI settings on each Oracle VM Server that communicates with a SUN.COMSTAR target to enable flow control.

To tune ISCSI, on each Oracle VM Server, perform the following steps:

- Open `/etc/iscsi/iscsid.conf` in a text editor.
- Locate the section titled `iSCSI settings`.
- Change the value of the entry `node.session.iscsi.InitialR2T` to `yes`, and the value of the entry `node.session.iscsi.ImmediateData` to `no`.
- Save the file.
- Restart the ISCSI daemon by issuing the following command:

```
# service iscsid restart
```

Note

The preferred resolution to this issue is to update your software. Manual configuration of Oracle VM Server settings is not generally advisable, as changes may be lost during future updates.

13.2.6 Troubleshooting Clustered Server Pools Oracle VM Server for x86

There are some situations where removing an Oracle VM Server from a server pool may generate an error. Typical examples include the situation where an OCFS2-based repository is still presented to the Oracle VM Server at the time that you attempt to remove it from the server pool, or if the Oracle VM Server has lost access to the server pool file system or the heartbeat function is failing for that Oracle VM Server. The following list describes steps that can be taken to handle these situations.

- Make sure that there are no repositories presented to the server when you attempt to remove it from the server pool. If this is the cause of the problem, the error that is displayed usually indicates that there are still OCFS2 file systems present. See the *Repositories Perspective* section in the Oracle VM Manager Online Help for more information.
- If a pool file system is causing the remove operation to fail, other processes might be working on the pool file system during the unmount. Try removing the Oracle VM Server at a later time.
- In a case where you try to remove a server from a clustered server pool on a newly installed instance of Oracle VM Manager, it is possible that the file server has not been refreshed since the server pool was discovered in your environment. Try refreshing all storage and all file systems on your storage before attempting to remove the Oracle VM Server.
- In the situation where the Oracle VM Server cannot be removed from the server pool because the server has lost network connectivity with the rest of the server pool, or the storage where the server pool file system is located, a critical event is usually generated for the server in question. Try acknowledging any critical events that have been generated for the Oracle VM Server in question. See the *Events Perspective* section in the Oracle VM Manager Online Help for more information. Once these events have been acknowledged you can try to remove the server from the server pool again. In most cases, the removal of the server from the server pool succeeds after critical events have been acknowledged, although some warnings may be generated during the removal process. Once the server has been removed from the server pool, you should resolve any networking or storage access issues that the server may be experiencing.
- If the server is still experiencing trouble accessing storage and all critical events have been acknowledged and you are still unable to remove it from the server pool, try to reboot the server to allow it to rejoin the cluster properly before attempting to remove it again.

- If the server pool file system has become corrupt for some reason, or a server still contains remnants of an old stale cluster, it may be necessary to completely erase the server pool and reconstruct it from scratch. This usually involves performing a series of manual steps on each Oracle VM Server in the cluster and should be attempted with the assistance of Oracle Support.

13.2.7 Allocating Memory for Multiple Infiniband HCAs

Out of memory errors can occur when using multiple Infiniband host channel adapters (HCA) with the PCIe Scalable Interface (psif) driver for paravirtualized environments on Oracle VM Server. These out of memory errors occur because the psif driver requires a minimum 30 MB of memory for the driver itself in addition to 50 MB of memory for each interface instance. The default memory allocation for dom0 does not provide enough memory to support multiple interfaces.

To resolve the out of memory errors, you should increase the dom0 memory allocation for Oracle VM Server. See Section 1.6, “Changing the Memory Size of the Management Domain”.

The following is an example of an out of memory error that is written to `/var/log/messages`:

```
time_stamp hostname kernel: [ 1465.466059] Out of memory: Kill process
8106 (python) score 0 or sacrifice child
Connection to hostname closed.l: [ 1465.466089] Killed process 8467
     (ovs-agent) total-vm:108488kB, anon-rss:12kB, file-rss:2064kB
```

13.2.8 Resolving Issue Where NIC Fails to Get IP Address if Configured for DHCP

In some cases, when you configure a network interface for Oracle VM Server to retrieve IP addresses via DHCP, and then bring up that interface, it cannot retrieve an IP address and the following error occurs:

```
Determining IP information for interface_name...
failed; no link present. Check cable?
```

This issue can be caused when the time it takes the network interface to come up is greater than the time set for the `LINKDELAY` environment variable.

To resolve this issue, set the value of the `LINKDELAY` to a value of 10 or higher in the `/etc/sysconfig/network-scripts/ifcfg-eth*` file, as in the following example:

```
DEVICE=eth3
BOOTPROTO=dhcp
HWADDR=00:00:00:00:00:00
ONBOOT=yes
LINKDELAY=10
```

13.3 Troubleshooting Oracle VM Manager

This section describes some problems you may encounter when using *Oracle VM Manager*, and explains how to resolve them.

13.3.1 Oracle VM Manager Log Files

Oracle VM Manager error messages are displayed in the user interface, in the Jobs tab, in the object's *Events* list and are also available in log files. Log files are stored in the following directory on the Oracle VM Manager host computer:

`/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs`

Oracle VM Manager that you can use for troubleshooting are as follows:

Log file	Description
access.log	Tracks HTTP access to the Web interface of the Oracle VM Manager and to the underlying *Oracle WebLogic Server* HTTP interface. This log can be used to track access and HTTP operations within Oracle VM Manager to help debug access issues and to audit access to the Oracle VM Manager.
AdminServer.log	Tracks events within the underlying Oracle WebLogic Server framework, including events triggered by Oracle VM Manager. This log can be used to track a variety of issues within Oracle VM Manager including TLS/SSL certificate issues, server availability issues, and any actions performed within Oracle VM Manager which are usually identifiable by searching for items containing the string com.oracle.ovm.mgr. Log in failures resulting from locked accounts (as opposed to incorrect credentials) are also in this file.
AdminServer-diagnostic.log	Tracks exceptions within the underlying Oracle WebLogic Server framework, including particular events triggered by Oracle VM Manager such as log in failures due to incorrect credentials. This log can be used to track Oracle VM Manager behavior that results in an exception or for log in failure, which can be tracked by searching for the string An incorrect username or password was specified.

Because log file format is determined by Oracle WebLogic Server, many of these files may be difficult to read. A log parsing tool is included with Oracle VM Manager to help extract useful information from the actual log files. The log parsing tool is named OvmLogTool.py and is located at:

/u01/app/oracle/ovm-manager-3/ovm_tools/bin

OvmLogTool.py can do the following things:

- Convert and combine all the AdminServer log files into one easier-to-read file.
- Create a filtered summary log file that only lists errors.
- Tail the AdminServer log, applying the filtering on the fly.

Usually analysis of the logs starts by generating an error summary log. The summary file can act as an index into the filtered file to investigate and analyze errors, providing you with time stamps an a shortened summary of each error that may need further investigation. To generate a summary log file, do the following:

```
# python OvmLogTool.py -s -o summary
processing input file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/
AdminServer/logs/AdminServer.log00001
processing input file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/
AdminServer/logs/AdminServer.log
```

This generates a file named summary in the local directory. You can use this to look for *errors* that occurred within Oracle VM Manager.

To get a full log of all events and errors within Oracle VM Manager you can do the following:

```
# python OvmLogTool.py -o filteredlog
processing input file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/
AdminServer/logs/AdminServer.log00001
processing input file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/
AdminServer/logs/AdminServer.log
```

This generates a file named filteredlog in the local directory. You can use this to look for all events that occurred within Oracle VM Manager.

Finally, you can use `OvmLogTool.py` to filter results on the fly while tailing the log:

```
# python OvmLogTool.py -t
tailing log file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/
AdminServer/logs/AdminServer.log
```

Use **Ctrl**+**C** to exit the program when you have finished tailing the log file.

13.3.2 Oracle VM Manager Web Interface Database Synchronization

The Oracle VM Manager Web Interface and the Oracle VM Manager Command Line Interface both use a representation of the Oracle VM data model to to more quickly retrieve data objects, which optimizes the performance of Oracle VM Manager. The representation of the data model is saved to a separate database to the primary Oracle VM Manager database. This separate database relies on events from Oracle VM Manager to stay synchronized with the actual data model.

It is possible, in specific cases, for the representation of the data model to become out of sync with the actual data model. As a result, some objects in the Oracle VM Manager Web Interface do not reflect the actual environment. A typical scenario where this may happen is during a virtual machine clone operation that fails. During this process, the virtual machine is actually created within the data model and the database used by the user interface layer. If a part of the whole operation fails, Oracle VM Manager attempts to clean up and roll back, resulting in the virtual machine being removed from the data model. However, in this case, an event is not generated to notify that the clean up has succeeded, and the virtual machine information remains in the user interface database. The result is that the cloned virtual machine is still shown in the Oracle VM Manager Web Interface and the Oracle VM Manager Command Line Interface even though it does not actually exist in the environment.

The user interface database is not resynchronized automatically when the service is restarted, as this can take a long time for large environments. To force database resynchronization, you must create a file on the Oracle VM Manager host before restarting the service. The following instructions explain how to force resynchronization.

Steps to resynchronize the Oracle VM Manager Web Interface database

- You must access the shell of the Oracle VM Manager host, either directly or over SSH.
- Change user to the 'oracle' user, using `su`.

  ```
  # su - oracle
  ```

- Touch a file called `/tmp/.resyncUI` as the oracle user.

  ```
  $ touch /tmp/.resyncUI
  ```

  If you are unable to do this as the oracle user, then touch the file as any other user but ensure that its permissions are such that any other user can delete the file:

  ```
  # touch /tmp/.resyncUI
  # chmod 666 /tmp/.resyncUI
  ```

- Restart the Oracle VM Manager service as root:

  ```
  # service ovmm restart
  ```

13.3.3 Increasing the Memory Allocated to Oracle WebLogic Server

In environments that host thousands of virtual machines and where large data sets exist, performance issues can occur with Oracle VM Manager, such as garbage collection taking a long time or out of memory conditions arise. Likewise, Oracle VM Manager can generally be slow to respond and a significant performance degradation occurs. To resolve these performance issues, you can increase the amount of memory that is allocated to Oracle WebLogic Server.

Note

You should not increase the memory allocation to the maximum limit that is available to the host server. As a guideline, you should leave at least 2GB available memory for the host operating system and other services.

To increase the memory allocated to Oracle WebLogic Server, do the following:

1. Start an ssh session to the Oracle VM Manager *host computer* as the *root* user.
2. Open the following file for editing: `/etc/sysconfig/ovmm`
3. Specify the amount of memory allocated to Oracle WebLogic Server as the value for the `JVM_MEMORY_MAX` property.

   Note

   The default is `JVM_MEMORY_MAX=4096m`.

4. Save and close `/etc/sysconfig/ovmm`.
5. Restart the Oracle VM Manager service.

   ```
   # service ovmm restart
   ```

13.3.4 No File Systems Found When Searching a Storage Server

On storage servers that have a very large number of file systems available, the UI may time out while refreshing the list of available file systems, resulting in a `No File Systems Available` message. This usually means that the time out value is set too low for the number of file systems that the UI needs to refresh. To resolve this, change the settings for the **Refresh Timeout Value** in the **Preferences Pane** on the **Tools and Resources** tab in the Oracle VM Manager Web Interface to increase the timeout value.

See the *Preferences* section in the Oracle VM Manager Online Help for more information on Oracle VM Manager UI preferences.

13.3.5 Cannot Discover Servers to Oracle VM Manager Due To Time Differences

Oracle VM Manager uses certificate-based authentication to maintain secure communication with the Oracle VM Agent that runs on each Oracle VM Server instance. This means that the system time on each Oracle VM Server must be within the bounds of the certificate *valid from* and *valid to* timestamps, or certificate validity is challenged and Oracle VM Manager is unable to authenticate to the Oracle VM Agent.

In most instances, this is not a problem, since servers are automatically configured to use the Oracle VM Manager as an NTP server as soon as ownership is taken. However, during server discovery the Oracle VM Manager uses a password to perform its initial authentication against a server and to provide its certificate for ongoing communication. During this phase of discovery, a check is performed to ensure that the system time on the Oracle VM Server is within the bounds required for certificate authentication to take place. If this is not the case, discovery fails and an error message is returned:

```
OVMAPI_4024E: Cannot take ownership of server myserver5.virtlab.info because the date
and time on the server are outside the valid range for the manager's SSL certificate.
The certificate is valid from 10/24/13 7:37 PM to 10/25/23 7:37 PM, but server's timestamp
```

```
is 9/28/13 8:14 PM.  Please verify the server's NTP and time settings.
```

In this situation, it is necessary that you access the affected Oracle VM Server directly and update its system time manually before attempting to rediscover the server using Oracle VM Manager. Once Oracle VM Manager has completed discovery and is able to take ownership of the server, its NTP configuration is updated automatically and it remains synchronized with the Oracle VM Manager host.

13.3.6 Cannot Create a Clustered Server Pool on a Disk that already has an OCFS2 File System

If you attempt to create a clustered server pool using a disk located on a storage device that already contains an OCFS2 file system, the Oracle VM Agent on the server detects the file system and refuses to overwrite it. This is normal behavior and protects you from accidentally setting up two OCFS2 file systems with matching UUIDs on the same disk, leading to instability and unexpected behavior.

If you are certain that the existing OCFS2 file system that is already present on the disk is no longer in use by any other server pool or repository, you can clean the disk by connecting to the Oracle VM Server and issuing the following command:

```
dd if=/dev/zero of=/dev/mapper/360a98000433468704234786f36394763 bs=1M count=256
```

Replace `/dev/mapper/360a98000433468704234786f36394763` with the correct path to the disk device where you are creating the new server pool cluster.

Warning

Using `dd` is data destructive. Make sure you are certain about the disk device name and that the OCFS2 file system that you are deleting is no longer in use. It is advisable to make backups of any data that exists on the disk that you are editing before proceeding. This operation should be performed by a skilled systems administrator.

13.3.7 Cannot Create a Repository on a Device that has Partitions

A repository cannot be hosted on a physical disk that has pre-existing partitions. If you attempt to create a repository on a disk that already has a partition, an error is generated notifying you that the backing device is not allowed to contain partitions. If you are intent on creating a repository on the selected disk, you must delete any pre-existing partitions. This may require you to directly access the Oracle VM Server where the disk is mounted and to manually remove the partition objects on the disk using the `fdisk` command. For example:

```
# fdisk /dev/sdb


WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
         switch off the mode (command 'c') and change display units to
         sectors (command 'u').

Command (m for help): p

Disk /dev/sdb: 500.1 GB, 500107862016 bytes
255 heads, 63 sectors/track, 60801 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x0001e791

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1        6528    52428800   83  Linux
```

```
Command (m for help): d
Partition number (1): 1

Command (m for help): w
```

Warning

Using `fdisk` is data destructive. Make sure you are certain about the disk device name and partitions that you are deleting. It is advisable to make backups of any data that exists on the disk that you are editing before proceeding. This operation should be performed by a skilled systems administrator.

Usually, restarting the affected Oracle VM Server after performing these operations is advisable. In the case of an iSCSI disk, connections to targets need to be re-initiated. In all cases, the storage needs to be refreshed. Simply restarting the Oracle VM Server ensures that all necessary actions are performed.

Once the Oracle VM Server has restarted, you may attempt to recreate the repository.

13.3.8 Removing Oracle VM Template Configuration Packages on Oracle Linux 5 Hosts

If you are running Oracle VM Manager on an Oracle Linux 5 host, removing all `ovm-template-config` related RPM packages in a single command results in an error and some packages are not successfully removed.

To resolve this, you must use two yum erase operations and ensure that `ovm-template-config` is removed last, as follows:

```
#yum erase ovm-template-config-authentication \
            ovm-template-config-datetime \
            ovm-template-config-firewall \
            ovm-template-config-network \
            ovm-template-config-selinux \
            ovm-template-config-ssh \
            ovm-template-config-system \
            ovm-template-config-user

#yum erase ovmd libovmapi xenstoreprovider ovm-template-config
```

13.3.9 Unable to Manage Oracle VM Server for x86 at Release 3.2.10 or 3.2.11, and Oracle VM Agent for SPARC at Release 3.3.1

As of Oracle VM Release 3.4.5, Oracle VM Manager uses the TLS version 1.2 (TLSv1.2) protocol for all connections for security reasons. As a result, management of Oracle VM Server for x86 at Release 3.2.10 or 3.2.11, and Oracle VM Agent for SPARC at Release 3.3.1, is not possible by default. As a workaround, you must enable the TLSv1 protocol, which is less secure.

For instructions on how to do this, see *Enabling the TLS Version 1 Protocol* in the *Oracle VM Installation and Upgrade Guide* .

13.4 Troubleshooting Virtual Machines

The section contains information on known issues you may encounter when creating or using *virtual machine*, and explains how to resolve them.

13.4.1 Setting the Guest's Clock

PVM guests may perform their own system clock management, for example, using the NTPD (Network Time Protocol daemon), or the *hypervisor* may perform system clock management for all guests.

You can set *paravirtualized* guests to manage their own system clocks by setting the *xen.independent_wallclock* parameter to 1 in the /etc/sysctl.conf file. For example:

```
"xen.independent_wallclock = 1"
```

If you want to set the hypervisor to manage paravirtualized guest system clocks, set *xen.independent_wallclock* to 0. Any attempts to set or modify the time in a guest will fail.

You can temporarily override the setting in the /proc file. For example:

```
"echo 1 > /proc/sys/xen/independent_wallclock"
```

Note

This setting does not apply to *hardware virtualized* guests.

13.4.2 Wallclock Time Skew Problems

Additional parameters may be needed in the boot loader (grub.conf) configuration file for certain operating system variants after the guest is installed. Specifically, for optimal clock accuracy, Linux guest boot parameters should be specified to ensure that the *pit* clock source is utilized. Adding *clock=pit nohpet nopmtimer* for most guest will result in the selection of *pit* as the clock source for the guest. Published templates for Oracle VM include these additional parameters.

Proper maintenance of virtual time can be tricky. The various parameters provide tuning for virtual time management and supplement, but do not replace, the need for an *ntp* time service running within guest. Ensure that the ntpd service is running and that the /etc/ntp.conf configuration file is pointing to valid time servers.

13.4.3 Mouse Pointer Tracking Problems

If your mouse pointer fails to track your cursor in a VNC Viewer session in a hardware virtualized guest, add the following to the Oracle VM Server configuration file located at /etc/xen/xend-config.sxp to force the device model to use absolute (tablet) coordinates:

```
usbdevice='tablet'
```

Restart the Oracle VM Server for the changes to take effect. You may need to do this for each Oracle VM Server in the *server pool*.

13.4.4 Cloning Virtual Machine from Oracle VM 2.*x* Template Stuck in Pending

When creating a virtual machine from an Oracle VM 2.*x* template, the *clone* job fails with the error:

```
OVMAPI_9039E Cannot place clone VM: template_name.tgz, in Server Pool: server-pool-uuid.
That server pool has no servers that can run the VM.
```

This is caused by a network configuration inconsistency with the vif = ['bridge=xenbr0'] entry in the virtual machine's configuration file.

To resolve this issue, remove any existing networks in the *virtual machine template*, and replace them with valid networks which have the Virtual Machine role. Start the clone job again and the virtual machine clone is created. Alternatively, remove any existing networks in the template, restart the clone job, and add in any networks *after* the clone job is complete.

13.4.5 Hardware Virtualized Guest Stops

When running hardware virtualized guests, the *QEMU* process (qemu-dm) may have its memory usage grow substantially, especially under heavy I/O loads. This may cause the hardware virtualized guest to stop as it runs out of memory. If the guest is stopped, increase the memory allocation for *dom0*, for example from 512 MB to 768 MB. See Section 1.6, “Changing the Memory Size of the Management Domain” for information on changing the dom0 memory allocation.

13.4.6 Migrating Virtual Machines

You cannot *migrate virtual machines* on computers with hardware that is not identical. To migrate virtual machines, you must have hardware that is the same make and model and the CPU must be in the same CPU family.

Virtual machines can be live migrated between instances of Oracle VM Server that are at the same release or later. For virtual machines running on an x86 platform, a rule exception is generated if you attempt to live migrate a virtual machine to an Oracle VM Server with an earlier release than the Oracle VM Server where the virtual machine is running.

13.4.7 Recovering From A Failed Local Virtual Machine Migration

In the event where a virtual machine hosted on a local repository is live migrated and the migration source, or target, Oracle VM Server becomes unavailable during the migration, Oracle VM Manager attempts to perform a rollback of the operation. This rollback process brings the original version of the virtual machine back online on the source Oracle VM Server and then performs a cleanup operation on the target Oracle VM Server when it becomes available again. This cleanup process involves killing the paused virtual machine that may have been copied to the target Oracle VM Server and then cleaning the target repository of virtual disks, virtual machine configurations and temporary files. Finally a repository refresh is performed on the repository on the source server to ensure that everything is in order.

Before the cleanup operation is triggered, an event is created within Oracle VM Manager to indicate that the migration job has failed or been aborted and to track the rollback process. When the event is generated within Oracle VM Manager, it is set with a 'WARNING' status. The rollback process is generated as a set of up to three different jobs that are each given a timeout period of 15 minutes, and which are triggered to attempt to run every 10 seconds. If these jobs succeed, Oracle VM Manager acknowledges the event. If the jobs all timeout, Oracle VM Manager still acknowledges the event, but a second user-acknowledgeable event is created with 'WARNING' status to indicate that the rollback failed. Depending on the cause of the rollback failure, Oracle VM Manager might also create user-acknowledgeable events with 'CRITICAL' status.

Because jobs are usually performed sequentially, it may take a total of 45 minutes before the entire rollback process times out and the new event indicating rollback failure is generated. The rollback failure event is also logged in the the log file `/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs/AdminServer.log` on the Oracle VM Manager host.

The information in the rollback failure event contains the rollback plan that Oracle VM Manager attempted to follow to cleanup a failed virtual machine migration. This event can be viewed using the `getEventsForObject` command with the Oracle VM Manager Command Line Interface, by viewing the events associated with the virtual machine within the Oracle VM Manager Web Interface or via the Oracle VM Web Services API.

The following content represents the typical output displayed within the description field for a rollback failure event:

```
Live VM Migration With Storage, started at 2015-11-04 09:51:13,205
```

```
    VM: [VirtualMachineDbImpl] 0004fb0000060000c71d489702c240b3<978> (MyVM)

    Source server: [ServerDbImpl] 30:30:38:37:30:32:58:4d:51:34:35:30:30:37:4c:52<386> (ovs216)
    Target server: [ServerDbImpl] 30:30:38:37:30:32:58:4d:51:34:35:30:30:38:58:42<238> (ovs215)

    Source repository: [RepositoryDbImpl] 0004fb0000030000c4fca9a963e2706c<479> (r216)
    Target repository: [RepositoryDbImpl] 0004fb0000030000f9ba13d5063e330a<382> (r215)

    Source vDisks to be migrated:
        /OVS/Repositories/0004fb0000030000c4fca9a963e2706c/VirtualDisks/
           0004fb00001200005213553a5bba24f.img

Migration job has failed or was aborted.
VM's server has been set back to: (ovs216)
Source vDisk files have been retained.

Constructed the following post-migration completion plan at 2015-11-04 09:51:36,686

    VM to be killed on server: (ovs215)

    To be deleted on target server (ovs215):
            vDisk: /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualDisks/
              0004fb00001200005213553a5bba24f.img
         tmp file: /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualDisks/
              tmp_dest.0004fb00001200005213553a5bba24f.img

    Also to be deleted on target server (ovs215):
         cfg file: /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
            0004fb0000060000c71d489702c240b3/vm.cfg
        directory: /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
            0004fb0000060000c71d489702c240b3
         tmp file: /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
            tmp_dest.0004fb0000060000c71d489702c240b3

    Source repository (r216) must be refreshed.
```

Note that the description of the event provides detailed information about the migration process and indicates that the migration job has failed. The message explains that the virtual machine is set back to run on the source server and that the source virtual disks have been retained. This means that the virtual machine may either be running or stopped on the source server, but from the perspective of Oracle VM Manager, the location of the virtual machine has been reverted. Most significantly, the output contains a 'post-migration completion plan'. This plan provides a full breakdown of the steps that must be performed to roll the environment back to its original state.

If an event like this appears for a failed migration of a locally hosted virtual machine, you must manually perform the rollback steps on the target server when it next becomes available. It is very important that you ensure that the rollback steps are performed on the systems indicated in the post-migration completion plan. Performing any of these steps on another server could have detrimental effects and could result in virtual machine corruption.

Kill the Virtual Machine on the Indicated Oracle VM Server

The first step in this plan involves killing the virtual machine on the indicated Oracle VM Server or servers. Depending on the state of the migration at the time that the target Oracle VM Server became unavailable, this may be require an action on either the target Oracle VM Server or both the target and source Oracle VM Servers. In some cases you may not need to perform this action on either Oracle VM Server. The appropriate action is logged in the event description.

During the migration, the virtual machine enters into a paused state as it is copied from the source Oracle VM Server to the target Oracle VM Server. Once the copy is complete, the virtual machine on the target

Oracle VM Server is not indicated within Oracle VM Manager in any way, as this would conflict with the virtual machine with the identical UUID that is located on the original source Oracle VM Server. This transition is performed within Oracle VM Manager when the migration is complete. As a result two virtual machines with identical UUIDs may exist within the environment for the period of the migration. If the target server goes offline at any point during the migration, it is frequently the case that at least one of these virtual machines must be killed off to prevent conflict. Since the representation of the virtual machine within Oracle VM Manager is not reliable until the rollback has been completed, it is necessary that you must perform the kill operation directly on the indicated Oracle VM Server. This is usually done over SSH as the root user, using the following command:

```
ovs-agent-rpc stop_vm "''" "'0004fb0000060000c71d489702c240b3'" "True"
```

Note that *0004fb0000060000c71d489702c240b3* should match the UUID of the virtual machine that you were originally migrating. Also pay attention to the quotes in each of the arguments presented here. The first argument for this command is empty, so a pair of single quotes are enclosed in a pair of double-quotes. The second argument is the UUID of the virtual machine that you intend to kill and is represented as enclosed in a pair of single quotes within a pair of double-quotes. Finally, the last argument is used to force the action and contains the text *True* enclosed in a pair of double-quotes.

Note that you should use this command to stop the virtual machine because it helps to identify the correct virtual machine domain to destroy, it maintains the integrity of your environment and logs any actions carried out on the virtual machine. Do not attempt to use Xen hypervisor tools to perform any actions on the virtual machine directly without explicit instruction from an Oracle Support representative.

Remove any Virtual Disks from the Repository on the Target Oracle VM Server

A live migration of a virtual machine that is hosted on local storage also requires that any virtual disks are copied from the repository hosted on the source server across to the repository of the target server. Therefore, it is necessary that you manually delete any of these files from the repository hosted on the target Oracle VM Server to clean the environment. To do this, you must SSH to the target Oracle VM Server and delete the files listed in the plan returned in the event description. For example:

```
rm -f /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualDisks/
    0004fb0000120000005213553a5bba24f.img
rm -f /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualDisks/
    tmp_dest.0004fb0000120000005213553a5bba24f.img
```

Remove the Virtual Machine Configuration from the Repository on the Target Oracle VM Server

The virtual machine configuration for the virtual machine is also copied from the repository hosted on the source server across to the repository of the target server during the migration. Therefore, it is necessary that you manually delete any of these files and directories from the repository hosted on the target Oracle VM Server to clean the environment. To do this, you must SSH to the target Oracle VM Server and delete the files listed in the plan returned in the event description. For example:

```
rm -f /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
    0004fb0000060000c71d489702c240b3/vm.cfg
rm -rf /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
    0004fb0000060000c71d489702c240b3
rm -f /OVS/Repositories/0004fb0000030000f9ba13d5063e330a/VirtualMachines/
    tmp_dest.0004fb0000060000c71d489702c240b3
```

Refresh the Repository on the Source Oracle VM Server

During the migration process, Oracle VM Manager updates its model of the source and target repositories hosted on each Oracle VM Server to match the environment as it would be after the migration is complete. It does not revert this representation unless an automated rollback is achieved. If the rollback has failed and you have performed manual steps to revert your environment to its original state, you must also refresh the repository within Oracle VM Manager so that the model accurately reflects the state of the repository. You can either do this using the Oracle VM Manager Web Interface or you can use the Oracle VM Manager Command Line Interface directly. For example:

```
ssh -l admin localhost -p 10000 refresh repository name="r216"
```

At this point, your environment should be completely reverted.

13.4.8 Migrating Large Hardware Virtualized Guest Results in CPU Soft Lock

On some hardware, such as the SUN FIRE X4170 M2 Server, migration of very large virtual machines using hardware virtualization can result in a soft lockup causing the virtual machine to become unresponsive. This lock is caused when the migration causes the virtual machine kernel to lose the clock source. Access to the console for the virtual machine shows a series of error messages similar to the following:

```
BUG: soft lockup - CPU#0 stuck for 315s! [kstop/0:2131]
```

To resolve this, the virtual machine must be restarted and the *`clocksource=jiffies`* option should be added to the HVM guest kernel command line, before rebooting the virtual machine again.

⚠

Important

This option should only be used on HVM guest systems that have actually resulted in a CPU soft lock.

13.4.9 Hardware Virtualized Guest Devices Not Working as Expected

Some devices, such as sound cards, may not work as expected in hardware virtualized guests. In a hardware virtualized guest, a device that requires physical memory addresses instead uses virtualized memory addresses, so incorrect memory location values may be set. This is because DMA (Direct Memory Access) is virtualized in hardware virtualized guest.

Hardware virtualized guest operating systems expect to be loaded in memory starting somewhere around address 0 and upwards. This is only possible for the first hardware virtualized guest loaded. Oracle VM Server virtualizes the memory address to be 0 to the size of allocated memory, but the guest operating system is actually loaded at another memory location. The difference is fixed up in the shadow page table, but the operating system is unaware of this.

For example, a sound is loaded into memory in a hardware virtualized guest running Microsoft Windows™ at an address of 100 MB may produce garbage through the sound card, instead of the intended audio. This is because the sound is actually loaded at 100 MB *plus* 256 MB. The sound card receives the address of 100 MB, but it is actually at 256 MB.

An IOMMU (Input/Output Memory Management Unit) in the computer's memory management unit would remove this problem as it would take care of mapping virtual addresses to physical addresses, and enable hardware virtualized guests direct access to the hardware.

13.4.10 Paravirtualized Guest Disk Devices are Not Recognized

If you opt to create a PVHVM or PVM, you must ensure that all disks that the virtual machine is configured to use are configured as paravirtual devices, or they may not be recognized by the virtual machine. If you discover that a disk or virtual cdrom device is not being recognized by your virtual machine, you may need to edit the `vm.cfg` file for the virtual machine directly. To do this, determine the UUID of the virtual machine, and then locate the configuration file in the repository, for example on an Oracle VM Server:

```
# vi /OVS/Repositories/UUID/vm.cfg
```

Locate each `disk` entry that contains a hardware device such as `hda`, `hdb`, or `hdc` and replace with an `xvd` mapping, such as `xvda`, `xvdb`, `xvdc` etc.

Restart the virtual machine with the new configuration, to check that it is able to discover the disk or virtual cdrom device.

13.4.11 Cannot Create a Virtual Machine from Installation Media

When creating a virtual machine, the following message may be displayed:

```
Error: There is no server supporting hardware virtualization in the selected server pool.
```

To resolve this issue, make sure the Oracle VM Server supports hardware virtualization. Follow these steps to check:

a. Run the following command to check if hardware virtualization is supported by the CPU:

   ```
   # cat /proc/cpuinfo |grep -E 'vmx|smx'
   ```

   If any information that contains `vmx` or `smx` is displayed, it means that the CPU supports hardware virtualization. Here is an example of the returned message:

   ```
   flags : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 clflush dts acpi mmx fxsr
   sse sse2 ss ht tm pbe nx lm constant_tsc pni monitor ds_cpl vmx est tm2 cx16 xtpr lahf_lm
   ```

   Note

   The `/proc/cpuinfo` command only shows virtualization capabilities starting with Linux 2.6.15 (Intel®) and Linux 2.6.16 (AMD). Use the `uname -r` command to query your kernel version.

b. Make sure you have enabled hardware virtualization in the BIOS.

c. Run the following command to check if the operating system supports hardware virtualization:

   ```
   # xm info |grep hvm
   ```

   The following is an example of the returned message:

   ```
   xen_caps : xen-3.0-x86_64 xen-3.0-x86_32p hvm-3.0-x86_32 hvm-3.0-x
   ```

If the CPU does not support hardware virtualization, use the paravirtualized method to create the virtual machine. See the *Servers and VMs Tab* section in the Oracle VM Manager Online Help for information on creating a *paravirtualized virtual machine*.

13.4.12 Cannot Change CD in the Virtual Machine

To change the CD in a virtual machine:

a. Unmount the first CD:

   ```
   # umount mount-point
   ```

b. Select the second ISO file, and click **Change CD**.

c. Mount the second CD:

   ```
   # mount /dev/cdrom mount-point
   ```

13.4.13 Generating Guest Dump Files on Oracle VM Server (x86)

The Xen hypervisor makes it possible to generate a core dump file for a virtual machine in the case that it crashes. This file can be useful for debugging and support purposes. Core dump files can be large and to avoid overwriting files, each file is named uniquely. When this facility is enabled, core dump files are saved to `/var/xen/dump` on the Oracle VM Server where the virtual machine was running when it crashed. This can rapidly use up available disk space on the dom0 system partition. If you enable this facility, you must ensure that enough disk space is available at this path on the Oracle VM Server, either by mounting an additional disk at this path, or by creating a symbolic link for this path to point to an alternate location with plenty of available disk space.

By default, this facility is disable at a system-wide level. It is possible to change this behavior by editing `/etc/xen/xend-config.sxp` directly and changing the lines:

```
# Whether to enable core-dumps when domains crash.
#(enable-dump no)
```

to:

```
# Whether to enable core-dumps when domains crash.
(enable-dump yes)
```

After making this change, you must reboot the Oracle VM Server for the change to take effect. Manually editing the global Xen configuration parameters on an Oracle VM Server is not supported by Oracle.

It is possible to override the system-wide behavior by setting this parameter directly in the `vm.cfg` for each individual virtual machine. This is the preferred approach to generating dump files, as it allows you to limit core dumps to only those virtual machines that you are interested in debugging. Therefore, this configuration option can be controlled for each virtual machine from within Oracle VM Manager. You can set this option by configuring the `Restart Action On Crash` option for a virtual machine. See the *Servers and VMs Tab* section in the Oracle VM Manager Online Help for more information on this parameter.

If you change the `Restart Action On Crash` option for a virtual machine, you must stop the virtual machine and then start it again before the change takes effect. This is different to restarting the virtual machine, as the `vm.cfg` configuration file for the virtual machine is only read by the Xen hypervisor when the virtual machine is started. If you have made the configuration change but have not properly restarted the virtual machine, a crash and reboot does not automatically cause the configuration option to take effect.

To test whether or not the core dump facility is working properly for a virtual machine, you may be able to directly trigger a crash by logging into the virtual machine and obtaining root privileges before issuing the following command:

```
# echo c >/proc/sysrq-trigger
```

This command assumes that the operating system on the virtual machine is Linux-based and that the System Request trigger is enabled within the kernel. After you have triggered the crash, check `/var/xen/dump` on the Oracle VM Server where the virtual machine was running to view the dump file.

13.4.14 Tuning a Linux-based Virtual Machine for Handling Storage Migration

When a virtual machine is hosted in a repository using local storage on the Oracle VM Server where it is running, migration of that virtual machine to another Oracle VM Server and repository requires that I/O on affected disks is not excessively high. If you are running an application that has high I/O during a migration, it may cause the guest or the application to hang. Steps can be taken to mitigate against this on guests that are running a Linux operating system by tuning virtual memory caching parameters within the guest kernel and by reducing the ext4 journaling commit frequency on any guests that may be running file sytems that are formatted with ext4.

Tuning virtual memory caching. On the guest command line, as the root user, you can tune the cache by using the `sysctl` command to set a number of kernel parameters. Oracle recommends that you reduce the cache size to 5% of the system memory (the default value is 10) and reduce the time that a memory page can remain dirty until it is flushed to around 20 seconds (the default is 30 seconds). You can do this temporarily by running the following commands:

```
# sysctl -w vm.dirty_background_ratio=5
# sysctl -w vm.dirty_expire_centisecs=2000
```

Alternatively edit `/etc/sysctl.conf` and add these lines:

```
vm.dirty_background_ratio=5
vm.dirty_expire_centisecs=2000
```

When you have done this, you can load these values into the kernel by running `sysctl -p`.

Tuning ext4 journaling. If the guest is using any filesystems that are formatted to use ext4, the journaling commit process may be affected by a migration. To protect against this, decrease the amount of time between journal commits and ensure that commits are performed asynchronously. To do this, you should tune your mount parameters for any ext4 filesystem that you have mounted within the guest. For example when mounting an ext4 formatted filesystem you might use the following options:

```
# mount -o commit=5,journal_async_commit /dev/xvdd /vdisk3
```

To perform this effectively for all ext4 mounts, you may need to edit your `/etc/fstab`.

Glossary

C

clone

The action or result of making an exact copy of an object. The object may be a virtual machine, virtual machine template, ISO file, or virtual disk. Cloning is similar to copying and maintains the integrity of the original object, while creating a new object based on the original. A clone customizer may be used to define cloning options to specify details of where the object components may reside when cloned, such as in a different storage repository.

D

dom0

An abbreviation for *domain zero*. The management domain with privileged access to the hardware and device drivers. Dom0 is the first domain started at boot time. Dom0 has more privileges than domU. It can access the hardware directly and can manage the device drivers for other domains. It can also start new domains.

See Also: *control domain*

domU

An unprivileged domain with no direct access to the hardware or device drivers. Each domU is started by dom0.

E

events

Events are used to register status information of "objects" within Oracle VM Manager for future reference or to make problems easier to trace back. Events are often, though not always, related to *jobs* that are initiated within Oracle VM Manager. For instance, when a job fails, an event is generated. Events can also be triggered through changes in the environment such as server crashes or storage disconnects. Therefore, events are used to alert you to potential problems that may need your attention.

Events are categorized by severity. Most events will be informational, but they can also be warnings or errors. If an event has an error level severity, you need to acknowledge the error event to clear the error and to perform further operations on the object that generated the error.

See Also: *jobs*

G

guest

A guest operating system that runs within a domain in Oracle VM Server. A guest may be paravirtualized or hardware virtualized. Multiple guests can run on the same Oracle VM Server.

H

hard partitioning

Hard partition, or CPU pinning, is the act of binding a virtual machine to one or more physical CPUs or cores. This prevents software within the virtual machine from running on any cores other than those specified for the virtual machine. By default, Oracle VM takes advantage of distributed resource scheduling, which allows a virtual machine to use all cores on an Oracle VM Server as required. In some situations, such as the requirement to conform with Oracle licensing policies for partitioned environments, it may be desirable to implement hard partitioning.

Hard partitioning can result in restrictions on live migration, DRS and DPM.

host computer

The physical computer on which the software is installed. Typically used to refer to either the computer on which Oracle VM Server or Oracle VM Manager is running.

hypervisor

A hypervisor, also called a monitor or Virtual Machine Manager (VMM), is a layer which abstracts the virtual hardware from the real hardware. As such it is the only privileged entity in the system which has full access to real hardware resources. It controls only the most basic resources of the system, including CPU and memory usage, privilege checks, and hardware interrupts.

Hosted hypervisors are designed to run within a traditional operating system. In other words, a hosted hypervisor adds a distinct software layer to the host operating system, and the guest operating system becomes a third software level above the hardware and the host-based hypervisor. A well-known example of a hosted hypervisor is Oracle VM VirtualBox. Others include VMware Server and Workstation, Microsoft Virtual PC, KVM, QEMU, and Parallels.

Native hypervisors are software systems that run directly on the host's hardware to control the hardware, and to monitor the guest operating systems. The guest operating system runs on a separate level above the hypervisor. Examples of this type of virtualization architecture are Oracle VM, Microsoft Hyper-V, VMware ESX, and Xen.

M

messaging

Oracle VM supports a messaging system that enables communication between the Oracle VM Manager and a guest running within a virtual machine on any Oracle VM Server, as long as the guest has the Oracle VM Guest Additions installed. This messaging system works by sending key/value pairs between the guest and Oracle VM Manager via a secured connection. Messaging allows greater administrative control over virtual machines and facilitates remote and automated configuration of virtual machines as they are started.

migrate

The act of moving a virtual machine from one Oracle VM Server to another, or to the Unassigned Virtual Machines folder. Technically, a migration can only be performed on a running virtual machine, however the Oracle VM Manager Web Interface and Oracle VM Manager Command Line Interface may combine multiple operations to make it appear that you can perform a migration on either a running or a stopped virtual machine.

multipath

The technique of creating more than one physical path between the server CPU and its storage devices. It results in better fault tolerance and performance enhancement. Oracle VM supports multipath I/O out of the box. Oracle VM Servers are installed with multipathing enabled because it is a requirement for SAN disks to be discovered by Oracle VM Manager

O

OCFS2

Oracle Cluster File System (OCFS2) is a general-purpose shared-disk cluster file system for Linux capable of providing both high performance and high availability. OCFS2 is developed by Oracle and is integrated within the mainstream Linux kernel. OCFS2 is used within Oracle VM to facilitate clustered server pools, storage of virtual machine images and for the purpose of allowing guests to share the same file system.

A clustered server pool always uses an OCFS2 file system to store the cluster configuration and to take advantage of OCFS2's heartbeat facility. There are two types of heartbeats used in OCFS2 to ensure high availability:

- The disk heartbeat: all Oracle VM Servers in the cluster write a time stamp to the server pool file system device.
- The network heartbeat: all Oracle VM Servers communicate through the network to signal to each other that every cluster member is alive.

These heartbeat functions exist directly within the kernel and are fundamental to the clustering functionality that Oracle VM offers for server pools. The server pool file system should be stored on a separate NFS server or on a small LUN if possible, as OCFS2's heartbeat facility can be disturbed by intensive I/O operations taking place on the same physical storage.

A storage repository configured on a LUN-based repository must be linked to a clustered server pool due to the nature of the OCFS2 file system. As a result, LUN-based repositories cannot be shared between multiple server pools, although it is possible to move an OCFS2 repository from one server pool to another.

For more information on OCFS2, please refer to https://oss.oracle.com/projects/ocfs2/.

Oracle VM Agent

An application installed with Oracle VM Server. The Oracle VM Agent receives and processes management requests, and provides event notifications and configuration data to the Oracle VM Manager. Oracle VM Manager manages the virtual machines running on Oracle VM Server by communicating with Oracle VM Agent. It contains three components: master Oracle VM Server, Utility Server, and Virtual Machine Server.

Oracle VM Manager

Oracle VM Manager is the management platform, which offers an easy-to-use, web-browser interface as well as a command-line interface (CLI). Oracle VM Manager tracks and manages the resources available in your virtual environment and allows you to easily manage Oracle VM Server pools. Oracle VM Manager lets you manage the virtual machine life cycle, including creating virtual machines from templates or from installation media, deleting, powering off, uploading, deployment and live migration of virtual machines. Oracle VM Manager also lets you manage resources including ISO files, templates and shared virtual disks.

Oracle VM Server

A self-contained virtualization environment designed to provide a lightweight, secure, server-based platform for running virtual machines. The Oracle VM Server comprises a hypervisor and a privileged domain (called dom0) that allow multiple domains or guest operation systems (such as Linux, Solaris, and Windows) to run on one physical machine. Includes Oracle VM Agent to enable communication with Oracle VM Manager.

The Oracle VM Server for x86 incorporates an open source Xen hypervisor component, which has been customized and optimized to integrate into the larger, Oracle - developed virtualization server. The Oracle VM Server for x86 is also responsible for access and security management and generally acts as the server administrative entity, because the hypervisor’s role is limited.

On Oracle VM Server for SPARC systems, the SPARC hypervisor is built into the SPARC firmware and is generally referred to as the Logical Domains Manager. As with the Xen hypervisor, each virtual machine is securely executed on a single computer and runs its own guest Oracle Solaris operating system

P

paravirtualized machine (PVM)

A virtual machine with a kernel that is recompiled to be made aware of the virtual environment. Runs at near native speed, with memory, disk and network access optimized for maximum performance.

Paravirtualized guests use generic, idealized device drivers, which are part of the guest’s OS. The I/O operations using these generic device drivers are mapped to the real device drivers in dom0. The generic, abstracted drivers in the guest seldom change and provide excellent guest stability. The dom0 domain, alternatively, can use the native hardware vendor drivers, and the guests can safely migrate to another dom0 with slightly different drivers.

For other resources such as CPU and memory, paravirtualized kernels make special “hypercalls” to the Xen hypervisor. These hypercalls provide better performance by reducing the number of instructions and context switches required to handle an incoming request. By contrast, on an emulated (hardware virtualized) guest, driver requests engage the guest’s interrupt handler, increasing the I/O operation overhead.

Q

QEMU

Also referred to as qemu-dm, which is the process name. The virtualization process which allows full virtualization of a PC system within another PC system.

S

server pool

Server pools logically organize one or more Oracle VM Servers into groups where virtual machines can run.

Each server pool can have up to 32 physical servers. Each Oracle VM Server can be a member of only one server pool. The server pool is the operational unit of Oracle VM. Policies are configured and enforced at the server pool level.

A minimum cluster of three Oracle VM Server nodes in each server pool is strongly recommended for high availability. If one node in the cluster experiences a hardware failure or is shut down for maintenance, failover redundancy is preserved with the other two nodes. Having a third node in the cluster also provides reserve capacity for production load requirements.

V

virtual disk

A file or set of files, usually on the host file system although it may also be a remote file system, that appears as a physical disk drive to the guest operating system.

virtual machine (VM)

A guest operating system and the associated application software that runs within Oracle VM Server. May be paravirtualized or hardware virtualized machines. Multiple virtual machines can run on the same Oracle VM Server.

virtual machine template

A template of a virtual machine. Contains basic configuration information such as the number of CPUs, memory size, hard disk size, and network interface card (NIC). Create virtual machines based on a virtual machine template using Oracle VM Manager.

W

WebLogic

Oracle WebLogic Server is a platform that includes an application server that can run java applications within a web-based framework. Oracle VM Manager runs as an application within Oracle WebLogic Server, taking advantage of many of Oracle WebLogic Server's many features to deliver a robust web UI through which Oracle VM can be fully managed.

The installation process behind Oracle VM Manager automatically installs and configures Oracle WebLogic Server on the system where Oracle VM Manager is installed. During this process, a *weblogic* user is set up within Oracle WebLogic Server to manage Oracle WebLogic Server configuration and to administer the underlying system. An *admin* user is also set up within Oracle WebLogic Server and is given permission to access the

Oracle VM Manager application. A typical setup uses the same password for both of these users, although this is not always the case and it is possible to configure a different password for each user depending on your requirements.

In general, users of the Oracle VM Manager application should avoid attempting to access the underlying Oracle WebLogic Server, or to change any configuration parameters here without guidance from Oracle Support.

Index