

Oracle® VM

Utilities Guide for Release 3

ORACLE®

E35591-03
June 2012

Oracle® VM: Utilities Guide for Release 3

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Abstract

Document generated on: 2012-06-01 (revision: 1531)

Table of Contents

Preface	v
1. Audience	v
2. Documentation Accessibility	v
3. Related Documents	v
4. Command Syntax	vi
5. Conventions	vi
1. Oracle VM Utilities Overview	1
1.1. Downloading and Installing Oracle VM Utilities	1
1.1.1. Installing Oracle VM Utilities on Oracle VM Manager	1
1.1.2. Installing Oracle VM Utilities on Oracle Linux	2
1.1.3. Directory Structure of Oracle VM Utilities	3
1.2. Introduction to Oracle VM Utilities	3
1.3. Introduction to Oracle VM Guest Additions	4
1.4. Introduction to Oracle VM Administrator Tool	4
1.5. Introduction to Oracle VM Diagnostic Capture	4
2. Using the Oracle VM Utilities	5
2.1. Using Oracle VM Manager Control	5
2.1.1. Command Line Options and Parameters of ovm_managercontrol	5
2.1.2. Examples of ovm_managercontrol Usage	6
2.2. Using Oracle VM Server Control	7
2.2.1. Command Line Options and Parameters of ovm_servercontrol	7
2.2.2. Examples of ovm_servercontrol Usage	8
2.3. Using Oracle VM Virtual Machine Control	10
2.3.1. Command Line Options and Parameters of ovm_vmcontrol	10
2.3.2. Examples of ovm_vmcontrol Usage	11
2.4. Using Oracle VM Retrieve Disk	12
2.4.1. Command Line Options and Parameters of ovm_vmdisk	12
2.4.2. Examples of ovm_vmdisk Usage	12
2.5. Using Oracle VM Virtual Machine Messaging	13
2.5.1. Command Line Options and Parameters of ovm_vmmessage	13
2.5.2. Examples of ovm_vmmessage Usage	14
2.6. Using Oracle VM Pool Control	14
2.6.1. Command Line Options and Parameters of ovm_poolcontrol	15
2.6.2. Examples of ovm_poolcontrol Usage	15
2.7. Using Oracle VM Repository Control	16
2.7.1. Command Line Options and Parameters of ovm_repocontrol	16
2.7.2. Examples of ovm_repocontrol Usage	17
3. Using the Oracle VM Guest Additions	19
3.1. Installing Oracle VM Guest Additions	19
3.2. Features of the Oracle VM Guest Additions	20
3.2.1. Using the Messaging Channel	20
3.2.2. Displaying the Guest IP Address	21

Preface

Table of Contents

1. Audience	v
2. Documentation Accessibility	v
3. Related Documents	v
4. Command Syntax	vi
5. Conventions	vi

The Oracle VM Utilities Guide is your reference to the administrative and diagnostic command line tools provided as an add-on or as part of the Oracle VM installation. The command line tools described in this guide are intended to help administrators perform certain tasks and collect troubleshooting information more easily. The operations made possible by these tools are typically not available through the Oracle VM Manager user interface. Usage of the tools requires root privileges on the system.

1. 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 an in depth knowledge of Oracle VM (see the [Oracle VM User's Guide](#)), and that you are familiar with Oracle Linux system administration and Linux command line operation.

2. Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

3. Related Documents

For more information, see the following documents in the Oracle VM Release 3 documentation.

- [Oracle VM Release Notes](#)
- [Oracle VM Installation and Upgrade Guide](#)
- [Oracle VM Getting Started Guide](#)
- [Oracle VM User's Guide](#)
- [Oracle VM Windows Paravirtual Drivers Installation Guide](#)

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

<http://www.oracle.com/virtualization>

4. 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:

Convention	Description
backslash \	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: <pre>dd if=/dev/rdsd/c0t1d0s6 of=/dev/rst0 bs=10b \ count=10000</pre>
braces { }	Braces indicate required items: <pre>.DEFINE {macro1}</pre>
brackets []	Brackets indicate optional items: <pre>cvtcrt <i>termname</i> [<i>outfile</i>]</pre>
ellipses ...	Ellipses indicate an arbitrary number of similar items: <pre>CHKVAL fieldname value1 value2 ... valueN</pre>
<i>italics</i>	Italic type indicates a variable. Substitute a value for the variable: <pre><i>library_name</i></pre>
vertical line	A vertical line indicates a choice within braces or brackets: <pre>FILE <i>filesize</i> [K M]</pre>

5. 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.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Chapter 1. Oracle VM Utilities Overview

Table of Contents

1.1. Downloading and Installing Oracle VM Utilities	1
1.1.1. Installing Oracle VM Utilities on Oracle VM Manager	1
1.1.2. Installing Oracle VM Utilities on Oracle Linux	2
1.1.3. Directory Structure of Oracle VM Utilities	3
1.2. Introduction to Oracle VM Utilities	3
1.3. Introduction to Oracle VM Guest Additions	4
1.4. Introduction to Oracle VM Administrator Tool	4
1.5. Introduction to Oracle VM Diagnostic Capture	4

The Oracle VM Utilities are a collection of command line scripts that allow you to perform a set of basic management tasks on Oracle VM Servers and virtual machines in an Oracle VM environment. These utilities are particularly useful to administrators who need to execute certain operations quickly and/or repeatedly. Using the command line scripts makes these tasks quicker and easier to perform.



Warning

The command line scripts in Oracle VM Utilities are provided as-is for your convenience. They are not officially and formally supported by Oracle.

1.1. Downloading and Installing Oracle VM Utilities

The Oracle VM Utilities are available for download as a `.zip` file via the Oracle VM Downloads page:

```
http://www.oracle.com/technetwork/server-storage/vm/downloads/index.html
```

The sections below describe how and where to install these utilities. An overview of the installation directory structure and files is also provided.

1.1.1. Installing Oracle VM Utilities on Oracle VM Manager

When installing Oracle VM Utilities on the same server as Oracle VM Manager, it is recommended that you put them in the software directory of the Oracle VM Manager application. The Oracle VM Utilities are bundled in a `.zip` archive. To install, simply download the archive and extract in the appropriate directory.



Note

In the example below, we downloaded the `.zip` file to a file server that is also accessible from the server where we intend to install the Oracle VM Utilities. We mount this network drive on the server file system and copy the `.zip` file into the installation directory, where we unzip it. Depending on your local configuration and preferences you may use different methods to place the downloaded file in the installation directory.

```
# cp ovm_utils_archive.zip /u01/app/oracle/ovm-manager-3/  
# cd /u01/app/oracle/ovm-manager-3  
# unzip ovm_utils_archive.zip
```

At this point, the command line scripts are ready to use. Note that these scripts execute a Java program and use the default Java VM on the host computer. On the server running Oracle VM Manager, the

appropriate Java VM is installed as part of the Oracle VM Manager application in `/u01/app/oracle/java`. No further configuration is required.

1.1.2. Installing Oracle VM Utilities on Oracle Linux

When installing Oracle VM Utilities on an Oracle Linux server or desktop computer, you may put them in the installation directory of your choice; for example: `/usr/local/bin`. The Oracle VM Utilities are bundled in a `.zip` archive. To install, simply download the archive, copy and extract in the appropriate directory.

```
# cp ovm_utils_archive.zip /usr/local/bin/oracle/
# cd /usr/local/bin/oracle
# unzip ovm_utils_archive.zip
```



Caution

Oracle VM Utilities **do not** work with the Open JDK or GNU Compiler for Java (GCJ). You must install a standard Java VM, version 1.6.x, and make sure it is defined as the default Java VM by the `JAVA_HOME` environment variable or included in the `PATH` variable on your system.

The command line scripts in Oracle VM Utilities execute a Java program and use the default Java VM on the host computer. To verify the exact path to the Java executable and the active Java version, use the following commands:

```
# which java
/usr/java
# java -version
java version "1.6.0_26"
Java(TM) SE Runtime Environment (build 1.6.0_26-b03)
Java(TM) Server VM (build 20.1-b02, mixed mode)
```

If a standard Java VM, version 1.6.x, is not available on your system, download and install the version suited for your platform from <http://java.com/en/download/>. To ensure that the appropriate Java VM version is used, set the `JAVA_HOME` and `PATH` variables as follows:

1. Edit the Oracle Linux shell profile.

```
# vi /etc/profile
- OR -
# vi /etc/bashrc
```

2. Add the lines below to the profile. If your Java path does not match `/usr/java`, replace with the actual path on your system.

```
JAVA_HOME=/usr/java
export JAVA_HOME
PATH=$PATH:$JAVA_HOME/bin
export PATH
```

3. Save the file. Log out and log back in to activate your changes.
4. Verify the Java path and version again, to make sure that the correct Java VM is used.

```
# which java
/usr/java
# java -version
java version "1.6.0_26"
Java(TM) SE Runtime Environment (build 1.6.0_26-b03)
Java(TM) Server VM (build 20.1-b02, mixed mode)
```

At this point, the command line scripts are ready to use.

1.1.3. Directory Structure of Oracle VM Utilities

When you unzip the Oracle VM Utilities archive, a subdirectory named `ovm_utils` is created, containing these utility scripts:

- `ovm_managercontrol`
- `ovm_servercontrol`
- `ovm_vmcontrol`
- `ovm_vmdisks`
- `ovm_vmmmessage`
- `ovm_poolcontrol`
- `ovm_repocontrol`

For each of these utility scripts, a *man page* is also included. These help files can be found in `.../ovm_utils/man/man8`.

```
# ls /u01/app/oracle/ovm-manager-3/ovm_utils/man/man8/  
ovm_managercontrol.8  ovm_repocontrol.8  ovm_vmcontrol.8  ovm_vmmmessage.8  
ovm_poolcontrol.8    ovm_servercontrol.8  ovm_vmdisks.8
```

Use the `man` command to display the help; for example:

```
# man man/man8/ovm_servercontrol.8
```

1.2. Introduction to Oracle VM Utilities

The Oracle VM Utilities are command line scripts used to execute certain basic operations on your Oracle VM environment. These utilities make a connection to the Oracle VM Manager host server, using the server host name and an administrative user name and password. After authentication, commands can be submitted to Oracle VM Manager from the command line.

The Oracle VM Utilities are:

- **ovm_servercontrol**: command line script to perform administrative operations on a physical Oracle VM Server
- **ovm_vmcontrol**: command line script to perform administrative operations on a virtual machine
- **ovm_vmdisks**: command line script to list virtual and raw physical disks attached to a virtual machine in order to facilitate backup
- **ovm_vmmmessage**: command line script to send and retrieve messages in the form of key/value pairs to and from a running virtual machine
- **ovm_poolcontrol**: command line script to perform administrative operations on a pool of Oracle VM Servers
- **ovm_repocontrol**: command line script to retrieve information about storage repositories and refresh their contents

Command line parameters and options, along with several examples, can be found in [Chapter 2, Using the Oracle VM Utilities](#)



Caution

Certain functions of the Oracle VM Utilities are restricted to version 3.1 and higher. If you attempt to execute an incompatible script on Oracle VM version 3.0, a warning message will be displayed to inform you that the feature is not supported.

1.3. Introduction to Oracle VM Guest Additions

The Oracle VM Utilities include a messaging tool, which allows sending key-value pairs to a virtual machine, or guest, and retrieve such messages from the guest. The `vmmessage` utility requires the Oracle VM Guest Additions to be installed on the guest side. The current edition of Oracle VM Guest Addition includes message channel and guest IP information. The Oracle VM Guest Additions allow direct integration between guest software and the virtualization layer, to assist in orchestration and automation of complex, multi-VM deployments.

A brief description of the installation, configuration and features of Oracle VM Guest Additions can be found in [Chapter 3, Using the Oracle VM Guest Additions](#)

1.4. Introduction to Oracle VM Administrator Tool

The following is listed in the help page on the command line:

```
Usage: ./ovm_admin [options]

Options:
--help: Shows this message
--createuser: Create new Oracle VM Manager admin user
--deleteuser: <admin> Delete Oracle VM Manager admin user
--listusers: List Oracle VM Manager users
--modifyuser: Modify Oracle VM Manager user password
--lockusers: <tries> Max login tries before locking account. This setting is global.
--unlockuser: <admin> Unlock user account
--modifyds: <SID> <host> <port> Modify Data Store 'OVMSDS'
--listconfig: List configuration
--rotatelogsdaily: <time> Rotate Logs Daily (HH:MM)
--rotatelogsbysize: <size> Rotate Logs By Size (KB)
```

1.5. Introduction to Oracle VM Diagnostic Capture

For diagnostic purposes, Oracle Support Services use a script called `VMPInfo3` that automatically collects vital troubleshooting information from your Oracle VM environment. For detailed information about the script, its purpose and usage, please consult the support note with [Doc ID 1364933.1](#). You can also find this document by logging on to [My Oracle Support](#) and searching the knowledge base for "`vmpinfo3`".

Chapter 2. Using the Oracle VM Utilities

Table of Contents

2.1. Using Oracle VM Manager Control	5
2.1.1. Command Line Options and Parameters of ovm_managercontrol	5
2.1.2. Examples of ovm_managercontrol Usage	6
2.2. Using Oracle VM Server Control	7
2.2.1. Command Line Options and Parameters of ovm_servercontrol	7
2.2.2. Examples of ovm_servercontrol Usage	8
2.3. Using Oracle VM Virtual Machine Control	10
2.3.1. Command Line Options and Parameters of ovm_vmcontrol	10
2.3.2. Examples of ovm_vmcontrol Usage	11
2.4. Using Oracle VM Retrieve Disk	12
2.4.1. Command Line Options and Parameters of ovm_vmdisk	12
2.4.2. Examples of ovm_vmdisk Usage	12
2.5. Using Oracle VM Virtual Machine Messaging	13
2.5.1. Command Line Options and Parameters of ovm_vmmmessage	13
2.5.2. Examples of ovm_vmmmessage Usage	14
2.6. Using Oracle VM Pool Control	14
2.6.1. Command Line Options and Parameters of ovm_poolcontrol	15
2.6.2. Examples of ovm_poolcontrol Usage	15
2.7. Using Oracle VM Repository Control	16
2.7.1. Command Line Options and Parameters of ovm_repocontrol	16
2.7.2. Examples of ovm_repocontrol Usage	17

This chapter gives you more detail on using the Oracle VM Utilities, including command line examples.

The Oracle VM Utilities make a connection to the Oracle VM Manager host server, using the server host name and an administrative user name and password. Oracle VM Manager listens on port 54321 (TCP, local access only) or 54322 (Secure TCP). All utilities share a common set of parameters to connect to the Oracle VM Manager instance. The following parameters are required and used by each :

- -u <user name of an Oracle VM Manager admin user>
- -p <password corresponding with the admin user name>
- -h <host name of the server running Oracle VM Manager>

2.1. Using Oracle VM Manager Control

The `ovm_managercontrol` utility operates at the level of the Oracle VM Manager. It allows the administrator to execute basic commands applicable to the configuration of the entire environment managed by the Oracle VM Manager.

2.1.1. Command Line Options and Parameters of ovm_managercontrol

The `ovm_managercontrol` utility has the following command line options:

- -u user name of an Oracle VM Manager admin user (required)
- -p password corresponding with the admin user name (required)
- -h host name of the server running Oracle VM Manager (required)

- **-X** use SSL to connect to Oracle VM Manager at `tcps://host:54322`
- **-k** name of the keystore file used for SSL
- **-P** password to open the keystore file
- **-G** enable GPG check
- **-K** URL to the GPG key
- **-B** base URL for the YUM server repository
- **-n** CPU compatibility group name
- **-s** name of the Oracle VM Server to add to or remove from a CPU compatibility group
- **-T** timeout in seconds for virtual machine console session (use together with `-c getsessiontimeout/setsessiontimeout`)
- **-c** console command (required)

[**commands:** `yuminfo`, `setupyum`, `addkeystore`, `keystoreinfo`, `createcpugroup`, `removecpugroup`, `addservertocpugroup`, `removeserverfromcpugroup`, `listcpugroups`, `getsessiontimeout`, `setsessiontimeout`]



Note

See the examples in the next section for typical combinations of parameters and options.

2.1.2. Examples of ovm_managercontrol Usage

Below are examples of `ovm_managercontrol` commands and their respective output.

- Set up a keystore file for SSL connectivity and check the keystore settings:

```
# ./ovm_managercontrol -u admin -p password -h localhost -c addkeystore -k mykeystore.ks -P password
Oracle VM Manager Control utility 0.5.2.
Connected.
Command : addkeystore
Adding keystore.
Restart of Oracle VM Manager is required for this setting to take effect. (/etc/init.d/ovmm stop ; /etc/init
Exit...
```

```
# ./ovm_managercontrol -u admin -p password -h localhost -c keystoreinfo
Oracle VM Manager Control utility 0.5.2.
Connected.
Command : keystoreinfo
Key Store information :
    Key Path      : '/u01/app/oracle/ovm-manager-3/keystore/mykeystore.ks'
Exit...
```

- Configure the YUM repository for Oracle VM Server updates:

```
# ./ovm_managercontrol -u admin -p password -h localhost -c setupyum \\  
-B http://yumserver.domain.com/repo -G -K http://secure.domain.com/key.gpg
```

- Configure a CPU compatibility group and add an Oracle VM Server:

```
# ./ovm_managercontrol -u admin -p password -h localhost -c createcpugroup -n MyCPUgroup_Opteron
Oracle VM Manager Control utility 0.5.2.
Connected.
```

```
Command : createcpugroup
Creating CPU Compatibility Group 'MyCPUgroup_Opteron'.
Exit...

# ./ovm_managercontrol -u admin -p password -h localhost -c addservertocpugroup -s MyServer1 -n MyCPUgro
Oracle VM Manager Control utility 0.5.2.
Connected.
Command : addservertocpugroup
Adding Server to CPU Compatibility group.
Exit...
```

- Display the CPU compatibility group configuration, that is: the compatibility groups and their member servers:

```
# ./ovm_managercontrol -u admin -p password -h localhost -c listcpugroups
Oracle VM Manager Control utility 0.5.2.
Connected.
Command : listcpugroups
Group : MyCPUgroup_Opteron
Server : MyServer1
Server : MyServer2
Server : MyServer3
Server : MyServer6
Server : MyServer7
Server : MyServer8

Group : MyCPUgroup_Xeon
Server : MyServer4
Server : MyServer5
Exit...
```

2.2. Using Oracle VM Server Control

The `ovm_servercontrol` utility operates on a physical Oracle VM Server. It allows the administrator to execute basic server commands on a given Oracle VM Server, query its status and retrieve essential information.

2.2.1. Command Line Options and Parameters of `ovm_servercontrol`

The `ovm_servercontrol` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
- **-p** password corresponding with the admin user name (required)
- **-h** host name of the server running Oracle VM Manager (required)
- **-X** use SSL to connect to Oracle VM Manager at `tcps://host:54322`
- **-s** Oracle VM Server name
- **-I** Oracle VM Server ID
- **-P** agent password for the Oracle VM Server to be discovered (use together with `-c discover`)
- **-r** storage repository name (for NFS export)
- **-i** ID of the NFS export
- **-o** options for the NFS export
- **-C** client host name for the NFS export

- **-A** acknowledge events (use together with -c events)

use **-A all** to acknowledge events for the Oracle VM Server and the virtual machines it hosts

- **-c** console command (required)

[**commands:** *start, stop, restart, kill, discover, mainton, maintoff, status, info, events, lock, upgrade, refresh, list, listnfsexports, createnfsexport, deletenfsexport*]



Note

The **mainton** and **maintoff** commands activate or deactivate the Oracle VM Server maintenance mode.

The **discover** command performs a discovery operation on an Oracle VM Server that was not yet previously discovered by Oracle VM Manager.

The **upgrade** command initiates a yum upgrade on the given Oracle VM Server.

The **list** command does not need an Oracle VM Server name argument. It lists every Oracle VM Server registered with the system.

2.2.2. Examples of ovm_servercontrol Usage

Below are examples of **ovm_servercontrol** commands and their respective output.

- List all registered Oracle VM Servers:

```
# ./ovm_servercontrol -u admin -p password -h localhost -c list
Oracle VM Server Control utility 0.5.2.
Connected.
Command : list
Server : 'OVS_01'
  ID      : '00:e0:81:4d:40:16:00:e0:81:4d:40:17:ff:ff:ff:ff'
  pool    : 'MyServerPool1'
  status  : 'Running'

Server : 'OVS_02'
  ID      : '00:e0:81:4d:5e:82:00:e0:81:4d:5e:83:ff:ff:ff:ff'
  pool    : 'unassigned'
  status  : 'Running'

Server : 'OVS_03'
  ID      : '00:e0:81:4d:5e:16:00:e0:81:4d:5e:17:ff:ff:ff:ff'
  pool    : 'MyServerPool1'
  status  : 'Running'

Server : 'OVS_04'
  ID      : '00:e0:81:4d:40:f5:00:e0:81:4d:40:be:00:e0:81:4d'
  pool    : 'unassigned'
  status  : 'Running'

Exit...
```

- Check the status of an Oracle VM Server:

```
# ./ovm_servercontrol -u admin -p password -h localhost -s OVS_03 -c status
Oracle VM Server Control utility 0.5.2.
Connected.
Command : status
Server : 'OVS_03' status : 'Running' serverpool : 'MyServerPool1'.
```

```
Exit...
```

- Display detailed information of an Oracle VM Server:

```
# ./ovm_servercontrol -u admin -p password -h localhost -s OVS_01 -c info
Oracle VM Server Control utility 0.5.2.
Connected.
Command : info
Server : OVS_01
      Status      : Running
      Management IP: 10.172.76.90
      Hostname     : OVS_01
      ID          : 00:e0:81:4d:40:16:00:e0:81:4d:40:17:ff:ff:ff:ff
      Pool        : MyServerPool1
      CPU threads  : 4
      CPU Usage    : 1.877896
      Total Memory : 32767
      Usable Memory: 31261
      Maintenance : false
      Master Server: true
      Up to date   : true
      Virtual Machines on this server :
        Name      : 'MyVM01'
        Status    : 'Stopped'
        Name      : 'MyVM02'
        Status    : 'Running'
Exit...
```

- Restart an Oracle VM Server:

```
# ./ovm_servercontrol -u admin -p password -h localhost -s OVS_02 -c restart
Oracle VM Server Control utility 0.5.2.
Connected.
Command : restart
restartserver 'OVS_02' completed.
Exit...
```

- Expose an OCFS2 repository using NFS:



Note

This example shows the creation of an NFS export and then lists the existing exports. The storage repository is exposed to two different clients, but then one export is deleted via its ID. The mount command at the end shows that the remaining export is still available.

```
# ./ovm_servercontrol -u admin -p password -h localhost -c createnfsexport -s OVS_04 \
-C NFS_02 -o rw,no_root_squash -r MyIscsiRepo
Oracle VM Server Control utility 0.5.2.
Connected.
Command : createnfsexport
Creating nfs export for repository 'MyIscsiRepo' on server.
Created repository export
Exit...

# ./ovm_servercontrol -u admin -p Manager1 -h localhost -c listnfsexports -s OVS_04
Oracle VM Server Control utility 0.5.2.
Connected.
Command : listnfsexports
NFS Exports :
      id          : 0004fb0000230000978a28e2dc85e06b
      client      : NFS_01
      options     :
      repository  : MyIscsiRepo
      path        : /OVS/Repositories/0004fb000003000031fac25b24740cca/
```

```

---
    id       : 0004fb00002300004b161fd1a0d106b7
    client    : NFS_02
    options   : rw,no_root_squash
    repository : MyIscsiRepo
    path      : /OVS/Repositories/0004fb000003000031fac25b24740cca/
---
Exit...

# ./ovm_servercontrol -u admin -p password -h localhost -c deletenfsexport -s OVS_04 \
-i 0004fb0000230000978a28e2dc85e06b
Oracle VM Server Control utility 0.5.2.
Connected.
Command : deletenfsexport
Deleting nfs export with ID '0004fb0000230000978a28e2dc85e06b'
Deleted repository export
Exit...

[root@NFS_02]# mount OVS_04:/OVS/Repositories/0004fb000003000031fac25b24740cca /mnt/repoexport
[root@NFS_02]# ls /mnt/repoexport
Assemblies  ISOs  lost+found  Templates  VirtualDisks  VirtualMachines

```

2.3. Using Oracle VM Virtual Machine Control

The `ovm_vmcontrol` utility operates on a virtual machine (VM). It passes basic virtual machine control commands from the command line to Oracle VM Manager. It allows you to start, stop, suspend, resume, kill, restart a VM. It is also possible to simply query the status of the virtual machine (Running, Stopped), and retrieve essential VM information such as memory, server pool, number of virtual CPUs and so on.

2.3.1. Command Line Options and Parameters of `ovm_vmcontrol`

The `ovm_vmcontrol` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
- **-p** password corresponding with the admin user name (required)
- **-h** host name of the server running Oracle VM Manager (required)
- **-X** use SSL to connect to Oracle VM Manager
- **-v** virtual machine name
- **-U** virtual machine UUID
- **-t** name of target Oracle VM Server for migration of a virtual machine (use together with `-c migrate`)
- **-s** comma separated list of physical thread numbers to bind virtual CPUs to (use together with `-c vcpuset`)
- **-T** tag list, a comma separated list of strings to be associated with a virtual machine as metadata (use together with `-c settags`)
- **-F** force the virtual disks associated with a virtual machine to be deleted (use together with `-c delete`)
- **-A** acknowledge virtual machine events (use together with `-c events`)
- **-c** console command (required)

[commands: `start`, `stop`, `suspend`, `resume`, `status`, `restart`, `kill`, `delete`, `info`, `fixcfg`, `events`, `lock`, `vcpuset`, `vcpuget`, `migrate`, `settags`, `gettags`, `list`]

**Note**

As you can see, the `ovm_vmcontrol` console command list contains a `lock` command. This special command allows you to lock a virtual machine for a certain period of time. When a virtual machine is locked by the utility, it is not possible for other tools or users (through the Oracle VM Manager user interface) to perform operations on that specific VM. This is particularly useful if you are an administrator and need to make a VM backup while the VM is stopped: you want to prevent anyone starting the VM before the backup finishes.

The `list` command does not need a VM name or UUID argument. It lists every VM and its status.

The `fixcfg` command verifies the `vm.cfg` file of the virtual machine and attempts to correct invalid entries. For example, an entry for `vncpasswd` will be cleared with this command, because the virtual machine console does not support this setting.

The `vcpuset` command hard-binds or pins virtual CPUs to threads. For example, `-c vcpuset -s (0,1,2)` physically binds `vcpu0` to `thread0`, `vcpu1` to `thread1`, `vcpu2` to `thread2`. Use the `vcpuset` command to retrieve information about pinned vCPUs for the selected virtual machine.

The `settags` and `gettags` commands are used to manage tag metadata for a given virtual machine.

2.3.2. Examples of ovm_vmcontrol Usage

Below are examples of `ovm_vmcontrol` commands and their respective output.

- Display detailed information of a particular virtual machine:

```
# ./ovm_vmcontrol -u admin -p password -h localhost -v MyVM02 -c info
Oracle VM VM Control utility 0.5.2.
Connected.
Command : info
Virtual Machine : MyVM02
  Status : Running
  Memory : 1024
  Uptime : 83 Minutes
  Server : OVS_01
  Pool   : MyServerPool1
  HA Mode: false
  VCPU  : 2
  Type  : Xen HVM
  OS    : Oracle Linux 6
vminfo 'MyVM02' completed.
```

- Check the status of a particular virtual machine:

```
# ./ovm_vmcontrol -u admin -p password -h localhost -v MyVM02 -c status
Oracle VM VM Control utility 0.5.2.
Connected.
Command : status
Virtual Machine : 'MyVM02' status : 'Running' on server 'OVS_01' serverpool : 'MyServerPool1'.
```

- Stop a particular virtual machine:

```
# ./ovm_vmcontrol -u admin -p password -h localhost -v MyVM01 -c stop
Oracle VM VM Control utility 0.5.2.
Connected.
Command : stop
```

```
stopvm 'MyVM01' completed.
```

- Start a particular virtual machine:

```
# ./ovm_vmcontrol -u admin -p password -h localhost -v MyVM01 -c start
Oracle VM VM Control utility 0.5.2.
Connected.
Command : start
startvm 'MyVM01' completed.
```

- Lock a particular virtual machine:

```
# ./ovm_vmcontrol -u admin -p password -h localhost -v MyVM01 -c lock
Oracle VM VM Control utility 0.5.2.
Connected.
Command : lock
Locking VM 'MyVM01'
Press Ctrl-C to unlock.
```

The lock command will sleep until you press **Ctrl+C** to release the lock. If the `ovm_control` utility is killed, the VM would be unlocked as well.

2.4. Using Oracle VM Retrieve Disk

The `ovm_vmdisks` utility is designed to help the administrator make backups of virtual machines, particularly when the VM's virtual disks are files in a storage repository on a remote NFS storage server.

2.4.1. Command Line Options and Parameters of `ovm_vmdisk`

The `ovm_vmdisks` utility takes a virtual machine name or UUID and lists out every virtual disk file for that VM, as well as the virtual machine configuration (`vm.cfg`) file. This allows you to take the file listings from the output of the utility and back them up to a server or tape.

The `ovm_vmdisks` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
- **-p** password corresponding with the admin user name (required)
- **-h** host name of the server running Oracle VM Manager (required)
- **-X** use SSL to connect to Oracle VM Manager
- **-v** virtual machine name
- **-U** virtual machine UUID

If the virtual disks are actual physical devices directly attached to the VM, the utility lists the device mapper entry on the Oracle VM Server to which the VM is assigned. In case the virtual disks are files on an NFS server, the utility lists the NFS server name, mount point and file name/location.

2.4.2. Examples of `ovm_vmdisk` Usage

Below is an example of `ovm_vmdisk` usage. It shows a virtual machine with three disks, each attached in a different way:

```
# ./ovm_vmdisks -u admin -p password -h localhost -v MyVM01
Oracle VM Retrieve Disk utility 0.5.2.
Connected.
Virtual Machine : 'MyVM01' status : 'Running'.
```

```

Assigned Server : OVS_01
Virtual Disk : 'MyVM01_bootdisk' size : 20GB
  file=/OVS/Repositories/0004fb0000030000b0272c74e714ab12/VirtualDisks/ \
  0004fb00001200006e548a28cb886f42.img
  mountpoint=/dev/mapper/3300000006160a212/VirtualDisks/ \
  0004fb00001200006e548a28cb886f42.img
Virtual Disk : 'MyVM01_datadisk' size : 250GB
  file=/OVS/Repositories/0004fb00000300005b06362f2d866a98/VirtualDisks/ \
  0004fb0000120000f217eb8c0fa70eef.img
  mountpoint=nfs01:/mnt/vol2/repo03/VirtualDisks/ \
  0004fb0000120000f217eb8c0fa70eef.img
Physical Disk : 'pd003'
  device=/dev/mapper/3300000009b7b2cc4
Config File :
  file=/OVS/Repositories/0004fb00000300005b06362f2d866a98/ \
  VirtualMachines/0004fb00000600008757e38248a544e6/vm.cfg
  mountpoint=nfs01:/mnt/vol2/repo03/VirtualMachines/ \
  0004fb00000600008757e38248a544e6/vm.cfg

```

As you can see in the console output, VM *'MyVM01'* is currently running. Note that backing up a running VM would be inconsistent, therefore it is highly recommended that you shut down the VM prior to making any backups. The console output shows that the virtual machine contains three disks, two virtual and one physical, and also provides the location of the virtual machine configuration file:

- **MyVM01_bootdisk:** a virtual disk on an ocfs2 storage repository. The ocfs2 repository was created on the device `/dev/mapper/3300000006160a212`
- **MyVM01_datadisk:** a virtual disk on an NFS storage server named `nfs01`. The full path to the virtual disk file on the NFS server is `/mnt/vol2/repo03/VirtualDisks/0004fb0000120000f217eb8c0fa70eef.img`
- **pd003:** a raw disk device at `/dev/mapper/3300000009b7b2cc4`
- **vm.cfg:** the virtual machine configuration file on NFS server `nfs01`. The full path to the configuration file is `nfs01:/mnt/vol2/repo03/VirtualMachines/0004fb00000600008757e38248a544e6/vm.cfg`

2.5. Using Oracle VM Virtual Machine Messaging

The `ovm_vmmessage` utility lets you send a message to a running virtual machine, or to query the value of a message sent from within a virtual machine to Oracle VM Manager via the Oracle VM API messaging interface. These values are basic key/value pairs such as `foo=bar`.

New Oracle VM templates released by Oracle will contain a configuration utility called `ovmd`. This utility is used to perform first-boot installation configuration either locally from the virtual machine console or remotely through the messaging interface provided by this utility. The `ovmd` utility also allows the owner of the virtual machine to send messages back to Oracle VM Manager. For more information about `ovmd`, and the Oracle VM Guest Additions in general, see [Chapter 3, Using the Oracle VM Guest Additions](#).

2.5.1. Command Line Options and Parameters of `ovm_vmmessage`

The `ovm_vmmessage` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
- **-p** password corresponding with the admin user name (required)
- **-h** host name of the server running Oracle VM Manager (required)
- **-X** use SSL to connect to Oracle VM Manager

- **-v** virtual machine name
- **-U** virtual machine UUID
- **-k** key to send
- **-V** value to send along with the key (required when using **-k**)
- **-q** key to query

Note that `ovm_vmmessage` only works on a running virtual machine. To send messages you combine **-k** `<key>` and **-V** `<value>`. To retrieve messages, use **-q** `<key>`.



Caution

The virtual machine name is the name you assign during the creation of the VM. However, the same name could be assigned to several different VMs. If that is the case, you must use the **-U** option and provide the unique identifier (UUID) of the VM.

2.5.2. Examples of ovm_vmmessage Usage

Below are examples of `ovm_vmmessage` commands and their respective output.

- Send a message key/value pair to a virtual machine:

```
# ./ovm_vmmessage -u admin -p password -h localhost -v MyVM02 -k foo -V bar
Oracle VM VM Message utility 0.5.2.
Connected.
VM : 'MyVM02' has status : Running.
Sending message.
Message sent successfully.
```

- Send a message key/value pair to a virtual machine, identifying the VM by its UUID:

```
# ./ovm_vmmessage -u admin -p password -h localhost \
-U 0004fb00000600001c925eac2ad5d328 -k foo -V bar
Oracle VM VM Message utility 0.5.2.
Connected.
VM : 'MyVM02' has status : Running.
Sending message.
Message sent successfully.
```

- Retrieve a message from a virtual machine, identifying the VM by its UUID:

```
# ./ovm_vmmessage -u admin -p password -h localhost \
-U 0004fb00000600001c925eac2ad5d328 -q foo
Oracle VM VM Message utility 0.5.2.
Connected.
VM : 'MyVM02' has status : Running.
Querying for key 'foo'.
Query successful.
Query for Key : 'foo' returned value 'bar'.
Key set 27 minutes ago.
```

2.6. Using Oracle VM Pool Control

The `ovm_poolcontrol` utility allows the administrator to quickly retrieve information about the status of a given server pool under the control of this Oracle VM Manager, and view and acknowledge events registered for both the server pool and its Oracle VM Server members.

2.6.1. Command Line Options and Parameters of ovm_poolcontrol

The `ovm_poolcontrol` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
 - **-p** password corresponding with the admin user name (required)
 - **-h** host name of the server running Oracle VM Manager (required)
 - **-X** use SSL to connect to Oracle VM Manager
 - **-s** server pool name
 - **-I** server pool UUID (which is a series of 16 hexadecimal numbers separated by colons)
 - **-S** Oracle VM Server name (use together with `-c addserver/removeserver`)
 - **-A** acknowledge events (use together with `-c events`)
- [all, pool]:** acknowledge events for both server pool and servers, or for the server pool only
- **-c** console command (required)

[commands: `list`, `addserver`, `removeserver`, `events`, `status`, `info`**]**



Note

With the `addserver` and `removeserver` commands, only a very basic add/remove of the server is performed; no assigning or unassigning of networks and repositories occurs.

2.6.2. Examples of ovm_poolcontrol Usage

Below are examples of `ovm_poolcontrol` commands and their respective output.

- List all server pools and their status:

```
# ./ovm_poolcontrol -u admin -p password -h localhost -c list
Oracle VM Pool Control utility 0.5.2.
Connected.
Command : list
Repository : 'MyServerPool1'
  ID       : '0004fb0000020000d2ee5d9394b371c5'
  Status   : 'Online'

Repository : 'MyServerPool2'
  ID       : '0004fb0000020000183c51866b2fa4a1'
  Status   : 'Online'

Exit...
```

- Display detailed information of a server pool:

```
# ./ovm_poolcontrol -u admin -p password -h localhost -s MyServerPool1 -c info
Oracle VM Pool Control utility 0.5.2.
Connected.
Command : info
Server Pool : MyServerPool1
  Status      : Online
  ID          : 0004fb0000020000d2ee5d9394b371c5
  Virtual IP  : 10.172.77.172
```

```

Clustered      : true
Master Server  : MyServer1
KeyMap         : en-us
Load Balancer  : Off
Pool Memory    : 57342
Available Memory: 54631
Total Threads  : 8
CPU Utilization : 23%
Servers in this server pool :
  Name      : 'MyServer1'
  Status    : 'Running'
  Name      : 'MyServer2'
  Status    : 'Running'
  Name      : 'MyServer3'
  Status    : 'Running'
Exit...

```

- Acknowledge server pool events:

```

# ./ovm_poolcontrol -u admin -p password -h localhost -s MyServerPool1 -c events -A pool
Oracle VM Pool Control utility 0.5.2.
Connected.
Command : events
Server Pool Events for pool 'MyServerPool'
Exit...

```

2.7. Using Oracle VM Repository Control

The `ovm_repocontrol` utility allows the administrator to quickly retrieve information about status and contents of a given storage repository, and send a refresh command to update the storage repository contents listed.

2.7.1. Command Line Options and Parameters of `ovm_repocontrol`

The `ovm_repocontrol` utility has the following command line options:

- **-u** user name of an Oracle VM Manager admin user (required)
- **-p** password corresponding with the admin user name (required)
- **-h** host name of the server running Oracle VM Manager (required)
- **-X** use SSL to connect to Oracle VM Manager
- **-r** storage repository name
- **-I** storage repository UUID
- **-s** name of an Oracle VM Server with access to the storage under the repository (use together with **-c** create)
- **-i** UUID of a LUN where a storage repository should be created (use together with **-c** create)
- **-c** console command (required)

[**commands:** `list`, `status`, `info`, `refresh`, `fixrepo`, `create`]



Note

The `refresh` command initiates a storage repository refresh operation through Oracle VM Manager.

The `list` command does not need a storage repository name or UUID argument. It lists every storage repository registered with the system.

The `fixrepo` command removes partial configuration information in case the creation of a storage repository on a physical disk has failed. This allows you to redo the creation operation for the storage repository.

2.7.2. Examples of ovm_repocontrol Usage

Below are examples of `ovm_servercontrol` commands and their respective output.

- List all registered storage repositories:

```
# ./ovm_repocontrol -u admin -p password -h localhost -c list
Oracle VM Repository Control utility 0.5.2.
Connected.
Command : list
Repository : 'MyIscsiRepo1'
  ID       : '0004fb0000030000b0272c74e714ab12'
  Status   : 'Online'

Repository : 'MyIscsiRepo2'
  ID       : '0004fb0000030000a1d2474a7325ab25'
  Status   : 'Online'

Repository : 'MyNfsRepo'
  ID       : '0004fb0000030000a3ff6bbfe68511a6'
  Status   : 'Online'

Exit...
```

- Create a new storage repository on a LUN:

```
# ./ovm_repocontrol -u admin -p password -h localhost -c create -r MyIscsiRepo3 -s MyServer4 \
-i 3600144f057ef8a0000004fb3194b0001
Oracle VM Repository Control utility 0.5.2.
Connected.
Command : create
Found LUN with id '3600144f057ef8a0000004fb3194b0001'
Creating filesystem...
Creating repository... Please wait...
Create repository completed.
Exit...
```

- Check the status of a storage repository:

```
# ./ovm_repocontrol -u admin -p password -h localhost -r MyIscsiRepo2 -c status
Oracle VM Repository Control utility 0.5.2.
Connected.
Command : status
Repository : 'MyIscsiRepo2'
  Status   : 'Online'

Exit...
```

- Display detailed information of a storage repository:

```
# ./ovm_repocontrol -u admin -p password -h localhost -r MyIscsiRepo1 -c info
Oracle VM Repository Control utility 0.5.2.
Connected.
Command : info
Repository : 'MyIscsiRepo1'
  Total Space   : 7500000 MB
  Available Space : 4146146 MB
```

```
Assigned Servers :
  Server : MyServer6
  Server : MyServer7
Assemblies :
Virtual CDroms :
  CDrom : ubuntu-11.10-desktop-i386.iso
  uuid : 0004fb0000150000d8de2efc475b4ea5.iso
  VM list : (MyUbuntu1110)
  CDrom : EnterpriseLinux-R6U1-Server-x86_64.iso
  uuid : 0004fb0000150000da320985ba0c2f9b.iso
  VM list : (MyOL6)
Virtual Disks :
  Virtual Disk : MyOL6_bootdisk
  uuid : 0004fb000012000086bf184d2996f0cf.img
  VM list : (MyOL6)
  Virtual Disk : MyOL6_datadisk
  uuid : 0004fb00001200007d18ce0a22e16658.img
  VM list : (MyOL6)
  Virtual Disk : MySolaris11_bootdisk
  uuid : 0004fb00001200008a27187d18c6fe0a.img
  VM list : (MySolaris11)
  Virtual Disk : MyUbuntu1110_bootdisk
  uuid : 0004fb000012000096f0ce0a27d18c6f.img
  VM list : (MyUbuntu1110)
Virtual Machine Configs :
  Config : 0004fb0000220000b1fc2e486c1c1b7a
  VM : (MySolaris11)
  Config : 0004fb0000220000b68857e77465eafc
  VM : (MyOL6)
  Config : 0004fb0000220000eb037b82dee6b10b
  VM : (MyUbuntu1110)
Exit...
```

- Refresh the contents of a storage repository:

```
# ./ovm_repocontrol -u admin -p password -h localhost -r MyIscsiRepol -c refresh
Oracle VM Repository Control utility 0.5.2.
Connected.
Command : refresh
refreshing 3300000006160a212
Lun Refresh completed...
Exit...
```

Chapter 3. Using the Oracle VM Guest Additions

Table of Contents

3.1. Installing Oracle VM Guest Additions	19
3.2. Features of the Oracle VM Guest Additions	20
3.2.1. Using the Messaging Channel	20
3.2.2. Displaying the Guest IP Address	21

This chapter gives you more detail on the installation, configuration and features of the Oracle VM |Guest Additions.

3.1. Installing Oracle VM Guest Additions

For the Oracle VM templates without Guest Additions installed, the following software packages can be downloaded from [Oracle Unbreakable Linux Network \(ULN\)](#) by choosing the proper channel to download and install into the existing Oracle Linux 5 or 6 guest with Oracle Unbreakable Enterprise Kernel (UEK).

```
kmod-ovmapi-uek
libovmapi
libovmapi-devel
ovmd
python-simplejson
xenstoreprovider
ovm-template-config
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
```

These packages can be downloaded from the following [ULN](#) channels:

- Oracle Linux 6 Add ons (x86_64)
- Oracle Linux 6 Add ons (i386)
- Enterprise Linux 5 Add ons (x86_64)
- Enterprise Linux 5 Add ons (i386)

In addition, the packages are available from Oracle's [Public YUM repository](#).

To install these packages, make sure that your virtual machine has public internet access and is connected to the appropriate ULN channel or YUM repository. Then use the following command syntax in your Oracle Linux guest, separating the package names by spaces:

```
# yum install libovmapi xenstoreprovider ovmd python-simplejson xenstoreprovider
```



Caution

When manually installing the downloaded packages, make sure that the `kmod-ovmapi-uek` version matches the UEK version of the virtual machine:

- UEK 2.36.32-100 requires `kmod-ovmapi-uek-1.0.0-27.100`
- UEK 2.36.32-200 requires `kmod-ovmapi-uek-1.0.0-27.200`
- UEK 2.36.32-300 requires `kmod-ovmapi-uek-1.0.0-27.300`

3.2. Features of the Oracle VM Guest Additions

Together with the `ovm_vmmmessage` utility on the Oracle VM Manager, the Oracle VM Guest Additions form a bi-directional messaging channel between Oracle VM Manager and the guest. More specifically, as part of the Oracle VM Guest Additions, a command line tool called `ovmd` is installed. It allows first-boot installation configuration, and is capable of sending and receiving messages consisting of key-value pairs just like the `ovm_vmmmessage` utility. As you can see in the help, `ovmd` has the following command line options:

```
# ovmd --help
usage: ovmd [options]

options:
-p, --set-param=PARAM    Set param in format: key=value.
-g, --get-param=KEY      Get param value.
-r, --delete-param=KEY   Delete param by key name.
-x, --delete-params      Delete all params.
-l, --list-params        List all params.
-e, --event=EVENT        Inject event.
-s, --script=SCRIPT      Run script.
-d, --debug=LEVEL        Set debug level: 0(DEBUG_OFF), 1(DEBUG_STDERR) or 2(DEBUG_SYSLOG). Default: 2.
-f, --pid-file=FILENAME  Set the pathname of the process ID (PID) file.
-t, --time-period=secs   Set the period for daemon mode(default=10 seconds).
-v, --version            Show version number and exit.
-h, --help              Show this help information.
```

3.2.1. Using the Messaging Channel

This sections provides an example of a message exchange between Oracle VM Manager and a running Oracle Linux virtual machine with Oracle VM Guest Additions installed. More information about the messaging utility can be found in [Section 2.5, “Using Oracle VM Virtual Machine Messaging”](#).

Sending a message from the guest to Oracle VM Manager.

Using `ovmd`, you send information to your Oracle VM Manager using the following syntax:

```
# ovmd -p key1=value1
```

The message shows up 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.

The message from the guest can also be retrieved via the Oracle VM Manager command line utility `ovm_vmmmessage`. To do so, you query the key and the value is returned in the response:

```
# ./ovm_vmmmessage -u admin -p password -h localhost -v MyVM02 -q key1
Oracle VM VM Message utility 0.5.2.
Connected.
VM : 'MyVM02' has status : Running.
Querying for key 'key1'.
Query successful.
Query for Key : 'key1' returned value 'value1'.
Key set 7 minutes ago.
```

Sending a message from Oracle VM Manager to a virtual machine.

Using `ovm_vmmessage`, you send information to a virtual machine using the following syntax:

```
# ./ovm_vmmessage -u admin -p password -h localhost -v MyVM02 -k key2 -V value2
Oracle VM VM Message utility 0.5.2.
Connected.
VM : 'MyVM02' has status : Running.
Sending message.
Message sent successfully.
```

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"}
```

3.2.2. Displaying the Guest IP Address

When the Oracle VM Guest Additions are installed, the virtual machine IP address becomes visible in the Oracle VM Manager user interface, as part of the detailed virtual machine information.



