ORACLE UNIVERSITY

Integrated Cloud Applications & Platform Services

Oracle VM Server for x86: Implementation

Student Guide

D82656GC20

Edition 2.0 | May 2017 | D100372

Learn more from Oracle University at **education.oracle.com**

ORACLE

Author

Dave Goff

Technical Contributors and Reviewers

S. Matt Taylor, Jr

Steve Miller

Greg King

Avi Miller

Mirella Tumolo

Editors

Aju Kumar

Chandrika Kennedy

Aishwarya Menon

Graphic Designers

Prakash Dharmalingam

Kavya Bellur

Publishers

Sujatha Nagendra

Giri Venugopal

Syed Ali

Jobi Varghese

Contents

1

Oracle VM Manager UI and Oracle VM CLI

Objectives

After completing this lesson, you should be able to:

- Describe the major components of the Oracle VM environment
- List the interfaces available to manage your Oracle VM environment
- Describe the control elements of the Oracle VM Manager UI
- Connect to and use the Oracle VM CLI and configure its behavior
- Use the Oracle VM CLI from the command line

Oracle VM Components

Oracle VM comprises:

- One or more Oracle VM servers in server pools
  - On each server dedicated to your Oracle VM environment, you install Oracle VM Server for x86. You provision and run your virtual machines on these servers.
- One Oracle VM Manager
  - The Oracle VM Manager is the administrative interface to your Oracle VM environment.
- External storage and a network infrastructure
  - The storage and networking infrastructure must be in place before you start configuring Oracle VM from the Oracle VM Manager.

One or More Oracle VM Servers

Each running Oracle VM server (with Oracle VM Server for x86 installed) has:

- A Xen hypervisor, which is the first component loaded on each Oracle VM server when it boots. The Xen hypervisor is a thin layer that runs directly on the server hardware. It manages the allocation of memory and CPU, and processor interrupts.
- A privileged domain called dom0. Dom0 is a special virtual machine, which loads after the hypervisor during the boot process on the Oracle VM server. Dom0 has all the necessary drivers to access the I/O subsystem on behalf of other virtual machines. Other domains (or virtual machines) are created, started, and stopped from dom0.

Oracle VM Manager

The Oracle VM Manager controls the Oracle VM environment. It offers the following choices for administration:

- A graphical user interface (GUI), which you can access by using a browser. You can carry out most administrative tasks by using the UI.

A command-line interface (CLI), which you access by using the `ssh` command from the Oracle VM Manager or from any remote location with network access to the Oracle VM Manager.

- The Oracle VM Manager Web Services API, a web service that offers RESTful interactions

Using the management interfaces provided by Oracle VM, you can configure your storage and networking infrastructure, create virtual machines, import templates and assemblies, clone virtual machines or templates to create new virtual machines, and control the life cycle of your virtual machines (start, suspend, stop, clone, migrate, delete).

The Oracle VM Manager stores information about objects and their status in a database. When you install the Oracle VM Manager software, you install MySQL Enterprise Edition, which is used as the Oracle VM Manager database.

External Storage

You provide external storage to your Oracle VM environment to create storage repositories, and optionally, assign external storage directly to your virtual machines to be used as physical disks.

Network Infrastructure

You use the Oracle VM Manager to configure your networks into your Oracle VM environment. The initial networking infrastructure, such as Ethernet switches, cabling, VLAN support, and network interfaces in the Oracle VM servers must be in place before you attempt to configure networking from the Oracle VM Manager. However, you can add to or change your network topology over time and use the Oracle VM Manager to incorporate these changes into your environment.

Oracle VM Server Roles

After you discover an Oracle VM server from the Oracle VM Manager, the server is assigned two roles by default:

- **Utility Server:** An Oracle VM server with the Utility Server role is a server that Oracle VM Manager dispatches to perform tasks such as importing templates or cloning.
- **VM Server:** An Oracle VM server with the VM Server role can be selected to run virtual machines.

The VM Server role is required for an Oracle VM server if you want to run virtual machines on that server.

An Oracle VM server with the Utility Server role is favored to perform operations other than hosting virtual machines, such as importing templates or creating repositories.

You can use Oracle VM server roles to control the workload for your Oracle VM servers.

In a small environment with only a few servers, you can retain the default roles and have each Oracle VM server capable of carrying out both functions.

In a large environment, you might consider configuring one or two Oracle VM servers as utility servers only, for tasks related to importing, copying, or cloning, to avoid impacting the performance of other Oracle VM servers dedicated to running virtual machines. Utility servers are also ideal for providing the network file system (NFS) exports of Oracle Cluster File System 2 (OCFS2) repositories for backup purposes.

When you are calculating CPU and memory resources for your anticipated virtual machine workload, do not include the CPU and memory resources for Oracle VM servers that do not have the VM Server role.

Deploying Server Pools

- To support running virtual machines, an Oracle VM server must be part of a server pool.
- You can control the behavior of your server pools with the following features:
  - Distributed Resource Scheduler (DRS)
  - Dynamic Power Management (DPM)
  - Anti-affinity
  - Server Processor Compatibility groups

A server pool is a grouping of Oracle VM servers into a logical unit. You can create many server pools, and any Oracle VM server can only be part of one server pool at a time.

Server Pool and Live Migration

Live migration is supported to and from Oracle VM servers within the same server pool. Oracle VM does not support live migration of virtual machines across server pools.

Server Processor Compatibility Groups

A server processor compatibility group is a group of Oracle VM servers with compatible processors, among which a running virtual machine can safely live migrate. Processor compatibility groups are created automatically upon discovering an Oracle VM server belonging to a new processor type and model number. You can create your own processor compatibility groups to:

- Support live migration between Oracle VM servers across different hardware generations and models
- Force virtual machines to live migrate within a specific subset of Oracle VM servers in a server pool

Server Pool Policies

You create server pool policies to further control the behavior of your Oracle VM servers and the virtual machines that run on these servers:

- DRS/DPM are server pool policies to manage server pool resources.

  DRS optimizes resource utilization across the entire server pool. DPM seeks to consolidate resources over fewer Oracle VM servers in the server pool.

- Anti-affinity allows you to specify virtual machines that must never run on the same Oracle VM server.

For more information about server pool policies, see the topics titled “What are Server Pool Policies?” and “What are Anti-Affinity Groups?” in the Oracle VM Concepts Guide, Part Number E64081-01 or later.

Deploying Clustered Server Pools

- A clustered server pool supports high availability for virtual machines and protection for repositories.
  - When creating a repository on an Internet Small Computer Systems Interface (iSCSI) or Fibre Channel (FC) logical unit number (LUN), you are required to specify a clustered server pool.
- A clustered server pool is limited to 32 Oracle VM servers.
- When creating a clustered server pool, you provide a server pool file system on an external LUN or NFS share.

Should all your Oracle VM server pools be clustered?

A clustered server pool offers the following advantages:

- It supports high availability for virtual machines running in the server pool.
- You can create repositories in a cluster on iSCSI or FC external storage. You can also use NFS storage, but NFS-type repositories use NFS semantics for concurrent access, not OCFS2-type protection.

  With unclustered server pools, you can create only NFS-type repositories or repositories on the internal disks of the Oracle VM servers in the unclustered server pool.

Note: Server pool policies, DRS/DPM, and anti-affinity are supported for clustered and unclustered server pools.

Cluster Facts

When creating a clustered server pool, you must provide a pool file system on an NFS share or an iSCSI/FC LUN to store the cluster information. The NFS mount or the LUN size must be at least 12 GB but does not need to be larger than 12 GB.

The server pool file system is the location of the global heartbeat for the server pool. Oracle VM servers read and write to the global heartbeat to indicate that they are alive. For this reason, select a server pool file system location with a good and stable bandwidth.

You learn more about clustered server pools and heartbeat operation in the lesson titled “Operations.”

The pool file system is also used by Oracle VM to store high availability information, such as the list of currently running virtual machines that have high availability turned on. This information is used to restart virtual machines if the Oracle VM server on which they were running fails. This information is independent of the Oracle VM Manager, therefore providing restart capability for highly available virtual machines, even if the Oracle VM Manager is not available.

A clustered server pool is limited to 32 Oracle VM servers. An unclustered server pool can have up to 64 Oracle VM servers.

Preparing Storage for the Oracle VM Environment

- You need storage for the following purposes:
  - Server pool file systems: One server pool file system for each clustered server pool
  - Storage repositories, which contains virtual resources, including virtual disks for your virtual machines
  - Physical disks (LUNs) for your virtual machines
- Consider using raw LUNs directly for your virtual machines that require optimal performance.
- All Oracle VM servers in the same server pool must have the same access to external storage.

Oracle VM supports FC, iSCSI, and NFS storage. The FC and iSCSI protocols present storage as block devices (LUNs), and NFS makes available storage as shares, which are file-based storage.

Note: With Oracle VM, physical disks are LUNs, which are allocated storage in an external storage array, and the internal disks in an Oracle VM server.

To bring FC or iSCSI storage elements in your Oracle VM environment, you discover this storage as storage area network (SAN) servers, one SAN server for each storage array. Similarly, NFS storage is discovered as file servers.

Before you configure storage from the Oracle VM Manager, you must decide which Oracle VM server should have access to which storage devices.

Generally, all Oracle VM servers in a server pool have access to the same storage. You can, however, expose certain storage repositories to a subset of the Oracle VM servers in a server pool.

Using Virtual Disks for Virtual Machines

Virtual disks are created from storage allocated to repositories. Virtual disks residing in OCFS2 repositories provide flexibility for your environment:

- You can create virtual disks by using sparse allocation or thin cloning. These types of virtual disk do not take up the entire available space.

- You can create an instant copy of a virtual disk by cloning it.
- You can use hot cloning for a virtual machine.
- You can share a virtual disk.
- You can increase the size of a virtual disk dynamically.

  Note: Increasing the size of a virtual or physical disk is discussed in the lesson titled "Managing Virtual Machines."

Using External Physical Disks for Virtual Machines

If performance is the ultimate goal, using external physical disks for virtual machines might be a good choice. Physical disks are iSCSI or FC LUNs that you make available to your environment when you configure SAN servers from the Oracle VM Manager.

Physical disks presented to a virtual machine can be used in the same way as physical disks made available to a physical host:

- You can use multipathing for physical disks destined for virtual machines (the OS in the guest running in the virtual machine does not need to implement failover, because it is handled at the Oracle VM server layer).
- You can allocate a physical disk to more than one virtual machine.
- You can use the features of the storage array where the physical disk resides to clone or resize the physical disk.
- You can use Oracle VM to copy a physical disk to a virtual disk, and a virtual disk to a physical disk (though these operations result in a slower copy operation).

Networking with Oracle VM

- All Oracle VM servers in a server pool need identical “logical” networking topology.
- When creating networks from the Oracle VM Manager, use the following building blocks:
  - Ethernet ports
  - Bonded ports
  - VLAN ports
- Specify one or more network roles for your networks:
  - Server Management
  - Live Migration
  - Cluster Heartbeat
  - Virtual Machine
  - Storage

You perform all networking configuration by using one of the Oracle VM Manager management interfaces.

All Oracle VM servers in a server pool must share the same “logical” networking topology for the management, live migration, and cluster heartbeat networks. The physical implementation can differ and the Oracle VM servers can have different bonds and even network interface card (NIC) types as part of these networks. You can provide different networks for supporting storage and for virtual machine traffic.

When you create a network by using the Oracle VM Manager, you specify which Oracle VM servers participate in the network, which unique subnet to use, which network channels (or roles) to assign to the new network, and on which port—bond or VLAN—the network traffic is expected to flow for each Oracle VM Server participating in the network.

You must configure bonded ports and VLAN ports before you can add them to a network.

Network Channels or Roles

When you create a network, you specify one or more network channels, also called roles:

- Server Management

  This is the network used by the Oracle VM Manager to communicate with the Oracle VM servers. The Oracle VM Manager must be able to access every Oracle VM server in the Oracle VM environment.

A Server Management network is created automatically when you discover the first Oracle VM server in your environment. If you discover an Oracle VM server on a different subnet, an additional Server Management network is created.

- Live Migration

  You use this type of network role to migrate virtual machines from one Oracle VM server to another within the same server pool. Optionally, you can encrypt the traffic on this network.

- Cluster Heartbeat

  This network role supports the network heartbeat traffic for all Oracle VM servers in a clustered server pool. You can use the same cluster heartbeat network for more than one server pool, and physically isolate the traffic on this network to the Oracle VM servers belonging to the same server pool.

- Storage

  The storage channel does not control which subnet can be used by storage traffic. The storage channel simply allows you to add IP addresses to ports or bonds in your Oracle VM servers to allow these servers to access the storage.

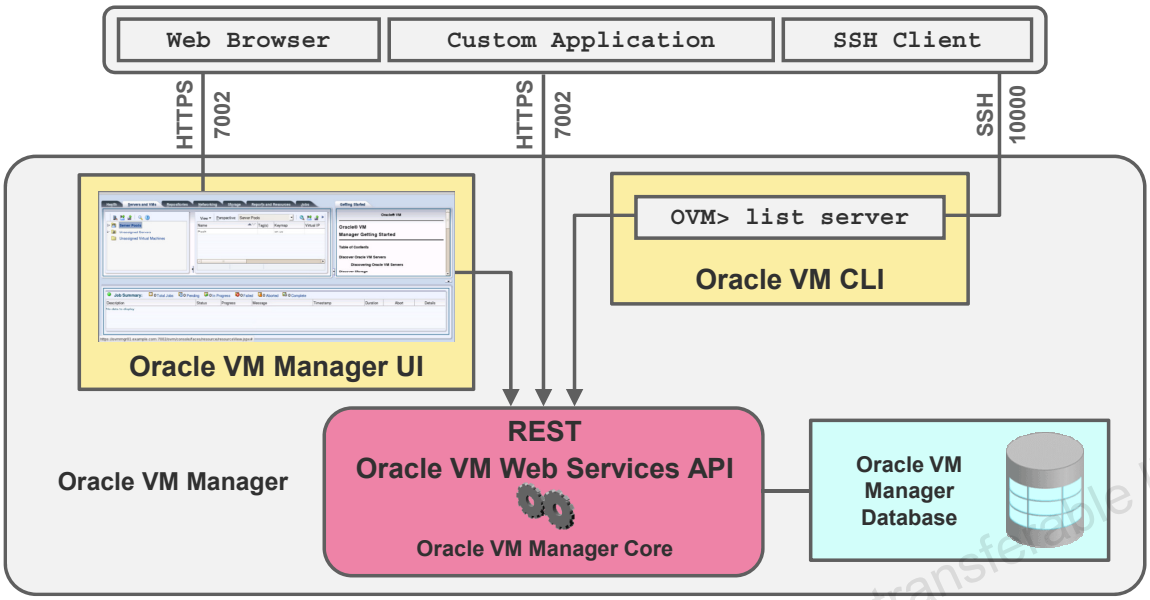
- Virtual Machine

  This network role is used by virtual machines to communicate among themselves (on the same Oracle VM server) and to access the external network. When you create a network with the virtual machine role, a Xen bridge is automatically created on each Oracle VM server participating in the network.

Managing Oracle VM

- In the first topic in this lesson, you reviewed the major components of the Oracle VM environment.
- In the rest of this lesson, you learn about the access points into your Oracle VM environment and how to manage your environment with:
  - Oracle VM Manager UI
  - Oracle VM CLI

Access Points for Management Interfaces



The diagram in the slide shows three ways in which you can interact with the Oracle VM Manager to operate and control your Oracle VM environment. Each access point is discussed in detail in this lesson.

Oracle VM Manager UI

The Oracle VM Manager UI is a browser-based graphical management interface. From the UI, you can perform all administrative functions, except for a few customizations that you perform from the Oracle VM CLI.

Oracle VM CLI

With the Oracle VM CLI, you can automate configuration and operational functions by writing scripts that include embedded Oracle VM CLI commands. Nearly all functionality exposed by the Oracle VM Manager UI is available with the Oracle VM CLI.

Oracle VM Web Services API

The Oracle VM Web Services API (WS-API) offers you a programmatic interface to your Oracle VM environment. From your programs, written in Java, Python, or any other language that supports access to web services, you can configure your environment, operate on virtual machines, and facilitate repetitive tasks like switching between server pool policies or creating new instances of frequently deployed virtual machines from templates.

Additional Ways to Manage Your Oracle VM Environment

- Oracle VM Utilities
  - Oracle VM Utilities are a collection of scripts that allow you to perform basic management tasks.
- Enterprise Manager Cloud Control
  - Use Enterprise Manager Cloud Control to build and manage all aspects of your cloud services.
  - You can manage all of your Oracle VM deployments from a single Enterprise Manager console.
- OpenStack
  - OpenStack is an open source management software for cloud resources.
  - You can deploy OpenStack with Oracle VM to deploy private and public clouds.

There are three additional ways to manage your Oracle VM environment: Oracle VM Utilities, Enterprise Manager Cloud Control, and OpenStack.

Oracle VM Utilities

Oracle VM Utilities are a collection of command-line scripts that allow you to perform a set of basic management tasks on various components of your Oracle VM environment. The Oracle VM Utilities are provided as-is for your convenience.

For more information about the Oracle VM Utilities, read the chapter titled “Oracle VM Utilities” in the *Oracle VM Administrator’s Guide*, Part Number E64083-01 or later.

Enterprise Management Cloud Control

Oracle Enterprise Manager Cloud Control is system management software that delivers:

- Centralized monitoring
- Administration including configuration, management, and support
- Life cycle management functionality for the complete IT infrastructure
- Private cloud self-service provisioning and policy-based resource management

Enterprise Manager Cloud Control plugs in to the Oracle VM Manager and exposes all the functionality of Oracle VM to Enterprise Manager. You can manage your entire Oracle VM environment from within Enterprise Manager Cloud Control.

For more information about the components of Enterprise Manager Cloud Control, read the topic titled “Enterprise Manager Cloud Control 12*c* Architecture” in the *Oracle Enterprise Manager Cloud Control Introduction* document, Part Number E25353-21 or later.

OpenStack

OpenStack is open source virtualization management software, which allows you to connect various technologies and components from different vendors and expose a unified API regardless of the underlying technology.

For help in integrating Oracle VM Server for x86 with OpenStack, read the *Getting Started with Oracle VM, Oracle Linux and OpenStack – Technology Preview* white paper available at http://www.oracle.com/technetwork/server-storage/vm/ovm-linux-openstack-2202503.pdf.

The OpenStack documentation is available at http://docs.openstack.org/.

Training is also available for Oracle OpenStack for Oracle Linux at http://oracle.com/education/linux.

Oracle VM Manager UI

- From the Oracle VM Manager UI, you can perform all administrative tasks, including:
  - Configuration and discovery tasks
  - Virtual machine operations
  - Monitoring status and resource usage
- To access the Oracle VM Manager UI:
  - The `ovmm` service must be running on the Oracle VM Manager host
  - Enter the following URL in your web browser:
    `https://<hostname>:7002/ovm/console`

hostname is the IP address or the host name of the Oracle VM Manager host, or `localhost` for local access.

Administrative tasks include:

- Configuration and discovery tasks, such as discovering Oracle VM servers, creating server pools, configuring storage and networking, and importing templates
- Virtual machine operations, such as cloning templates into virtual machines, starting and stopping virtual machines, and accessing their console
- Monitoring the status and resource usage for Oracle VM servers and virtual machines in your environment

Accessing the Oracle VM Manager UI

The `ovmm` service must be running on the Oracle VM Manager host to log in successfully to the Oracle VM Manager UI.

Use the `service` command to check the status of the `ovmm` service:

```
[root@vmgr ~]# service ovmm status
Oracle VM Manager is running...
[root@vmgr ~]#
```

If the service is not running, start it with the `service` command:

```
[root@vmgr ~]# service ovmm start
```

You access the Oracle VM Manager UI by entering the following URL in your web browser:

`https://<hostname>:7002/ovm/console`

where `hostname` is the IP address or host name of the Oracle VM Manager host, and the port is `7002`.

For example: `https://myhost.example.com:7002/ovm/console`

You are prompted for an administrator user name and a password. The default administrator name is `admin` and the password for `admin` is the password that you specified during the installation of the Oracle VM Manager software.

The traffic between the Oracle VM Manager UI and the client accessing it is encrypted. If the firewall is enabled on your Oracle VM Manager host, you must configure it to allow TCP traffic on `7002`, the port that Oracle VM Manager uses to listen for connections.

The Oracle VM Manager UI uses cookies to store session data. Therefore, to successfully log in and use the Oracle VM Manager UI, your web browser must accept cookies from the Oracle VM Manager host.

You can start more than one session to the Oracle VM Manager UI but unexpected results might occur.

Oracle VM Manager UI: Authentication

- Communication between clients and the Oracle VM Manager UI is secured by using a secure sockets layer (SSL) certificate.
- The certificate is generated automatically when you install the Oracle VM Manager software.
- During the initial access to the Oracle VM Manager UI, the Oracle VM Manager offers a self-signed certificate.
- You can replace the installation-generated SSL certificate with a certificate signed by an external certificate authority (CA).

During the Oracle VM Manager installation, a self-signed SSL certificate is generated and stored in a custom keystore on the Oracle VM Manager host.

The Oracle VM Manager keystore is backed up during the MySQL database backup that is performed automatically every 24 hours.

SSL Terminology

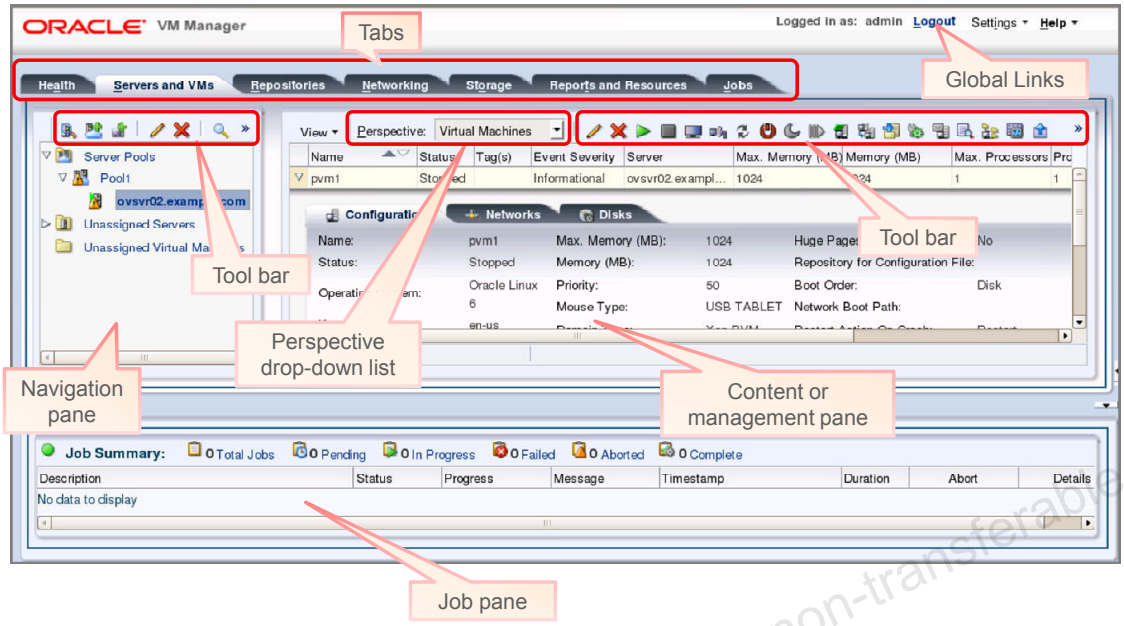
A keystore, located on a host, is a repository of certificates used for SSL encryption.

A certificate contains a public key plus information about the organization issuing the certificate.

The communication between your web browser client and the Oracle VM Manager UI uses a one-way authentication. The client accepts the Oracle VM Manager UI's SSL certificate and the client establishes its identity by sending a username and a password. Because the Oracle VM Manager UI application uses a self-signed SSL certificate, your browser issues a certificate error when connecting to the Oracle VM Manager UI for the first time.

You can install the Oracle VM Manager self-signed certificate on your client host to avoid SSL validation issues when you access the Oracle VM Manager UI. Alternatively, you can obtain an SSL certificate signed by a third-party certificate authority (CA) and install this new SSL certificate on your Oracle VM Manager host. You can find the procedure to change your SSL certificate in the topic titled "Changing the Default SSL Certificate," in the *Oracle VM Administrator's Guide*, Part Number E64083-01 or later.

Oracle VM Manager UI: Structure



The Oracle VM Manager UI provides a set of tabs, panes, icons, toolbars, and Global Links.

There are seven tabs on the main window of the Oracle VM Manager UI: Health, Servers and VMs, Repositories, Networking, Storage, Reports and Resources, and Jobs.

After you log in to the Oracle VM Manager UI, the window for the Health tab is shown. The screenshot in the slide shows the window displayed when you select the Servers and VMs tab.

The user interface displays context-sensitive information, relevant to the selection in the navigation (left) pane and the management (right) pane.

From the Perspective drop-down list in the management pane, you can choose a view that is valid for your selection in the navigation pane.

When you access the Oracle VM Manager UI for the first time, the *Getting Started* pane is active. You can click the Collapse button to close this pane.

Making Changes to Your Oracle VM Manager UI

After installing the Oracle VM Manager software, you can change the following in the UI:

- `admin` password
- UI session timeout
- Accessibility options
- Refresh job timeout value
- Statistics collection interval

After the successful installation of the Oracle VM Manager software, you can perform several tasks to customize your environment. The topics in this section cover the following:

- Change the admin password to better secure your environment.
- Change the UI session timeout value.
- Customize the accessibility options by selecting larger fonts, enabling the screen reader, and using a higher screen contrast.
- Change the refresh job timeout value to specify the number of seconds to wait before timing out long running jobs, like refreshing a large number of file systems on storage servers.
- Change the statistics collection interval to specify how frequently statistics are collected for servers.

Changing the admin Password

- Requires the WebLogic password
  - The WebLogic password is set during the Oracle VM Manager installation.
  - The default password is the password given to the admin user account during the Oracle VM Manager installation.
- You use the Oracle VM Manager Administrator Tool to change the password.
- You invoke the Oracle VM Manager Administrator Tool using the ovm_admin command.
  - Enter the current password.
  - Enter a new password.

The admin password is set during installation of the Oracle VM Manager.

You need the WebLogic password to change the admin password.

The WebLogic password is:

- The same as the admin password that you set during the installation of the Oracle VM Manager software

OR

- A different password if you changed the WebLogic password at any time after the installation

The Oracle VM Manager Administrator Tool (ovm_admin) provides the ability to perform various user management functions from the command line. You use this tool to change the admin password.

Procedure to Change the admin Password

- Log in to the Oracle VM Manager host as the root user and perform the following:

```
# cd /u01/app/oracle/ovm-manager-3/bin
#./ovm_admin --modifyuser
Oracle VM Manager Release 3.4.2 Admin tool
Please enter the username: admin
Please enter the current password: Welcome1
Please enter a new password for admin (minimum 8
 chars. with one numeric/special char.): MyAdmin1
 (example only)
Please re-enter the password: MyAdmin1
Please enter the password for WebLogic: Welcome1
Initializing WebLogic Scripting Tool (WLST) ...
Some command output omitted.
```

- This procedure changes only the admin password.

Log in to the Oracle VM Manager host as the root user and perform the following:

```
# cd /u01/app/oracle/ovm-manager-3/bin
#./ovm_admin --modifyuser
Oracle VM Manager Release 3.4.2 Admin tool
Please enter the username: admin
Please enter the current password: Welcome1
Please enter a new password for admin (minimum 8 chars. with one
numeric/special char.): MyAdmin1 (example only)
Please re-enter the password: MyAdmin1
Please enter the password for WebLogic: Welcome1
Initializing WebLogic Scripting Tool (WLST) ...
Welcome to WebLogic Server Administration Scripting Shell
Type help() for help on available commands
Connecting to WebLogic server ...
Connected ...
```

```
Modifying user 'admin' ...
Modified user 'admin' successfully ...
Disconnected from weblogic server: AdminServer
Exiting...
```

Note: This procedure does not change the WebLogic Server password, even if the `admin` and WebLogic passwords were the same before performing the procedure.

Changing the Oracle VM Manager UI Session Timeout Value

- Changes the amount of time the UI can remain inactive before timing out
- The default is 30 minutes (1800 seconds).
- Follow the instructions in the *Oracle VM Administrator's Guide*.

Changing the UI Session Timeout

The session timeout value for the Oracle VM Manager UI is set to 30 minutes. The Oracle VM Manager runs on Oracle WebLogic Server, and the timeout is set as part of the Oracle VM Manager UI WebLogic application.

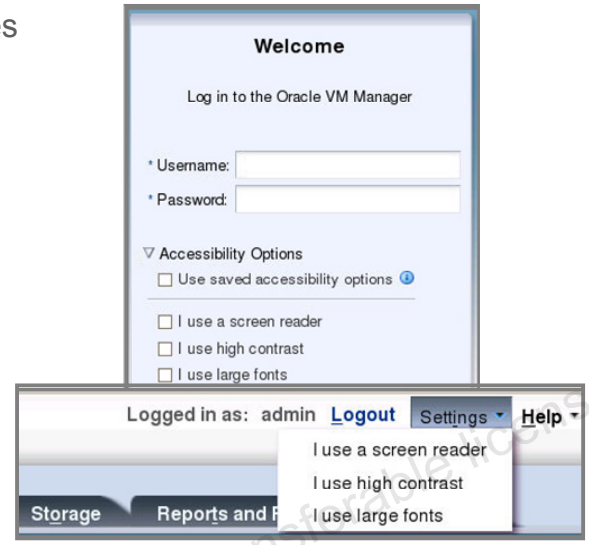
In some instances, the timeout might be longer than the value set in WebLogic because some of the views in the Oracle VM Manager update automatically and these updates also have a timeout value that can add to the Oracle VM Manager session timeout.

To change the UI timeout setting, follow the steps described in the topic titled "Configuring Oracle VM Manager Web Interface Session Timeout" in the *Oracle VM Administrator's Guide*, Part Number E64083-01 or later.

For more information about updating a WebLogic application, consult the Oracle WebLogic documentation for the release of the WebLogic Server in your Oracle VM environment. You can find the release number for WebLogic on the Welcome screen of the WebLogic Administration Console. You can access the Administration Console at https://<*Oracle VM Manager hostname*|*IP*>:7002/console. Log in as user `weblogic`, with the password that you specified during the Oracle VM Manager installation.

Changing Accessibility Options

- You can change the following accessibility features to make the Oracle VM UI more usable:
  - Support for Screen Reader
  - Support for High Contrast
  - Support for Large Fonts
- You can change these features before or after logging in to the UI:
  - From the login page
  - From the Settings drop-down menu in the Global Links section



The Oracle VM Manager UI allows you to configure the following accessibility features:

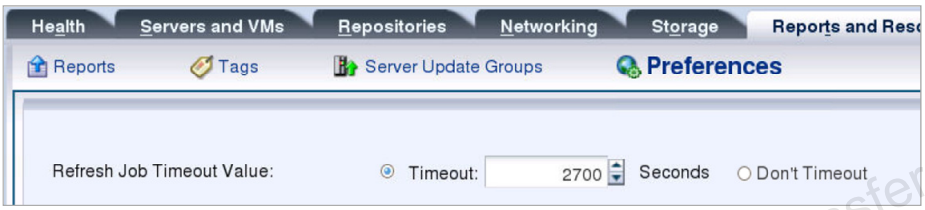
- Support for Screen Reader
- Support for High Contrast
- Support for Large Fonts

You can configure these options from two locations:

- On the Welcome screen when accessing the Oracle VM Manager UI; click the Expand button next to Accessibility Options.
- After logging in to the Oracle VM Manager UI, from the Settings menu located in the top-right corner of the Oracle VM Manager main window (as a Global Link)

Changing the Refresh Job Timeout Value

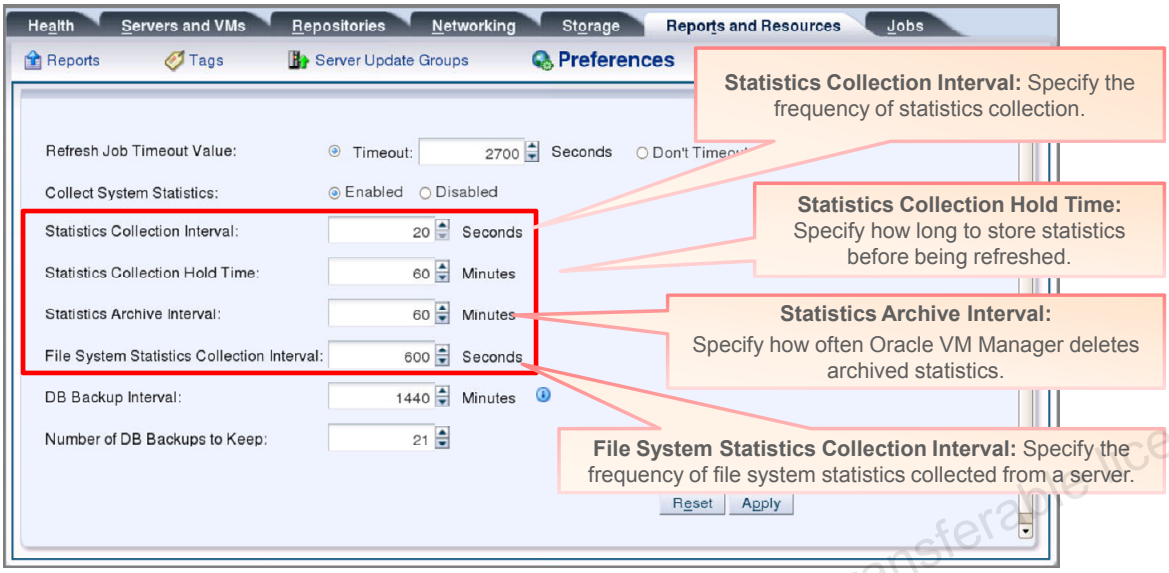
- The radio buttons determine whether to set a timeout value on long running jobs or not.
- The default value is 2700 seconds (45 minutes).
- Consider adjusting for storage servers with many file systems.
  - Adjust the timeout value if you see a `No File Systems Available` message.
  - The message usually means the value is set too low for the number of file systems.



The refresh timeout value, set at 2700 seconds during the Oracle VM Manager installation, controls whether a timeout value is applied when refreshing the list of file systems available in your environment.

You set this timeout on the Preferences subtab of the “Reports and Resources” tab. The timeout is set or unset by using a radio button. If the timeout button is set, you also specify, in the field next to the radio button, the number of seconds to wait before a refresh of available NFS file systems times out.

Changing Statistics Collection



The Oracle VM Manager periodically collects vital statistics such as memory and CPU usage.

Statistics Collection Interval: This field, located on the Preferences subtab of the “Reports and Resources” tab, allows you to specify how frequently statistics are collected for Oracle VM servers. The statistics, in conjunction with threshold values, are used on the Health tab to determine the status of Oracle VM servers and virtual machines. You can set this value between 20 seconds and 31,536,000 seconds (one year).

Statistics Collection Hold Time: This field, located below the Statistics Collection Interval, allows you to specify how long statistics are stored by the Oracle VM Manager before they are refreshed. The value for this field is between 15 minutes and 4320 minutes (3 days). The default value is 60 minutes.

Statistics Archive Interval: The Oracle VM Manager uses the value from the **Statistics Collection Hold Time** field to calculate how many archived statistics to delete. For example, if you set the hold time to 15 minutes and the archive interval to 2 days, the Oracle VM Manager would delete archived statistics that are older than 15 minutes every 2 days. The value for this field can be between 30 minutes and 525,600 minutes (1 year). The default value is 60 minutes.

File System Statistics Collection Interval: These statistics are used to display the total available disk space, the currently used disk space, and the remaining disk space for any file system that is mounted on an Oracle VM Server within the environment. You can use these statistics to monitor utilization by selecting the Server and VM Statistics subtab of the Health tab.

Quiz

Which of the following tabs can be seen from the main window of the Oracle VM Manager UI?

a. Health
b. Storage
c. Virtual Machines
d. Jobs

ORACLE

Answer: a, b, d

Oracle VM CLI

- The Oracle VM Command Line Interface (CLI) provides you with the ability to perform nearly the same functions as those available through the Oracle VM Manager UI.
- The Oracle VM CLI is installed as part of the Oracle VM Manager installation.
- The CLI is controlled by the `ovmcli` Oracle VM Manager service.
- The CLI logs are located in the `/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs` directory.
- You can extend the usability of the Oracle VM CLI with scripts.

Oracle VM CLI Service

The Oracle VM CLI is controlled by the `ovmcli` service. The `ovmm` service must also be running to use the Oracle VM CLI.

Use the `service` command to examine the status of the `ovmcli` service or to start the `ovmcli` service:

```
# service ovmcli status
Oracle VM Manager CLI is running...
```

Alternatively, you can use the `systemctl status ovmcli` command on hosts running the Oracle Linux 7 operating system.

Oracle VM CLI Logs (`CLI.log` and `CLIAudit.log`)

When you execute commands within the Oracle VM CLI session, the results of these commands are logged to the `CLI.log` file, located in the `/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs` directory.

In addition to this log file, the `CLIAudit.log` file (in the same directory) logs each command issued.

Some actions, like the `help` command or `exit` command, are not logged to the `CLIAudit.log` file. These actions appear in the `CLI.log` file.

The `CLIAudit.log` file contains the following information:

- Time stamp
- Client IP address
- Username
- Command

The log files are rotated when the file size reaches 5 MB, with up to 10 rotations, in the same way as the other Oracle VM log files.

You can change the default location of the Oracle VM CLI log files by modifying the `CLIConfigParams.file`. You learn how to customize the Oracle VM CLI later in this topic, in the slide titled “Customizing Your Oracle VM CLI Sessions.”

Oracle VM CLI and Scripting

You can extend the usefulness of the Oracle VM CLI by creating scripts with CLI commands to perform tasks such as discovering servers and creating server pools and virtual machines. Moreover, the Oracle VM CLI is a powerful tool to extract information about your environment that can be used for reporting and as part of your backup and data protection planning.

Connecting to the Oracle VM CLI

- You can initiate an Oracle VM CLI session from any host with network access to your Oracle VM Manager host.
- Use the following command to connect to the Oracle VM:
  - From the Oracle VM Manager host:

```
# ssh -l admin localhost -p 10000 or ssh admin@localhost -p 10000
```

  - Remotely:

```
# ssh -l admin <hostname of OVM Mgr host> -p 10000
```

- When connecting, you are prompted for the password of the `admin` user.

Use the following command to connect to the Oracle VM CLI from the Oracle VM Manager host:

```
# ssh -l admin localhost -p 10000
```

or

```
# ssh admin@localhost -p 10000
```

If accessing the Oracle VM CLI remotely:

```
# ssh -l admin <IP address or hostname of Oracle VM Manager host> \ -p
10000
```

Example:

```
# ssh -l admin 10.150.36.227 -p 10000
```

Connecting for the First Time

When connecting for the first time, you are prompted to accept the host key of the Oracle VM Manager host. Respond `yes` to continue with the connection. The host key for the Oracle VM Manager host is stored in the `known_hosts` file in the `.ssh` directory in your home directory. You are also prompted for the password for the `admin` user.

Connecting with Key-Based Authentication

You can also connect using public-key authentication by configuring public key–based Secure Shell (SSH) authentication.

To review the procedure for this task, consult the topic titled “Key-based Authentication” in the *Oracle VM Manager Command Line Interface User’s Guide*, Part Number E64086-01 or later.

Multiple Connections

You can make multiple connections to the Oracle VM CLI.

Using the Oracle VM CLI

After connecting and providing the password for the `admin` user, the Oracle VM CLI prompt appears:

```
# ssh -l admin myManager.example.com -p 10000
admin@myManager.example.com's password: Welcome1
OVM> ?      ← At the prompt, enter ? for a list of commands
     add
     create
     ….
     showversion
OVM> list server ← Command is verb + object
Command: list server
Status: Success
Time: 2016-05-19 11:45:12,482 MDT
Data:
  id:ff:20:00:08:ff:ff:ff:ff:ff:ff:d6:69:9a:4f:14:00  name:ovm2
  id:ff:20:00:08:ff:ff:ff:ff:ff:ff:9c:90:a6:4f:14:00  name:ovm1
OVM>
```

You can also use an SSH client like PuTTY to access the Oracle VM CLI.

The Oracle VM CLI prompt is `OVM>`.

Oracle VM CLI Usage

- To display the list of commands available, start with the `?` option.
- To find the syntax of a particular command, enter the command followed by the `?` option. The list of relevant objects for the command is displayed.
- Work your way through the command, by entering the next object, plus a space and the `?` option, until you have specified all required parameters.
- Tab completion is supported.
- You can use the Up and Down keys to recall commands from the history of commands entered for the current session.
- If the command requires a particular object, you state this object's identity by using its name or ID:

  Examples:

  ```
  OVM> add Server name=MyServer to ServerPool name=MyServerPool
  OVM> show vm id=0004fb0000140000f77dbd46637d0d24
  ```

- Commands are not case-sensitive. Data values that you use for objects are case-sensitive.

  `OVM> **List VM**` and `**list VM**` return the same information.

  `OVM> **show vm=MyVM**` and `OVM> **show vm=myVM**` do not return the same results.

- You can configure the output mode to define how the Oracle VM CLI returns results, for example in plain text or in XML, by using the `set` command:

  `OVM> **set OutputMode=Xml**`

You can find the complete list of commands and their usage in the document titled *Oracle VM Manager Command Line Interface User's Guide*, Part Number E64086-01 or later.

More Oracle VM CLI Usage

- To run a CLI command with the SSH connection:

```
# ssh -l admin localhost -p 10000 list server
```

  - Alternatively:

```
# ssh admin@localhost -p 10000 list server
```

- To run a command in the background with `&` and `-n` parameters:

```
 # ssh -l admin localhost -p 10000 -n "importVirtualAppliance Repository
name=MyRepository url=http://myserver.example.com/myassembly.ova" &
```

- To configure the output mode for results:

```
OVM> set OutputMode={Verbose|Sparse|Xml}
```

  - For example, to keep the output to a minimum:

```
OVM> set OutputMode=Sparse
```

CLI Command with the SSH Connection

You can pass a command directly as part of the `ssh` command in two ways:

```
# ssh -l admin localhost -p 10000 list server
admin@localhost's password: Welcome1
OVM> list server
Command: list server
Status: Success
Time: 2016-08-03 22:17:40,561 UTC
Data:
  id:b8:8b:9f:ea:a8:6e:da:0f:6d:74:51:a9:42:86:71:86
name:ovsvr01.example.com
  id:39:7e:44:79:65:0b:5b:d7:ec:60:d1:ed:b0:d3:e2:99
name:ovsvr02.example.com
OVM> Connection closed.
#
```

Alternatively, using `# ssh admin@localhost -p 10000 list server` returns the same results.

You can also run a command in the background by using the Linux `&` parameter. Add the `-n` parameter, which sends keyboard input, like keyboard breaks, to `/dev/null`. Running a command in the background is useful for long-running jobs like import or copy operations.

Example:

```
# ssh -l admin localhost -p 10000 -n "importVirtualAppliance Repository \
name=fc_repos1 url=http://10.150.36.208/OVM_OL6U5_x86_64_PVM.ova" &
```

You can run this type of command only with key-based authentication because you cannot interact with the command running in the background.

Options for the Oracle VM CLI

With the `set` command, you can control several behaviors of the Oracle VM CLI.

- Configuring the Output Mode: This option determines the amount of information returned with the results of your commands:
  - Verbose mode includes the command being executed, the status, time and time zone.
  - Sparse mode includes just the results.
  - Xml returns the results in XML format.

  For example, to keep the output to a minimum:

  ```
  OVM> set OutputMode=Sparse
  ```
- Configuring the command timeout value: You can set the `CommandTimeout` value between 1 and 43200 seconds. The default is 7200 or two hours.
- Configuring the end of line control characters: You can set the `EndlineChars` parameter to CRLF, CR, or LF, depending on your client system. LF is used on Linux and UNIX-like systems.
- Configuring whether the CLI runs in synchronous or asynchronous mode: The `CommandMode` parameter, when set to `Asynchronous`, is useful to regain control of the CLI prompt for a long-running job like an import operation. When the job completes, you can reset the parameter `CommandMode` to its `Synchronous` default value.

Finding Objects and Commands

- To display the list of all objects:

```
OVM> showobjtypes
    AccessGroup
    AntiAffinityGroup
    ....
    Vnic
    VolumeGroup
OVM>
```

- To display special commands for an object type:

```
OVM> showcustomcmds job
    abort
    getDebugTranscript
    getQueuedJobInfo
OVM> showcustomcmds port
OVM>
```

Finding the List of Available Objects

When constructing commands with the Oracle VM CLI, you often need to find the name of the object, as well as the commands available to manipulate the object.

The `showobjtypes` CLI command displays the list of available objects.

Basic Commands to Manipulate Objects

The basic commands to manipulate objects are `list`, `show`, `create` or `add`, `edit`, and `delete` or `remove`. For example, to create a new server pool, use the command `create ServerPool`. To add a server to the new server pool, use the command `add Server`.

Custom (or Advanced) Commands for Objects

When manipulating objects, you might want to perform an action beyond the scope provided by the basic commands. For example, after creating (or discovering) a file server with the `create FileServer` command, you must perform additional actions on the new file server, such as adding an admin server or refreshing the file server. Use the `showcustomcmds <object>` to list the additional actions against a particular object. In this example, to display the list of available “custom” commands for the file server object, use the following command:

```
showcustomcmds job
```

```
   addAdminServer
   removeAdminServer
   addRefreshServer
   removeRefreshServer
   refresh
OVM>
```

If an object does not offer commands beyond what is available with the basic commands, the `showcustomcmds <object>` command returns no results. Example:

```
OVM> showcustomcmds port
OVM>
```

For a latest list of CLI commands, including modifications to commands by a release and new commands, consult the latest version of the *Oracle VM Manager Command Line Interface User's Guide* (for example: E64086-01).

Customizing Your Oracle VM CLI Sessions

- The configuration file for the Oracle VM CLI is stored in the `/u01/app/oracle/ovm-manager-3/ovm_cli/config/CLIConfigParams.xml` file.
- By changing the parameters in this file, you can change the behavior of the Oracle VM CLI.
- Examples:
  - Change the port on which the Oracle VM CLI accept connections:
    `sshPort="15000"`
  - Change the CLI timeout from its default value of 45 minutes to 60 minutes:
    `clientInactivityTimeout="60"`

When the Oracle VM CLI starts, it reads its CLI configuration file. The configuration file is located in the directory `/u01/app/oracle/ovm-manager-3/ovm_cli/config`.

Keeping Your Oracle VM CLI Session Alive

You can change the Oracle VM CLI timeout value from its default of 45 minutes set in the `CLIConfigParams.xml` file, to a value between 1 and 1440 minutes (1 day). However, you might be disconnected from your session even if the value for the `clientInactivityTimeout` parameter in the `CLIConfigParams.xml` file has not been reached. This situation can be caused by inactivity for your SSH session or by a firewall rule in your environment.

To keep your SSH session alive for a longer period of time, use one of the following techniques to change the `ServerAliveInterval` in effect for your SSH session.

- Invoke the Oracle VM CLI with the `ServerAliveInterval` option.

  Example:

  ```
  # ssh admin@localhost -p 10000 -o ServerAliveInterval=60
  ```

  OR

- Include the `ServerAliveInterval` option in the `.ssh/config` file in your home directory:

  ```
  Host 10.150.36.227 ← IP address of the Oracle VM Manager host
      ServerAliveInterval    40 ← Interval in seconds
  ```

The `ServerAliveInterval` option is used to keep the SSH connection alive. It specifies the number of seconds of inactivity before the client sends a message to the server, requesting a response from the server.

Changing the Connection Channel Expiration for Public Key Authentication

Previously in this topic, you learned that you can set up public key authentication to connect to the Oracle VM CLI. After you configure public key authentication, you are no longer prompted for a password when accessing the Oracle VM CLI, and the connection channel is identified to the Oracle VM CLI Server by your client IP address and username.

The connection channel for public key authentication expires after a designated period of time, or if the Oracle VM Manager is restarted. The default for keeping the channel open is 1 week (10080 minutes). You can modify this setting by editing the `CLIConfigParams.xml` file, located in the `/u01/app/oracle/ovm-manager-3/ovm_cli/config` directory. Change the value for the `publicKeyAuthChannelTimeout` parameter, set at "10080," to a value of your choice.

A value of `-1` keeps the channel open indefinitely.

Example:

```
publicKeyAuthChannelTimeout="-1"
```

Quiz

Which command can be used to access the Oracle VM CLI locally from host `ovmmgr01.example.com` where the Oracle VM Manager software is installed?

a. `ssh -l admin ovmmgr01.example.com -p 7002`

b. `ssh admin -l localhost -p 7002`

c. `ssh -l admin ovmmgr01.example.com -p 1000`

d. `ssh -l admin localhost -p 10000`

Answer: d

Summary

In this lesson, you should have learned how to:

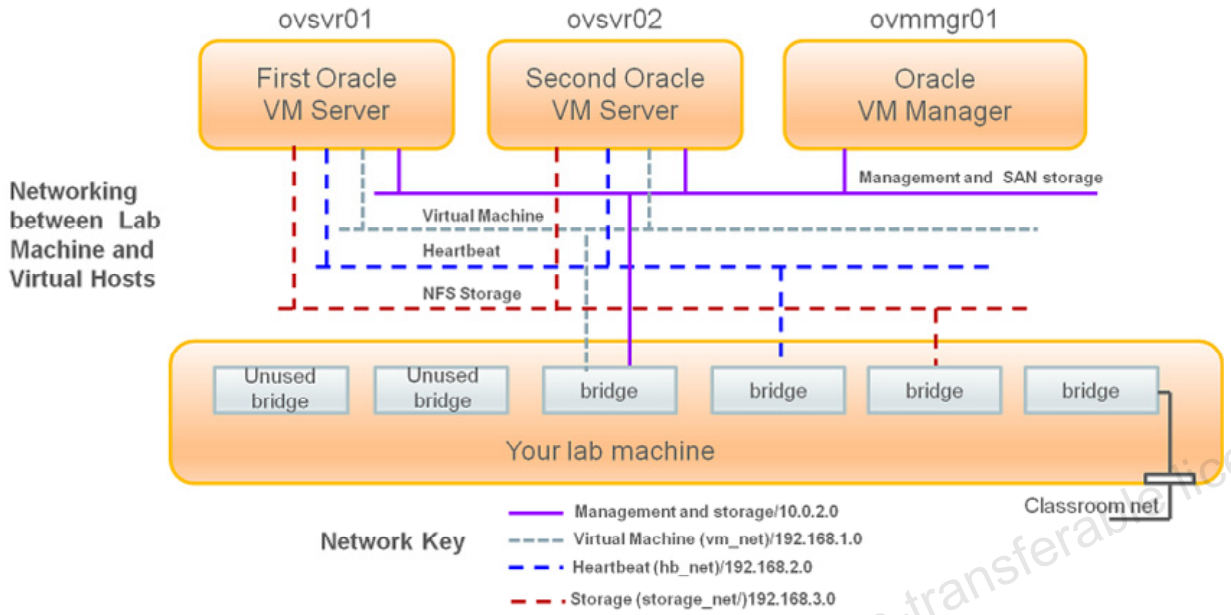
- Describe the major components of the Oracle VM environment
- List the interfaces available to manage your Oracle VM environment
- Describe the control elements of the Oracle VM Manager UI
- Connect to and use the Oracle VM CLI and configure its behavior
- Use the Oracle VM CLI from the command line

Practice Overview

In this practice, you:

- Get familiar with the hosts and networks in your practice environment
- Access your lab machine and exercise ways to access the virtual machines running on your lab machine
- Access the Oracle VM Manager UI and examine the initial state of your Oracle VM environment
- Execute the Refresh All action from the Oracle VM Manager UI
- Start the Oracle VM CLI, perform basic operations, and examine the Oracle VM CLI's configuration and log files

Your Practice Environment



Your lab machine is a server, which you access directly (in the classroom) or remotely.

Your environment contains three hosts running as virtual machines on your lab machine.

- `ovsvr01.example.com` is the first Oracle VM server.
- `ovsvr02.example.com` is the second Oracle VM server.
- `ovmmgr01.example.com` is the host where your Oracle VM Manager runs.

Note: The `host01` virtual machine that is used for the P2V practice exercise is not shown in the diagram in the slide.

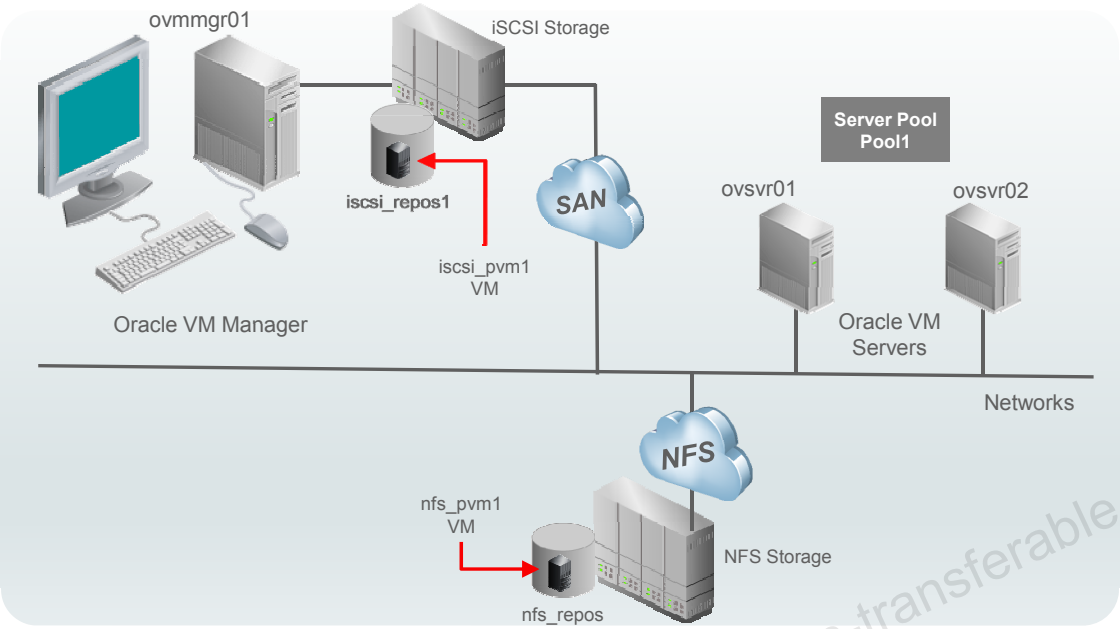
When you need access to the virtual hosts as part of a practice, you log in to them from your lab machine.

Your environment contains the following networks:

- **Classroom network:** This network allows your lab machine to communicate with the outside. Your Oracle VM environment does not use this network. This network's subnet depends on the location for your lab environment.
- **Management network:** This network allows your three virtual hosts to communicate with your lab machine. It also allows the hosts to communicate with each other. This network is configured on the `192.0.2.0` subnet. This network also allows the two Oracle VM servers deployed from your Oracle VM environment to access the iSCSI storage server configured on the Oracle VM Manager host.

- **Virtual Machine network:** This network allows the virtual machines deployed from your Oracle VM environment to communicate with each other, and also to communicate with your lab machine. This network is configured on the `192.168.1.0` subnet.
- **Heartbeat network:** This network allows the two Oracle VM servers deployed as part of your Oracle VM environment to communicate with each other, for the heartbeat function of your Oracle VM server pool. This network is configured on the `192.168.2.0` subnet.
- **Storage network:** This network allows the two Oracle VM servers deployed from your Oracle VM environment to communicate with the NFS share on your classroom lab machine. This network in configured on the `192.168.3.0` subnet.

Initial State of Your Oracle VM Environment



Your Oracle VM environment has been pre-built and pre-configured for you. This environment contains the following components:

- The Oracle VM Manager host, with host name `ovmmgr01.example.com`. The Oracle VM Manager software has been installed in the `ovmmgr01` host.
- Two Oracle VM servers, with host names `ovsvr01.example.com` and `ovsvr02.example.com`. The Oracle VM Server for x86 software has been installed in the `ovsvr01` and `ovsvr02` hosts.
- The networking infrastructure, which is described in the previous slide
- An NFS server and an iSCSI storage server. The storage elements for these two servers have been configured and are accessible by the two Oracle VM servers.
- A clustered server pool named `Pool1`. The server pool has been created and both Oracle VM servers have been added to this server pool.
- Two repositories:
  - An NFS-type repository named `nfs_repos` located on a share in the NFS storage
  - An iSCSI-type repository named `iscsi_repos1` located on a LUN in the iSCSI storage

Both repositories have been presented to the Oracle VM servers in the `Pool1` server pool.

Two virtual machines:

- The `iscsi_pvm1` virtual machine's resources reside in the `iscsi_repos1` repository. This virtual machine was installed from an Oracle Linux 6.5 ISO file as a basic server.
- The `nfs_pvm1` virtual machine's resources reside in the `nfs_repos1` repository. This virtual machine was cloned from an assembly containing an Oracle Linux 6.5 virtual machine.

2

Oracle VM Web Services

Objectives

After completing this lesson, you should be able to:

- Explain the role of Oracle VM Web Services
- Describe the REST interface of Oracle VM Web Services
- Use simple coding examples with the REST interface
- Locate and navigate the Oracle VM Web Services SDK file

Oracle VM Web Services

- Web services provide a programmable way to interact with an application across HTTP.
- Oracle VM Web Services expose an Application Programming Interface (API) that you can use to build applications to view, create, update, and remove objects in your Oracle VM environment.

What Are Web Services?

Web services provide a way to interact with web resources over HTTP or HTTPS. For example, if a website provides weather information, you can build a request to this site to obtain weather information for a particular city and/or date.

The web services provider presents a network address, as a URL, and a set of resources that can be queried.

Web services are not limited to queries only. Web services providers can expose a set of operations that act on objects, or even create new objects.

Web services generally format requests and responses in a parsable format, such as XML or JSON. This makes it possible for you to construct requests that can be parsed by the web services, and receive responses that can be parsed by your client application.

Note: XML and JSON are formats for encoding the information exchanged between applications.

The Oracle VM Web Services Application Programming Interface (WS-API) provides a web services interface for building applications to access the Oracle VM Manager.

For releases of Oracle VM before 3.4.1, the Oracle VM WS-API provides two different types of interactions to the same core functions: RESTful and SOAP; however, the SOAP API is deprecated as of Release 3.4. Oracle recommends using the REST web service for any client application.

Both interfaces expose exactly the same functionality.

RESTful and REST are used interchangeably in this lesson.

Using the Oracle VM WS-API

- All functionality available with the Oracle VM Manager is exposed through the Oracle VM WS-API.
- You can create programs with the Oracle VM WS-API to perform the following tasks:
  - Automating configuration
  - Monitoring and notification
  - Triggering behaviors
- The Oracle VM WS-API is language agnostic:
  - Select REST as the preferred programming language and a web services interface.
- Oracle VM provides a software development kit (SDK) to help you build applications.

The Oracle VM WS-API is used to provide a programmable interface to the Oracle VM Manager functionality. You can use the Oracle VM WS-API to automate the configuration of your Oracle VM environment, monitor certain conditions (such as repository usage), or trigger events (such as switching a server pool policy based on time of day).

To build a program that uses the functionality provided by the Oracle VM WS-API, you use the REST programming language for Oracle VM 3.4.2 environments.

Because web services use well-known protocols and data formats, you might choose to install programming libraries available for your programming language to assist you with any development tasks. For example, you might want to install the Python Requests library that offers support for HTTP basic authentication. An example using the Requests library is provided later in this lesson.

If you choose Java as the programming language, you can take advantage of the client-side package available in the software development kit (SDK) bundled in the Oracle VM Manager installation ISO file.

The SDK file is called `OvmSDK_3.4.2.<Build Number>.zip`.

This file contains:

- Documentation for the Oracle VM Web Services API as exposed by the provided library and sample source code
- The Oracle VM web services client library
- Sample Java source code that you can use to start building your own client application by using either the SOAP or REST API

  Note: The Oracle VM Web Services API documentation for Oracle VM 3.4 still contains information for both SOAP and REST.

REST

- REST APIs use different HTTP methods or “verbs” to trigger actions on an object.
  - Methods include GET, POST, PUT, and DELETE.
- The object is referenced by using a URI that identifies it.

  Example: `https://hostname:port/ovm/core/wsapi/rest/Server/<uuid>`
- Parameters required by a method are usually contained in the body of the HTTP request in a parsable format.
- XML or JSON is the Internet media type used for parameters or response objects.
- You specify the format in the headers of the HTTP request.

Building a Uniform Resource Identifier (URI)

For all REST requests, you construct a URI for each request in your client application, for example, in your Python program. These requests are constructed by using:

- A base URI, for example, `https://hostname:port/ovm/core/wsapi/rest`
- A URI pattern or mapping that indicates the exact object or action that you wish to perform, for example: `https://hostname:port/ovm/core/wsapi/rest/ServerPool`

Setting the Internet Media Type

Depending on the client application and the parser that you prefer, you can set the Internet media type used to encode data sent in requests or received in response to either XML or JSON. You achieve this selection by setting the “Accept” and “Content-Type” HTTP headers to either “application/json” or “application/xml.” An example is provided in the next slide titled “HTTP GET Request Example.”

Testing the REST API

You can test the REST API by using a standard web browser, if you have a way to set HTTP headers in your requests.

In Firefox, you can use the Modify Headers plug-in to modify the headers of your HTTP requests to include authentication information and set the Internet media type. Using a standard web browser, you can perform queries to list objects and their attributes.

REST Authentication

- Authentication in REST is achieved by using HTTP Basic authentication.
  - You submit a `base64` encoded username and password combination in the HTTP Authorization header of your request.
- After authentication, a session is created and a cookie is used for subsequent requests.
- Using Python, you can take advantage of a library such as the Requests library to:
  - Set up a session
  - Handle authentication, including encoding of username and password
  - Update headers
  - Perform a basic request

On the first HTTP request from your client application, you must authenticate. Authentication is achieved by using HTTP Basic Authentication, as described in RFC 2617. This involves sending a Base64 encoded representation of the string `username:password` within the HTTP Authorization header of your request.

Base64 is an encoding scheme to convert strings into a common form for electronic transmission.

For example, if your username is `admin` and password is `Welcome1`, you construct an Authorization header containing the value `YWRtaW46V2VsY29tZTE=`. This step can be greatly simplified if you use a library, such as the Requests library, that takes care of the proper representation for the username and password.

The Requests library is an HTTP library for Python that is written in Python. You can download the library from this site: http://docs.python-requests.org/en/latest/. You can use the installer program for the operating system of your Oracle VM Manager host system. For example, with Oracle Linux 7.1, you can use `yum install python-requests` to install the library.

After the client has been successfully authenticated, the HTTP response from the request includes a cookie containing session information. All future requests during a session make use of the same cookie data to show that authentication has already been achieved.

HTTP GET Request: Example

Python script to list Oracle VM servers and their attributes

```
import requests
s=requests.Session()
s.auth=('admin','Welcome1')
s.headers.update({'Accept': 'application/json', 'Content-Type':
   'application/json'}) baseUri='https://ovmm.example.com:7002/ovm/core/wsapi/rest'
r=s.get(baseUri+'/Server')
print r.json()
```

The Python script in the slide makes a request to the Oracle VM Manager to list all the Oracle VM servers and their attributes. The script communicates with the Oracle VM Manager by using the Oracle VM WS-API.

The following numbered explanations provide a description of each line in the Python script:

1. Imports the Python Requests library that handles all HTTP requests, authentication, and session maintenance
2. Instantiates a session object from the Requests library, which you can use for all of your HTTP requests. This session is referenced as `s` in the rest of the script.
3. Provides authentication information for the session object. The information, consisting of a username and password, is required for HTTP basic authentication.

   The Requests library automatically converts the information to a base64 encoded string. The information is used only to authenticate during the first connection. From that point, the Requests library uses a cookie to handle subsequent session authentication.
4. Updates the headers for all future requests within the established session to set the Internet media type. In this case, the Internet media type is set to JSON.
5. Sets a variable called `baseUri` to the base URI to access the REST API provided by the Oracle VM WS-API

   This variable is then used to create complete URIs in subsequent requests.

6. Uses the established session to send an HTTP `GET` request to the REST API provided by the Oracle VM Manager

   The `GET` requests a list of all Oracle VM servers and their attributes.

7. Uses the Requests library's default JSON parser to parse the response and print it to screen

HTTP POST Request: Example

Python script to create a network object

```
import json
 data = {
         'name':'MyNetwork',
         'description': 'A test network using the REST API',
     }
uri='{base}/Network'.format(base=baseUri) r=s.post(uri,data=json.dumps(data))
print r.json()
```


The Python script in the slide builds on the example in the first script, where you already instantiated and configured a session object by using the Requests library. If you want to run this code as a separate script, you need to add most of the steps in the previous script.

The following numbered explanations provide a description of each line in the Python script:

1. Imports the JSON library that is needed to encode a Python data object as a JSON string that the REST API can parse
2. The `data` code block creates a Python data object containing the keys and values expected by the REST API for the `POST` operation to be performed.
3. Creates a variable containing the full URI to create a `Network` object in the REST API. The variable is created by concatenating the `baseURI` (defined in the previous script) with the path to the Network method. Performing this step as a separate operation keeps the code concise and readable.
4. Uses the Requests session established in the previous script to submit a `POST` request to the URI that was constructed in the previous line

   In this `POST` request, the `data` parameter is first encoded as a JSON string that the REST API can parse. This step is performed by using the JSON library's `dumps` method.

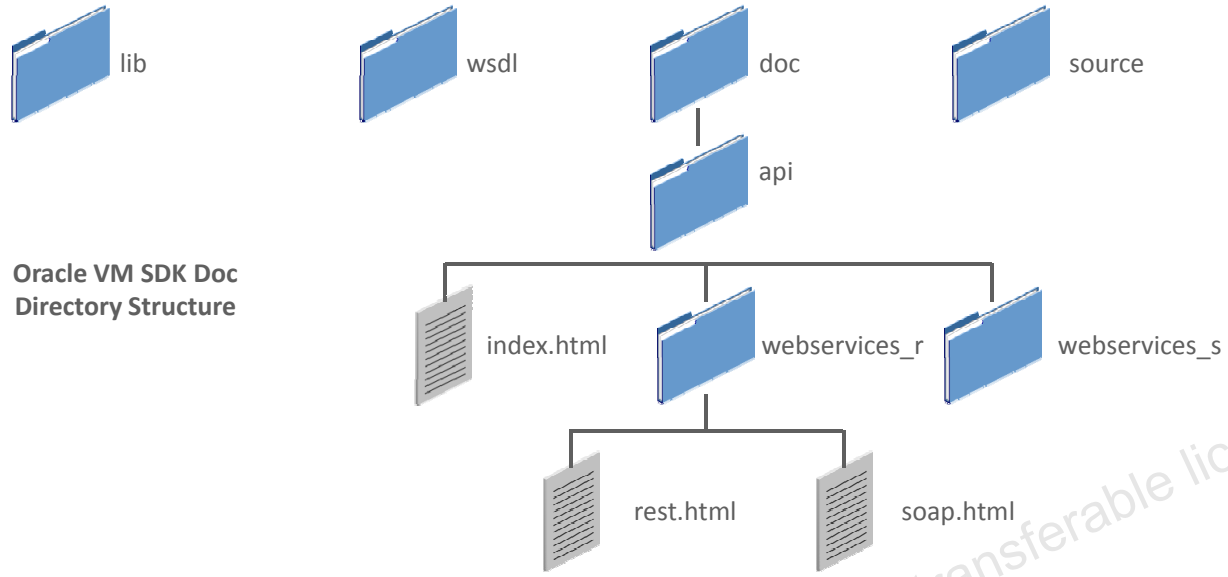
   The resulting `data` parameter is then attached to the `POST` request.

5. Prints the JSON response returned by the REST API as a result of the request

   The API returns a job object describing the job that was triggered to perform the task of creating a new network.

A full description of the Oracle VM Manager REST API, including the URI mappings and the accepted HTTP methods, as well as a full description of the attributes for any object type is provided in the SDK. Examples of some of the actions that you can perform by using the REST API are provided for both Java and Python in the *Oracle VM Web Services API Developer's Guide*, Part Number E64087-01 or later.

Navigating the Oracle VM WS-API SDK



The diagram in the slide shows the structure of the `doc` directory within the SDK file. You can find the SDK file in the top level directory of the installation ISO file for the Oracle VM Manager software.

Client Applications with Java

If you want to develop applications with Java, you can use the sample client Java program in the `source` directory structure. The Java sample program uses a set of client-side classes that hides the details of the REST invocation. These sample classes are distributed in the `.jar` file found in the `lib` directory.

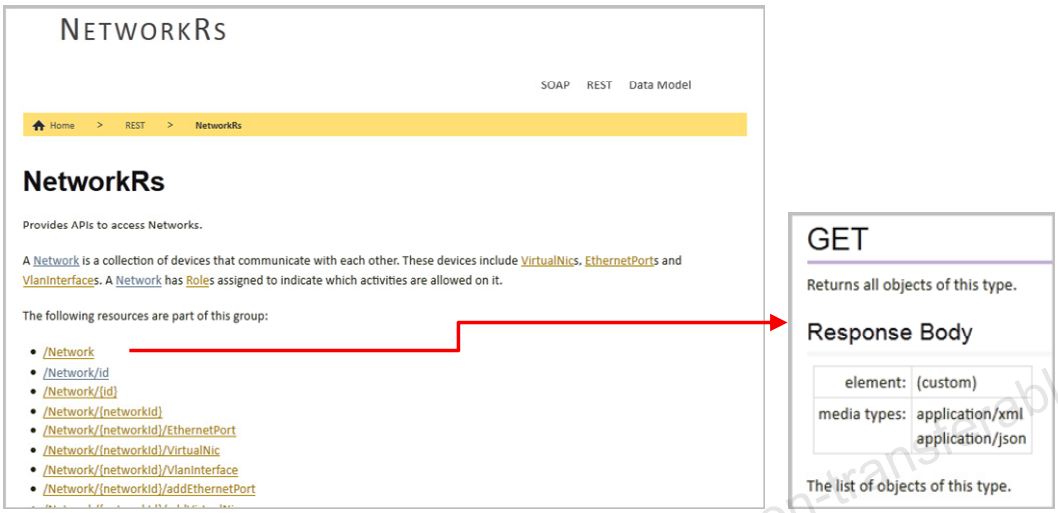
Client Applications with REST

If you want to develop REST-based applications by using a programming language such as Python, you might want to access the list of all resources that you can manipulate in your programs. You can find this list in the SDK file by opening the `index.html` file in the `doc/api` directory of the SDK file from your browser. From that location, select Web Services API via REST. This selection enables you to browse the information stored in the `webservices_r` directory. From this location, you can examine the resources and the operations available on these resources. The information in this directory was created by using Enunciate.

Note: The `soap.html` file still exists in the `doc/api` directory of the SDK directory structure.

Browsing RESTful API Resources

SDK path: `file:///stage/doc/api/webservices_r/resource_NetworkRs.html`



Resources with the RESTful API

The larger screenshot in the slide provides an example of the information provided in the `webservices_r` directory for the Network resource.

Resources have names such as Network or Job. You create new objects or manipulate existing objects by specifying the appropriate resources in the URI.

For example, use `/rest/Network` in the URI as part of a GET operation to return all objects of type `Network`. The `GET` operation for the Network resource is shown in the smaller screenshot in the slide.

To create a new network, use `/rest/Network` in the URI as part of a `POST` operation and provide a network name and other parameters in the body of the request. The network name is a required parameter.

Some resources are created as part of other resources. For example, use `/rest/FileSystem/{fileSystemId}/Repository` in a `POST` request to create a repository, because a repository is dependent on the file system on which the repository is to be created.

Note: When you are creating new resources by using the RESTful API, you must provide at least the parameters that are required for creating the new resource. You format the parameter information in XML or JSON format and provide this information programmatically as part of the HTTP POST request. This process is described in this lesson, in the slide titled “REST.”

Quiz

When creating a REST request to the Oracle VM WS-API, which HTTP verb would you choose to create a new network?

a. NEW
b. BUILD
c. POST
d. PUT

Answer: c

Summary

In this lesson, you should have learned how to:

- Explain the role of Oracle VM Web Services
- Describe the REST interface of Oracle VM Web Services
- Use simple coding examples with the REST interface
- Locate and navigate the Oracle VM Web Services SDK file

Practice Overview

In this practice, you:

- Set up your environment to use the Oracle VM Web Services API with REST services
- Access the Oracle VM Web Services API with REST services from your web browser
- Access the Oracle VM Web Services API with REST services by using an interactive Python session
- Locate and navigate the Oracle VM Web Services SDK file

3

Managing Virtual Machines

Objectives

After completing this lesson, you should be able to:

- List and compare virtualization types
- Identify and illustrate the various methods for creating virtual machines
- Manage virtual machine components such as disks, hot plug-in operations for disks and network, and dynamic CPU and memory changes
- Clone, migrate, and modify virtual machines

Lesson Overview

- The first section of this lesson discusses virtualization modes available for virtual machines.
- In the next section, you explore the ways to create virtual machines.
- After learning how to create virtual machines, you step through several types of operations on these virtual machines, including cloning and migrating the resources of the virtual machines.

Oracle VM Virtualization Modes

- Virtual machines run in one of three modes:
  - Paravirtualized mode (PVM)
  - Hardware virtualized mode (HVM)
  - Hardware virtualized mode using paravirtualized drivers (PVHVM)
- When creating a virtual machine by using the Oracle VM Manager UI or the Oracle VM CLI, you choose its virtualization mode by selecting a domain type. The choices are:
  - XEN_HVM
  - XEN_HVM_PV_DRIVERS
  - XEN_PVM
  - LDOMS_PVM
  - Unknown

Virtualization Modes

Virtual machines run in one of three modes: paravirtualized (PVM), hardware virtualized (HVM), or hardware virtualized by using paravirtualized drivers (PVHVM).

Note

- With Oracle VM, a virtual machine represents a domain with allocated resources. In the following topic, the term “guest” is used to refer to the operating system running in the virtual machine’s domain.
- When creating a virtual machine by using the Oracle VM Manager UI, you select its virtualization mode by selecting a domain type. The choices are XEN_HVM, XEN_HVM_PV_DRIVERS, XEN_PVM, LDOMS_PVM, or Unknown if the domain type is not known.

In this topic, you learn about the characteristics, requirements, advantages, and disadvantages of the three virtualization modes. Note however that the virtualization spectrum is constantly experiencing changes, as virtualization techniques improve and hardware virtualization support expands. Even the terminology is in flux.

Consult the documentation, particularly the Oracle VM Release notes, to find the latest information about the virtualization support with Oracle VM. You can also find information about the virtualization spectrum at the Xen Project website http://xenproject.org/.

Paravirtualized Mode

- In paravirtualized mode, the kernel is aware of the virtual environment.
- The guest runs in a XEN_PVM domain type on the Oracle VM server.
- The Oracle VM server does not require hardware virtualization assist in the system BIOS.
- Because there is no emulated BIOS for PVM guests, you cannot boot your guest OS using an emulated CD-ROM/DVD and an ISO file.

PVM Requirements

The guest runs in a virtual machine with the domain type set to XEN_PVM on the Oracle VM server. With fully paravirtualized guests, the host platform does not require hardware virtualization assist in the system BIOS.

PVM Advantages and Disadvantages

The guest requires an optimized kernel and special drivers to manage virtual resources and their associated physical resources. This requires changes to the OS running in the guest.

Some operating systems, like Microsoft Windows, do not support paravirtualized mode. However, Windows can take advantage of available Oracle VM paravirtualized drivers to optimize access to virtual disks and network.

Paravirtualized drivers are discussed when the PVHVM virtualization mode is introduced later in this topic.

PVM supports a larger number of disk devices compared to the other virtualization modes. To view virtual machine maximums, refer to the “Configuration Limits” topic in the Oracle VM release notes for your release of Oracle VM.

Hardware Virtualized Mode

- The HVM guest operating system can run completely unmodified.
- The Oracle VM servers must run on hardware that supports virtualization.
- You can boot the guest by using the virtual CD-ROM/DVD with an ISO file residing in an accessible repository.
- There is a restriction on the maximum number of emulated integrated development environment (IDE) and SCSI devices.

With Oracle VM, if your server hardware supports hardware virtualization (either Intel VT or AMD-V), the guest operating system can run completely unmodified. The guest operates using the native kernel and the operating system–supplied drivers. The guest has no knowledge that it runs in a virtualized environment.

HVM Requirements

The Oracle VM servers must run on hardware that supports virtualization and this support must be enabled in the BIOS.

HVM Advantages and Disadvantages

- In the current implementation, there might be a performance penalty for certain types of guests and access types, but hardware virtualization also allows many Microsoft Windows operating systems and legacy operating systems to run unmodified.
- You can install the OS in your guests using the virtual CD-ROM/DVD with an ISO file residing in an accessible repository.
- There is a restriction on the maximum number of emulated IDE and SCSI devices.

Hardware Virtualized with Paravirtualized Drivers

- PVHVM guests are similar to HVM guests, but the guests can be configured to use paravirtualized drivers for improved performance.
- For a guest running in PVHVM mode, your Oracle VM server must still provide hardware virtualization assist.
- The paravirtualized drivers must be installed in your guest.
  - Current releases of Oracle Linux have the PV drivers in the kernel packages.
- You can install the OS in your guest by using the virtual CD-ROM/DVD and selecting an ISO image.

PVHVM guests are similar to HVM guests, but the guests can be configured to use paravirtualized drivers for improved performance of the guest.

Note: Recall that for PVHVM-type virtual machines, you select the XEN_HVM_PV_DRIVERS domain type when you create your virtual machine by using the Oracle VM Manager UI or the Oracle VM CLI.

PVHVM Requirements

A guest running in the PVHVM mode still requires the virtualization support for the Oracle VM server where it runs. You must enable this support in the system BIOS.

The paravirtualized drivers must be installed in your guest. In releases of Oracle Linux (4.7, 5.3, 6.x or later, and for all Unbreakable Enterprise Kernel [UEK] kernels), you do not have to install the paravirtualized drivers because they are already present in the kernel packages. Refer to the *Oracle VM: Comparison of Guest Virtualization Modes: HVM, PVM and HVM with PV Drivers* document (Doc ID 757719.1) for more information on PVHVM requirements.

Install the paravirtualized drivers in your Microsoft Windows guest to increase its performance level. Consult *Oracle VM Paravirtual Drivers for Microsoft Windows*, Part Number E64085-03 or later for information on how to install the paravirtualized drivers in your Microsoft Windows guest.

PVHVM Advantages and Disadvantages

- PVHVM offers better performance than HVM. It might also offer better performance than PVM, for some types of OS and later releases of the OS.
- You can install the OS in your guest by using the virtual CD/DVD and selecting an ISO image.
- The installation of paravirtualized drivers require significant guest modification and can cause issues in your guest.
- Your PVHVM guests can use nearly the same number of paravirtual disks as a PVM guest.

Converting an HVM to PVM

- You can convert an HVM guest to a PVM or PVHVM guest.
  - A Microsoft Windows guest can run only as HVM or PVHVM.
- If you are running a current version of Oracle Linux or the Unbreakable Enterprise Kernel (UEK), you can change the domain type of your virtual machine from HVM to PVM.
  - Older Oracle Linux versions require additional conversion steps.

You can convert an HVM guest to a PVM or PVHVM guest.

Microsoft Windows

After installation, Microsoft Windows runs as an HVM guest. You can install paravirtual drivers for Microsoft Windows to allow the guest to run as an HVM with PV drivers. You can find information about installing the paravirtual drivers in *Oracle VM Paravirtual Drivers for Microsoft Windows*, Part Number E64085-03 or later.

Oracle Linux

If you are running a guest with a current version of Oracle Linux or the UEK, or a Red Hat Compatible kernel (RHCK), the guest kernel is a pvops kernel. For this type of guest, you can change the domain type of your virtual machine from HVM to PVM by editing the virtual machine by using one of the Oracle VM management interfaces. Pvops kernels are discussed in the next slide.

Pvops Kernel

- Pvops kernels are kernels containing both paravirtualized (PV) drivers as well as native devices drivers.
- By default, an HVM guest with a pvops kernel runs as fully virtualized with PV drivers (PVHVM).
- You can force Oracle Linux guests to run as pure HVM by adding the parameter `xen_emul_unplug=never` to the kernel entry in the `/boot/grub/grub.conf` file.

ORACLE

A pvops kernel can boot natively on bare metal or as HVM in a virtual environment, or boot fully paravirtualized in a virtual environment such as Oracle VM.

As noted previously, current Oracle Linux releases with UEK or RHCK have pvops kernels. With such kernels, the default behavior is to present only paravirtualized drivers when running an HVM guest. If you want your Oracle Linux UEK HVM guest to use HVM virtualization only, modify the `/boot/grub/grub.conf` file in the guest and append the boot option `'xen_emul_unplug=never'` to your default boot kernel. After you reboot the guest, the kernel presents emulated hardware drivers only.

In the next topic, you explore the various methods to create virtual machines. Some methods are dependent on a particular virtualization mode. These restrictions are highlighted as needed.

Creating Virtual Machines

- The next section in this lesson addresses virtual machine creation.
- You can create virtual machines by:
  - Using installation media
  - Importing virtual resources
  - Using the P2V (Physical to Virtual) utility
  - Creating your own templates or virtual appliances, and cloning these resources into new templates or virtual appliances

You create virtual machines by using any of the Oracle VM management interfaces available to you: the Oracle VM Manager UI, the Oracle VM CLI, or the Oracle VM WS-API. If you manage your Oracle VM environment by using Enterprise Manager Cloud Control, you can perform the equivalent tasks from the Enterprise Manager Cloud Control console.

Creating a Virtual Machine by Using Installation Media

- Create your virtual machine by using any of the Oracle VM Manager management interfaces.
- If you select XEN_HVM or XEN_HVM_PV_DRIVERS as the domain type from the Oracle VM Manager UI:
  - Add the ISO file as CD/DVD to the list of virtual disks
  - Select CDROM as the boot option
- If you select XEN_PVM as the domain type:
  - Prepare access to the installation media from a remote location
  - Select PXE as the boot option
  - Provide the HTTP or NFS path to the remote, exploded ISO file

To install the guest OS in a virtual machine by using the installation media, create the virtual machine first by using the Oracle VM Manager UI, the Oracle VM CLI, or the Oracle VM WS-API. With any of these management interfaces, you must specify a domain type.

Virtual Machines with a XEN_HVM or XEN_HVM_PV_DRIVERS Domain Type

If you select the XEN_HVM or XEN_HVM_PV_DRIVERS domain type from the Oracle VM Manager UI, you can use local installation media to install the guest operating system in your virtual machine. To make the installation media available to your virtual machine, select CD/DVD as the disk type and select an ISO file in an accessible repository.

In the Boot Options screen of the “Create Virtual Machine” Wizard, select CDROM.

Virtual Machines with a XEN_PVM Domain Type

If you select XEN_PVM as the domain type, you cannot use the emulated CD/DVD device. In this case, prepare access to the installation media from a remote location, using HTTP or NFS. At the remote location, make the content of the installation media available to the virtual machine by mounting the ISO file as a loop device, or by copying the contents of the ISO file to a directory location. In the Boot Options screen of the “Create Virtual Machine” Wizard, select PXE as the boot option and provide the path to the remote, exploded ISO file in the Network Boot Path field.

Creating a Virtual Machine for Installation with an ISO File: Oracle VM CLI Example

- Create the virtual machine by using the `create vm` command.
  - Specify a name, domain type, memory, repository, boot option, and server pool.
- Add a virtual disk by using the `create virtualdisk` command.
  - Specify a disk name, size, whether to use sparse allocation, and the target repository.
- Create a disk mapping for an existing ISO file by using the `create VmDiskMapping` command.
  - Specify the disk slot, the ISO file disk name, and the target virtual machine.
- Start the virtual machine by using the `start vm` command.

The following steps demonstrate one way to create a virtual machine with the Oracle VM CLI and prepare it for installation by using an ISO file:

1. Create a virtual machine, specify the domain type as HVM or HVM with PV drivers, specify the name of the repository where the virtual machine configuration is to be stored, and specify the boot option to trigger a boot from the CDROM device.

```
OVM> create vm name=myvm domainType=XEN_HVM memory=1024 repository=fc_repos1
bootOrder=CDROM on ServerPool name=Pool1
Command: create vm name=myvm domainType=XEN_HVM repository=fc_repos1
bootOrder=CDROM on ServerPool name=Pool1
Status: Success
Time: 2016-08-16 13:34:39,436 MDT
JobId: 1402601677254
Data:
  id:0004fb00000600005fddd9908a3699fe  name:myvm
```

2. Add a virtual disk to the virtual machine's disk configuration:

```
OVM> create virtualdisk name=myvmDisk1 size=10 sparse=yes on Repository
name=iscsi_repos1
Command: create virtualdisk name=myvmDisk1 size=10 sparse=yes on Repository
name=iscsi_repos1
```

```
Status: Success
Time: 2016-08-17 12:26:38,313 MDT
JobId: 1403029594353
Data:
  id:0004fb0000120000609730 1609e986a7.img  name:myvmDisk1
```

3. Add an existing ISO file to the virtual machine's disk configuration:

```
OVM> create VmDiskMapping name="ISO file" slot=1 virtualCd=OL6U5.iso on vm
name=myvm
Command: create VmDiskMapping name="ISO file" slot=1 virtualCd=OL6U5.iso on vm
name=myvm
Status: Success
Time: 2016-08-17 12:41:27,728 MDT
JobId: 1403030486760
Data:
  id:0004fb0000130000b40570ee833883d0  name:ISO file
```

4. Start the virtual machine:

```
OVM> start vm name=myvm
Command: start vm name=myvm
Status: Success
Time: 2016-08-17 12:42:15,483 MDT
JobId: 1403030525543
```

- Access the virtual machine's console from the Oracle VM Manager UI.

When you access the console of the new virtual machine, the welcome screen is displayed. You are ready for installation.

After Installation

After creating a virtual machine from installation media, you can reuse this virtual machine by using cloning. Cloning is discussed later in this topic.

You can also automate the provisioning of virtual machines from your newly installed virtual machine by adding the Oracle VM Guest Additions to your new virtual machine and using the Oracle VM Template configuration facility. The procedure to perform this type of customization is discussed in the lesson titled "Oracle VM Guest Additions."

Creating a Virtual Machine by Importing Virtual Resources

- You can import a template, a virtual appliance, a virtual machine, or virtual disks, by using the Oracle VM management interfaces.
- Using the imported template, virtual appliance, or virtual machine, you create new virtual machines by using cloning.
- If you import individual virtual disks, create a new virtual machine and add the virtual disks to the virtual machine's configuration.

Importing Templates

When importing a template, you can import:

- A single compressed file, containing all of the components of the template, by specifying the URL or FTP location for this file
- All the noncompressed components of the template
  - Example:

    `ftp://mysftpserver/mytemplatedir/system.img`

    `ftp://mysftpserver/mytemplatedir/vm.cfg`
- During the import operation:
  - The template is decompressed (if needed)
  - The configuration file is stored in the `Templates` directory
  - The virtual disks are stored in the `Virtual Disks` directory

A template is a fully preinstalled, preconfigured virtual machine that you can repeatedly clone to create new templates or new virtual machines. Alternatively, you can import a template and provide your own customization to the newly imported template before using it to clone new virtual machines.

A template is composed of a virtual machine configuration file, and one or more virtual disks. When importing the template, you can import:

- A single compressed file by specifying the URL for this file, for example:
  `ftp://myftpserver.example.com/template1.tgz`

-OR-

- All the noncompressed components of the template. If you are using the Oracle VM Manager UI, list the components on separate lines in the Import VM Template window.

  Example:

  `http://myhttpserver.example.com/template/System-sda.img`

  `http://myhttpserver.example.com/template/vm.cfg`

  Note: The order is not relevant.

When the import starts, a directory is created for the new template. The directory name corresponds to the UUID assigned to the new template. If the template is a compressed file, it is decompressed in a temporary directory created in the template directory.

The virtual disks in the template are stored in the `Virtual Disks` directory in the repository selected for the import operation. The configuration file is stored in a subdirectory of the `Templates` directory in the target repository. The name of the subdirectory is the UUID assigned to the new template.

If the configuration file for the template contains an entry for the template simple name, that name is assigned to the template. Otherwise, the name of the imported item (or the name of the first imported item for multiple items) is used for the new template name.

If one or more network interfaces are present in the template's configuration file, the import operation adds the necessary VNICs from the pool of available VNICs. After the import operation completes, you can edit the template and select a network for each of the VNICs in the template. Any virtual machine created by cloning this template will have proper network assignment for each VNIC.

Importing Virtual Appliances

- When importing a virtual appliance, you can import:
  - The virtual appliance packaged into an archive file of type `.ova`
  - The components of this virtual appliance:
    - As the first step, import the virtual disks (in raw format) for the virtual appliance.
    - As the second step, import the `.ovf` file.
- You can view the contents of an virtual appliance by using the `tar` command.
- After importing the virtual appliance, you can create a template from one of the virtual machines in the virtual appliance, or create a virtual machine directly from the virtual appliance.

A virtual appliance contains one or more virtual machine components, and a file of type `.ovf` that describes the components of the virtual machines in the virtual appliance, according to the Open Virtualization Format (OVF) standard. The `.ovf` file can also contain information that describes the interconnectivity between the virtual machines in the virtual appliance.

Normally, an virtual appliance is packaged into an archive file of type `.ova`. This file contains the `.ovf` file and the other components of the virtual appliance. These components include the virtual disks for all the virtual machines in the virtual appliance. The archive file can also include a manifest file (`.mf` file type) that contains the SHA1-1 digest for the files in the virtual appliance.

Use the `tar` command or any other utility that supports the TAR format to view the contents of a virtual appliance. For example, you can obtain a virtual appliance from the Oracle Software Delivery Cloud that contains a virtual machine with the Oracle Secure Global Desktop software installed. In the following example, a utility that supports the TAR format is used to show the contents of the virtual appliance:

Viewing the archive from a Windows platform:

Listing archive: `C:\Users\user1\Desktop\OVM_SGD4U7_x86_64_PVHVM.ova`

```
    Date      Time         Size  Compressed         Name
----------  -----    --------  ----------- ----------------------
2012-08-03 14:54:45       5388       5632   OVM_SGD4U7_x86_64_PVHVM.ovf
2012-08-03 15:04:32        195        512   OVM_SGD4U7_x86_64_PVHVM.mf
2012-08-03 14:41:29  818382088  818382336   System.img
2012-08-03 14:47:39 1120256552 1120257024   Product.img
----------  ------   --------  ------------ ----------------------
                    1938644223 1938645504  4 files, 0 folders
```

You can also create your own virtual appliances by using the Oracle Virtual Assembly Builder. The Oracle Virtual Assembly Builder is discussed later in this lesson.

If your virtual appliance is packaged as a `.ova` file, you can import your virtual appliance directly from the Oracle VM Manager UI. From the Import Virtual Appliance screen, specify the URL to the virtual appliance. The following protocols are supported: HTTP, HTTPS, and FTP.

If your virtual appliance is made up of an `.ovf` file and one or more virtual disk images, you import the virtual disks first by using the Oracle VM Manager, and then import the `.ovf` file using the Import VM Virtual Appliance Wizard.

Note

- When importing the virtual disks for the virtual appliance individually, only raw disk format is accepted. If your virtual appliance contains virtual disks with `.vmdk` or `.vdi` format, you must import your virtual appliance as a `.ova` file or convert the virtual disks to the raw format by using the `qemu-img` command. The `qemu-img` command is available with most Linux distributions.
- You cannot create virtual machines directly from an imported virtual appliance. First, create a template from one of the virtual machines in the virtual appliance. Then use the new template to create a virtual machine, as described in the previous topic.
- When you create a template from a virtual machine in the virtual appliance using the Oracle VM Manager UI or Oracle VM CLI, the disk images of the virtual machine that are in a format other than raw format are converted to raw format. This process can cause the disk images to grow in size.
- If you want to deploy all the virtual machines in a virtual appliance, including the associated networking configuration, at the same time, you must use Enterprise Manager Cloud Control.

Importing Virtual Machines

- When you import a virtual machine, specify the URLs for:
  - Each component of the virtual machine
    -OR-
  - For a single archive file
- After the import operation, the virtual machine appears in a server pool or in the Unassigned Virtual Machines folder.

A virtual machine is composed of a configuration file and one or more virtual disks.

Note that there is no difference between a template and a virtual machine except for the location of the configuration file. The template's configuration file resides in its repository's Templates directory, whereas a virtual machine's configuration file resides in its repository's VirtualMachines directory. The Oracle VM CLI offers the same commands to display, import, or clone these two entities.

Importing a Virtual Machine with the Oracle VM Manager UI

To import the components of a virtual machine, place the files that make up the virtual machine in a location accessible by the Oracle VM Manager using one of these protocols: HTTP, HTTPS, or FTP. Alternatively, you can compress the files into a single archive file (for example, a `.tgz` or `.zip` file).

In the Import Virtual Machine Wizard, specify the URLs for each component of the virtual machine or specify the single archive file.

When you import a template or a virtual appliance, the template or virtual appliance is stored in a repository.

When you import a virtual machine, its components are stored in a repository, and the virtual machine also appears in a server pool or in the Unassigned Virtual Machines folder.

The virtual machine's final location depends on how you launch the Import Virtual Machine operation:

- If you launch the operation from a server pool, the imported virtual machine is deployed to the server pool, and you can store the virtual machine's components in any repository that is presented to at least one Oracle VM server in the server pool.
- If you launch the operation from the Unassigned Virtual Machines folder, the new virtual machine appears in the Unassigned Virtual Machines folder, and is not associated with a server pool. You can select any repository for storing the components of the imported virtual machine.

After the import operation, you can:

- Edit the virtual machine to perform additional configuration on the virtual machine. For example, you can associate an existing network to any VNIC present in the configuration of the virtual machine.
- Create a template from the virtual machine, and use this template to create additional virtual machines through cloning.

Importing Virtual Disks

- If necessary, convert the virtual disks to raw format.
  - Example:

```
# qemu-img convert -f vmdk -O raw disk1.vmdk disk1.img
```

- To import virtual disks:
  - Make the virtual disk images available from a remote location
  - Use the Import Virtual Disk Wizard from the Oracle VM Manager UI or `importVirtualDisk` from the Oracle VM CLI
- After importing the virtual disks, create your virtual machine and specify the imported disks as part of the disk configuration of your virtual machine.

You can create a new virtual machine with existing disk images by following these steps:

- If the virtual disks are in a format other than raw, convert the virtual disks to raw format by using the `qemu-img` command or Oracle VM VirtualBox's `VboxManage clonehd` command.
  Note: Raw format for a virtual disk means that there is no additional info or structure embedded in the virtual disk. It is a bit representation of the data written to the virtual disk.
- Make the virtual disk images available from a remote location, using one of the following protocols: HTTP, HTTPS, and FTP.
- Import the virtual disks into your Oracle VM environment by using the Import Virtual Disk Wizard.
- Launch the "Create Virtual Machine" Wizard, specify parameters for your new virtual machine, and add the virtual disks that you imported previously to your new virtual machine's disk configuration.

If the first virtual disk specified in the list of virtual disks has a bootable OS installed on it, you can start your new virtual machine.

Quiz

Which of the following is a valid way to create virtual machines with bootable guests?

a. By cloning a template containing a bootable virtual disk
b. By installing from installation media
c. By importing an appliance from another virtualization platform
d. By importing a bootable virtual disk and including this disk in a virtual machine
e. All of the above

Answer: e

Creating a Virtual Machine by Using the P2V Utility

- With the P2V utility, you transfer a computer's operating system and applications to an Oracle VM template.
  - The domain type is set to HVM.
- When running the P2V utility, you step through the following stages:
  - A configuration stage
  - A collection stage
  - An import stage
- After completing the transfer:
  - Edit the new template and change settings as needed

  -OR-
  - Create a virtual machine and add the virtual disks created by the P2V utility to your virtual machine's disk configuration

With the P2V conversion utility, you transfer a computer's operating system (Linux or Microsoft Windows) and applications to an Oracle VM hardware virtualized (HVM) guest image by converting the physical disks on the computer to virtual disk images.

The P2V utility is included on the Oracle VM Server for x86 CD.

Note: Consult the Oracle VM Release Notes for availability by release.

To launch the P2V utility, restart and boot the physical computer from the Oracle VM Server for x86 CD.

Note that Oracle VM 3.4 requires a 64-bit processor to boot from the Oracle VM Server for x86 CD.

You can use a P2V kickstart file to automate the creation of hardware virtualized guest images from physical computers.

When running the P2V utility, you proceed with the following stages:

- The configuration stage: For this stage, you provide information to:
  - Provide a network identity to the physical server running the P2V utility
  - Create the list of disks on the physical server to convert to virtual disks
  - Set the parameters for the creation of the new virtual machine

  At the end of this stage, a web (HTTPD) server is started on the physical server.

- The collection stage: For this step, you note the files that you want to import in the new virtual machine.

  You get the list of files by accessing the HTTP server running on the physical server from the browser of any host with network access to this HTTP server.
- The import stage on the Oracle VM Manager: To trigger this step, you start the Import Template Wizard on the Oracle VM Manager UI and specify the files noted in the collection step.

  When you click OK, the import operation begins.

  You can follow the progress of the P2V utility and import operation from the console of the physical machine. You can also follow the progress of the import operation by examining the `/var/log/ovs-agent.log` file on the Oracle VM server specified in the Import Template screen.

After the Import Stage

When the import job completes, you terminate the P2V utility on the console of the physical machine. At this point, a new template appears in the repository from which you triggered the import operation.

The last step is to edit the new template and change settings as needed. For example, you might want to remove the CD/DVD specification from the disk configuration and boot options.

If the configuration file for your new template is not matching your requirements, you can delete the template (without deleting its virtual disks) and create a new template or virtual machine. You then add the virtual disks created as part of the import operation of the P2V utility to the disk configuration of your new template or virtual machine.

Resources

Find out more information about the P2V utility by reading the following topics in the *Oracle VM Administrator's Guide*, Part Number E64083-01 or later:

- Using the P2V Utility
- Using the P2V Utility with a Kickstart File
- P2V Parameters

Creating a Virtual Machine by Creating Your Own Templates and Virtual Appliances

- You can create a template by using one of the following processes:
  - Creating a template package with an existing `vm.cfg` file and existing disk images
  - Creating a template from a virtual machine:
    - Create a virtual machine and perform a minimal install of Oracle Linux.
    - Add needed system and applications packages.
    - Use the Oracle VM Guest Additions to trigger configuration at first boot.
    - Convert the virtual machine to a template.
- You can create a virtual appliance by using the Oracle Virtual Assembly Builder.

Creating a Template Package with Existing `vm.cfg` File and Disk Images

You can package an existing `vm.cfg` file and virtual disks into a compressed `.tar` file and import this template in your Oracle VM environment. After you import the template, edit the template to change settings and add virtual disks or LUNs as needed.

Creating a Base Template Starting with a Minimum Install of Oracle Linux

Create a virtual machine, install Oracle Linux by using the minimal installation option, add the packages needed for your applications, use the Oracle VM Guest Additions and the Oracle VM Template configuration process to set up your virtual machine for first boot configuration, and finally, create a template from the virtual machine. Any virtual machine cloned from this template enters into a configuration wizard when it boots for the first time.

Note: The Oracle VM Guest Additions and the Oracle VM Template configuration feature are discussed in the lesson titled “Oracle VM Guest Additions.”

Using the Oracle Virtual Assembly Builder

You can use the Oracle Virtual Assembly Builder to create virtual appliances by capturing the configuration of your Oracle Fusion Middleware and Oracle Database installations into components called software appliances. You can also modify the logical connections between the software appliances before you group them into software assemblies. You then deploy your virtual appliance into your Oracle VM environment.

Creating a Virtual Machine by Migrating Virtual Machines from Other Platforms

- Many virtualization products support the packaging of their virtual machines into virtual appliances.
- You can import a virtual appliance if it is packaged as a `.ova` file.
- After creating or obtaining the `.ova` file:
  - Import the virtual appliance
  - Create a template from one of the virtual machines in the virtual appliance and clone this template to create a virtual machine or another template
  - Create a virtual machine directly from the virtual appliance

You learned previously in this lesson that you can import a virtual appliance in your Oracle VM environment. You can import a virtual appliance:

- As a collection of files, including a mandatory `.ovf` type file
- As a `.ova` archive file, which is a package of one or more virtual machines adhering to the OVF standard

Many virtualization products support the packaging of virtual machines produced on their virtualization platform into virtual appliances. You can import such a virtual appliance if it is packaged as a `.ova` file adhering to the OVF standard, and if the virtual disk images in the package are raw disk images, or formatted as VMDK, VHD, or VDI disk images. This means that you can import virtual machines from the VMware, Microsoft Hyper-V, and Oracle VM VirtualBox virtualization platforms.

For example, you can export a virtual machine from Oracle VM VirtualBox using the Export Appliance Wizard. You can export one or more virtual machines in a single export operation.

After exporting the virtual machine as an appliance, you import the appliance as an virtual appliance. From there, you create a template from one of the virtual machines in the virtual appliance, and clone this template to create a virtual machine. Or, create a virtual machine directly from the virtual appliance. Consult the online documentation on how to perform this operation.

In most cases, the virtual appliance produces templates with the domain type set to HVM. You can install the release-appropriate paravirtualized drivers into the virtual machine created from the HVM-type template to convert it to a PVHVM type virtual machine.

Creating a Virtual Machine by Using Cloning

- When you clone a virtual machine or a template, you create a new virtual machine (or template) based on the source object.
- Cloning a virtual machine with virtual disks residing in an OCFS2 repository uses a reflink operation, which is also called fast copy operation.
- A new clone obtains a new object UUID and a new MAC address for each cloned VNIC.

What Happens During Cloning?

- A new UUID is assigned to the cloned object, which is a virtual machine or a template.
- For each network interface or VNIC present in the source object, a new VNIC is created in the clone. If the VNIC is assigned to a network, the VNIC in the clone is also assigned to this network. If the network specified for the source network interface does not exist in your Oracle VM environment, a new network is created as part of this assignment.
- Each virtual or physical disk to be cloned is copied or cloned to a new virtual or physical disk for the cloned object.

There are requirements for the successful cloning of virtual and physical disks. These requirements are discussed in the following two slides, titled “Simple Cloning” and “Advanced Cloning.”

Cloning and Disk Space

When you clone a virtual machine residing in an OCFS2-type repository, the cloning operation is very fast because it uses the reflink feature of OCFS2 to clone the virtual disks. The reflink feature uses a copy-on-write technique. These virtual disks occupy very little space initially, but it is possible to run out of space as more disk space is needed by the clones.

Simple Cloning

- If you perform a simple cloning operation, the clone is created exactly like the source object.
- For thin cloning to take place with simple cloning, all components in the source object must reside in:
  - OCFS2 repositories
  - Vendor-managed storage that supports thin provisioning

With simple cloning, each resource for the source object is cloned into the same location as the source object. For example, if the configuration file (`vm.cfg`) for the source object resides in an NFS repository, the configuration file for the clone is created in that same repository. Similarly, each disk is cloned into the same location as the source disk.

Simple Cloning and Disk Resources

- If a source disk is a virtual disk in a repository, a new virtual disk is created in the same repository, using thin cloning for an OCFS2-type repository, or a copy operation for a virtual disk residing in an NFS-type repository.
- If the source disk is a physical disk (LUN) in a storage array managed by a vendor-specific plug-in, and the plug-in supports it, a new LUN is created in the storage array.
- If the source disk is a physical disk (LUN) in a generic storage array, the cloning operation fails because a generic storage array does not support thin cloning.

Advanced Cloning

- For advanced cloning, you can create different storage and network mappings.
- All advanced cloning operations require the use of a clone customizer.
- You can reuse a clone customizer if all of the storage targets for the disk mappings support thin provisioning.
- If the source object that you want to clone has shared disks assigned, do not map these shared disks in the clone customizer.

Advanced Cloning Facts

- If the source virtual disk is in a repository, you can direct the cloning of the virtual disk to the same repository or to another repository, or to an existing physical disk (LUN) in any storage array.
- If the source virtual disk is a physical disk (LUN) in a storage array managed by a vendor-specific plug-in, and the plug-in supports it, you can choose to have the disk cloned in the same storage array, using thin provisioning, or you can choose to have the disk copied to another existing LUN in any storage array.
- If the virtual disk is a physical disk (LUN) in a generic storage array, you can choose to have the disk cloned to any other existing disk, in any storage array.

All advanced cloning operations require the use of a clone customizer. If a clone customizer references an existing LUN in any storage array, you can use the clone customizer only once.

Cloning Virtual Machines with Shared Disks

If your virtual machine has access to shared disks, the shared disks are cloned during the cloning process. If you do not want this to happen, create a clone customizer for the virtual machine or template that you want to clone and remove the storage mappings for the shared disks. After the cloning operation, assign the shared disks to the new clones.

Virtual Machine Operations

- In the first section of this lesson, you learned about the various ways to create virtual machines.
- The next section of this lesson is dedicated to virtual machine operations and includes the following topics:
  - Migrating virtual machines, including live migration
  - Protecting virtual machine resources
  - Making changes to a running virtual machine

Migrating Virtual Machines

- Migration is the action of moving a virtual machine from one Oracle VM server to another.
- If the virtual machine is running, the move operation is called live migration.
- Live migration can be triggered:
  - From any of the Oracle VM management interfaces
  - By a DRS or DPM policy action
  - When an Oracle VM server enters maintenance mode

What is migration? Migration is the action of moving a virtual machine from one Oracle VM server to another. If the virtual machine is running, the move operation is called live migration; otherwise, the move is called cold migration.

Note: If an Oracle VM server fails, virtual machines running on this server that have high availability enabled restart on an available Oracle VM server in the server pool. This is not a live migration operation. During live migration, a virtual machine keeps running during the entire operation.

You can trigger a live migration operation from the Oracle VM Manager UI, the Oracle VM CLI, or the Oracle VM WS-API. Other processes within your Oracle VM environment can also trigger a live migration operation:

- If you put an Oracle VM server in maintenance mode, Oracle VM attempts to migrate the virtual machines running on the Oracle VM server entering maintenance mode.

  If there are not enough resources on the other Oracle VM servers in the server pool, the live migration does not take place.
- Distributed Resource Scheduler (DRS) or Dynamic Power Management (DPM) operations: These two Oracle VM server pool policies attempt to balance or consolidate resources on the Oracle VM servers within a single server pool.
- A DRS policy uses live migration to move running virtual machines to other Oracle VM servers in a server pool if the Oracle VM server exceeds a specified CPU or network threshold for a specified period of time.
- Similarly, DPM uses live migration to move running virtual machines to other Oracle VM servers in a server pool to consolidate running virtual machines on fewer Oracle VM servers.

Live Migration Facts

- A virtual machine can only live migrate between Oracle VM servers belonging to the same server processor compatibility group.
- Live migration of virtual machines using hard partitioning is not permitted under the terms of the hard partitioning license.
- You can use live migration for virtual machines in an Oracle RAC deployment.

Live Migration and Processor Compatibility Groups

For live migration to succeed, the source and target Oracle VM servers used during the migration must be in the same server pool and have the same processor type and model number. When you discover an Oracle VM server, the server is added to an existing processor compatibility group if it exists, or a new compatibility group is created automatically if the Oracle VM server is the first discovered server of its processor type and model number.

You can add granularity for live migration to your Oracle VM server pools by creating your own processor compatibility group, which might be a subset of the Oracle VM servers in an existing processor compatibility group. When live migration occurs for a virtual machine, it migrates to a server that is part of the custom compatibility group if the source and target Oracle VM servers used during the live migration belong to that same group.

Live Migration and Hard Partitioning

Hard partitioning is a technique employed to conform to Oracle licensing requirement, for Oracle per-processor licensed products. With Oracle VM, the technique consists of binding vCPUs to physical CPUs (threads or cores). The technique is often referred to as CPU pinning. The bound vCPUs cannot be scheduled on physical CPUs other than those specified in the binding operation.

Live migration of virtual machines using hard partitioning is not permitted under the terms of the hard partitioning license. You must not use DRS and DPM for server pools containing virtual machines that have been modified to implement hard partitioning. High availability failover is permitted because you are granted a period of time to recover from the failure and return your virtual machines to the initial hard partitioning rules. If you have questions about hard partitioning and the limits it imposes in your Oracle VM environment, please contact My Oracle Support.

Live Migration and Clustering Software

You can use live migration for virtual machines in an Oracle RAC deployment. During live migration, there is a suspend period during which the last memory dirty pages are copied to the migrated virtual machine on the target Oracle VM server in the server pool. For highly active applications, the suspend time can be long enough to reach the threshold set for the Oracle Clusterware heartbeat. This situation can cause node fencing at the Oracle Clusterware level.

The recommendations in the document titled "Oracle Real Application Clusters in Oracle VM Environments" can help you minimize potential failures during live migration of RAC nodes.

Protecting Virtual Machine Resources with the Inbound Migration Lock

- You can use the inbound migration lock feature to protect the resources allocated to one or more virtual machines.
  - You activate this feature at the Oracle VM server level with the edit function.
- This feature blocks actions that can reduce the resources on the Oracle VM server where the virtual machines are running, such as:
  - The creation of new virtual machines on the Oracle VM server
  - The migration of virtual machines to that Oracle VM server
- Other mechanisms that trigger migration actions, such as DRS and DPM, comply with the inbound migration lock feature.

When a new virtual machine is started on an Oracle VM server, or a virtual machine is migrated to an Oracle VM server, server resources are allocated to this new virtual machine, possibly impacting the operation of virtual machines already running on the server.

At the Oracle VM server level, you can block actions that further deplete server resources by editing the Oracle VM server and activating the inbound migration lock feature.

With the Oracle VM CLI, use the `edit server` command to perform this change:

```
OVM> edit server name=ovm1 inboundMigrationLocked=Yes
Command: edit server name=ovm1 inboundMigrationLocked=Yes
Status: Success
Time: 2014-06-19 16:40:09,499 MDT
JobId: 1403217609061
OVM>
```

This feature can be activated in addition to the other protection mechanisms available with Oracle VM, such as high availability, DRS and DPM, and anti-affinity groups. In case of a conflict, the inbound migration lock feature overrides the other migration attempts. The inbound migration lock feature does not prevent virtual machines running on the protected Oracle VM server from being migrated to another Oracle VM server.

Making Changes to a Virtual Machine

- You can make the following changes to a running virtual machine:
  - Add network devices.
  - Add or remove virtual and physical disks.
  - Add or remove virtual CPU resources.
  - Increase and decrease virtual memory.
  - Increase the size of a virtual disk or LUN.
- The ability to make the change depends on the virtual machine domain type and the guest OS.

Adding Network Devices

You can add VNICs to a running virtual machine. The VNIC is available immediately in the virtual machine. You cannot remove VNICs from a running virtual machine using the Oracle VM management interfaces.

Adding or Removing Virtual or Physical Disks

You can add virtual and physical disks. They can then be used by the OS in the virtual machine. You can also remove virtual and physical disks but this operation can cause problems if the removal is not clean.

Increasing or Decreasing Virtual Memory and Number of Virtual CPUs (vCPUs)

For PVM guests: You can increase the virtual memory of a virtual machine, up to the maximum memory setting of your virtual machine. The same applies to vCPUs. You can increase the number of vCPUs up to the maximum number of processors set for your virtual machine.

For HVM guests: The maximum memory size and the memory size must always be the same. However, you can set the maximum number of processors in your virtual machine to a value higher than the number of processors when the virtual machine is not running, and you can change the current number of processors up to the maximum number while the virtual machine is running.

Note: The ability to increase or decrease virtual memory and virtual CPU resources depends greatly on the support provided by the OS running in your virtual machines. For more information about support for increasing or decreasing virtual resources, consult the release notes for the Oracle Linux level or other operating systems running in your virtual machines.

Increasing the Size of a Virtual or Physical Disk

You can increase the size of a virtual disk or physical disk in a virtual machine, but the result is different for the two types of disks:

- For a virtual disk, the new size is not reflected in the virtual machine. If you reboot your virtual machine, the new size is accessible.

  Note: You can use live migration to force the guest OS to recognize the new disk size.

- For a physical disk, you must first refresh the SAN server if the new resized physical disk resides in the storage array managed with a generic plug-in.

  In your virtual machine, you must inform the guest kernel of the new device size using the `partprobe` or similar command. The `partprobe` command is included in the `parted` package.

To avoid the limitations inherent to resizing a virtual or physical disk, consider using the Logical Volume Manager (LVM) in your virtual machine. With LVM, you can add new virtual disks on the fly, add them as Physical Volumes (PVs), and use the added space to extend your existing logical volumes (LVs).

Virtual Machine Life Cycle

- Virtual machines in your Oracle VM environment step through stages that are similar to those experienced by physical hosts.
- You can design a test/production life cycle for your virtual machines by dedicating servers for the test cycle, and move these virtual machines to more powerful servers when they enter the production cycle.
- In the lesson titled “Operations,” you explore ways of reconfiguring your environment to respond to the changing needs of your environment.

Virtual machines in your Oracle VM environment step through stages that are similar to those experienced by physical hosts. The business applications running on those virtual machines might need additional network bandwidth and disk space, and more resources such as CPUs and memory. They also need access to services to provide ongoing release and patch management, backup, and restore. Many of these services are provided in a way that is very similar to those provided to physical hosts. For example, if your virtual machines are running Oracle Linux, you can provide them with access to the Unbreakable Linux Network or ULN.

Virtual machines, however, offer increased flexibility and mobility:

- You can provision virtual machines dynamically using templates and virtual appliances.
- You can migrate virtual machines between Oracle VM servers in the same server pool.
- You can use hot cloning to capture an image of your virtual machine for backup purposes.
- You can design a test or product life cycle for your virtual machines by dedicating servers for the test cycle, and migrating machines to more powerful servers when they enter the production cycle.

Enterprise Manager Cloud Control can provide many lifecycle management services to your Oracle VM environment. Oracle VM also offers many capabilities to grow your environment.

In the lesson titled “Operations,” you explore ways of growing and reconfiguring your Oracle VM environment to respond to the changing needs of your business applications.

Quiz

Live migration of a virtual machine can be triggered:

a. When an Oracle VM server enters maintenance mode
b. By a DRS or DPM policy
c. By a migration request from one of the Oracle VM Manager management interfaces
d. By a request from within the virtual machine
e. All of the above

Answer: a, b, c

Summary

In this lesson, you should have learned how to:

- List and compare virtualization types
- Identify and illustrate the various methods for creating virtual machines
- Manage virtual machine components such as disks, hot plug-in operations for disks and network, and dynamic CPU and memory changes
- Clone, migrate, and modify virtual machines

Practice Overview

In this practice, you:

- Create virtual machines by using cloning and examine the new clones
- Examine templates and virtual appliances
- Use P2V to create a new template and create a virtual machine from the template
- Add a virtual disk to a running virtual machine and perform a resize operation on the new disk

4

Oracle VM Guest Additions

Objectives

After completing this lesson, you should be able to:

- Explain the benefits of the Oracle VM Guest Additions
- Install the Oracle VM Guest Additions in your virtual machines
- Exercise the messaging feature between the Oracle VM Manager and virtual machines
- Enable and disable Oracle VM Template configuration scripts for specific targets
- Configure a virtual machine with the first-boot wizard

Oracle VM Guest Additions

- The Oracle VM Guest Additions is a set of packages that you install in a virtual machine running the Oracle Linux 5, 6, or 7 operating system.
- The Guest Additions package provides the following enhancements:
  - Bidirectional communication between the Oracle VM Manager and the operating system running in the virtual machine
  - Guest IP information visible from the Oracle VM Manager
  - Ability to configure or reconfigure your virtual machines during the first boot

ORACLE

The Oracle VM Guest Additions is a powerful tool that provides fine-grained control over the configuration and behavior of components running in your virtual machines.

In addition to establishing a bidirectional communication between the Oracle VM Manager and the guest operating system in the virtual machine, the tool includes a series of configuration templates that allow you to configure the virtual machine during its first boot. The virtual machine enters a configuration wizard to obtain information about networking, firewall, date and time, and any other configuration items that you have enabled with the Oracle VM Guest Additions. By taking advantage of the communication channel between the Oracle VM Manager and the virtual machine, you can provide answers to this wizard automatically, eliminating the need to respond to the questions from the virtual machine's console.

This type of setup, which triggers an initial configuration in a virtual machine, is most useful when performed in a virtual machine that is then converted to a template. Any virtual machine cloned from such a template enters the configuration wizard when it boots for the first time.

Installing the Oracle VM Guest Additions

- Create an Oracle Linux virtual machine running Oracle Linux 5, 6, or 7, with the Oracle Unbreakable Enterprise Kernel (UEK).
- Add the following packages to your virtual machine:

```
ovm-template-config-datetime
libovmapi
libovmapi-devel
ovmd
python-simplejson
xenstoreprovider
ovm-template-config
ovm-template-config-authentication
ovm-template-config-firewall
ovm-template-config-network
ovm-template-config-selinux
ovm-template-config-ssh
ovm-template-config-system
ovm-template-config-user
```


You download these packages from the following Unbreakable Linux Network (ULN) channels:

- Oracle Linux 7 Add ons (x86_65)
- 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 to the ULN channels, the packages are available from a repository on the Oracle Linux yum server.

If you plan to use ULN or the Oracle Linux yum server, make sure that your virtual machine has public Internet access, or that you have configured a private Yum repository.

Install the latest version of all the packages listed in the slide.

Note: Current Oracle Linux templates and virtual appliances that are available from the Oracle Software Delivery Cloud have the Oracle VM Guest Additions already installed.

Using the Oracle VM Guest Additions Daemon and Template-Driven Configuration

- When you install the Oracle VM Guest Additions in your virtual machine, the `ovmd` command is installed in `/usr/sbin`. This command runs as a daemon (service) and as a command.
- With the `ovmd` command, you can:
  - Send messages to the Oracle VM Manager or receive messages from the Oracle VM Manager
  - Run a script (usually a configuration script)
  - Enable or disable initial configuration at the next boot

In this section, you learn how to send and receive messages with the `ovmd` command. In the next section, you learn how to prepare your virtual machine to enter into a configuration dialog when the virtual machine boots for the first time, or any subsequent boot where you want the virtual machine to be reconfigured.

Note: When the `ovmd` service is running in a virtual machine, the virtual machine's IP information is visible from the Oracle VM Manager UI.

Sending and Receiving Messages

With the `ovmd` command in your virtual machine, you can send messages to and receive messages from the Oracle VM Manager.

- Sending a message from the Oracle VM Manager by using the Oracle VM CLI:

```
OVM> sendVmMessage vm name=pvm7 key=shape message=circle log=yes
Command: sendVmMessage vm name=pvm7 key=color message=red log=yes
Status: Success
Time: 2016-09-10 16:56:59,652 MDT
JobId: 1410389818948
OVM>
```

- Retrieving a message from the virtual machine by using the `ovmd` command:

```
# ovmd -l
{"color":"red"}
```

The messaging facility is an important component of the automatic configuration mechanism provided by the Oracle VM Guest Additions.

You can send one or more messages from the Oracle VM Manager by using the Oracle VM Manager UI, the Oracle VM CLI, the Oracle VM WS-API, or the `ovm_vmmessage` utility. The `ovm_vmmessage` utility is part of the Oracle VM Utilities. The example in the slide uses the Oracle VM CLI.

Each message sent (and received) contains one pair of `key:value`, for example, `color=red`.

You use the `ovmd -l` command from the virtual machine to retrieve the message.

Sending messages to a virtual machine from the Oracle VM Manager is the type of communication that takes place when setting up a virtual machine for automatic configuration during the first boot. During the first boot, the virtual machine enters the configuration wizard and waits for the input needed for its configuration process.

In the following slides, you learn how to set up the automatic configuration to take place during the first boot of a virtual machine.

Configuring a Virtual Machine by Using the Oracle VM Template Configuration Scripts

The Oracle VM Template configuration script named `ovm-template-config` is the master script used to set up automatic configuration in the virtual machine:

- This master script works in conjunction with a set of modular configuration scripts (`authentication`, `datetime`, `firewall`, `network`, `selinux`, `ssh`, `system`, and `user`).
- You can enable or disable these scripts for a particular target.
- The two widely used targets are `configure` and `cleanup`.
- The enabled scripts run for the selected target.

All of the components needed for setting up automatic configuration of your virtual machines are installed and created when you install the Oracle VM Guest Additions packages.

Oracle VM Template Configuration Scripts

The Oracle VM Template configuration script named `ovm-template-config` is the master script used to set up automatic configuration.

The master script works in conjunction with a set of modular configuration scripts, with one script module for each major configuration area in the OS of your virtual machine. These scripts are named `authentication`, `datetime`, `firewall`, `network`, `selinux`, `ssh`, `system`, and `user`. The master script also works with `ovmd` running in the virtual machine to capture configuration parameters that have been sent to the guest using the Oracle VM messaging facility.

Targets

The automatic configuration setup also includes targets, which are equivalent to the run levels in the System V type UNIX implementation. With the automatic configuration provided by the Oracle VM Guest Additions, the run levels are replaced with targets, which specify when the configuration modules are run.

The two main targets are:

- `configure`

  The `configure` target gets activated when a virtual machine boots for the first time or when it is invoked directly.

- `cleanup`

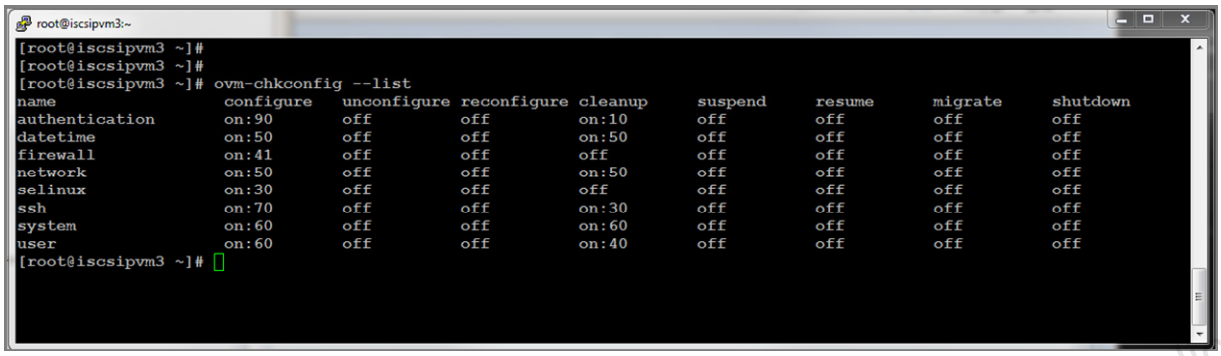
  You activate the `cleanup` target manually to initiate a cleanup in the virtual machine that you want to restore to its original state.

Note: There are other targets available but they are not used at this time.

You can enable or disable the configuration scripts to run for a particular target. To help you with this task, the Oracle VM Guest Additions provides a command called `ovm-chkconfig`, which works like the `chkconfig` Linux command.

Initial Settings for the Configuration Scripts and Targets

- After you install the Guest Additions packages, the initial settings for the configuration modules and their associated targets are shown in the following image:



- All of the configuration modules (scripts) are set to execute for the `configure` target.
- For the `cleanup` target, the `firewall` and the `selinux` configuration scripts are disabled.

The `ovm-chkconfig --list` command output in the slide shows:

- The targets, which are listed in the top row labeled `name`
- The configuration modules, which are listed in the first column. Each module is associated with a script with the same name.

Note that only the `configure` and `cleanup` targets have enabled modules. The other targets are not implemented at this time. For the `configuration` target, all of the configuration modules are set to execute. For the `cleanup` target, the `firewall` and the `selinux` configuration are not reset.

You can change the list of configuration scripts that run for any target. For example, run this command to disable the execution of the `datetime` configuration script for the `configure` target:

```
# ovm-chkconfig -target=configure datetime off
```

All of the configuration scripts are stored in the `/etc/template.d/scripts` directory in the virtual machine where you install the Oracle VM Guest Additions packages.

You can query these scripts to find out what input is expected by the script.

Using Enumerate with Configuration Scripts

You can list the `key:value` pairs expected by a configuration script by using the `ovm-template-config` master script with the `enumerate` option and by supplying a script name.

Example:

```
[root@myvm scripts]# ovm-template-config --human-readable \        --enumerate
configure --script firewall
  [('41',
 'firewall',
 [{u'description': u'Whether to enable network firewall: True            or False.',
   u'hidden': True,
   u'key': u'com.oracle.linux.network.firewall'}])]
[root@myvm scripts]#
```


The example in the slide shows that the `firewall` script, invoked for the `configure` target, expects a key set to `com.oracle.linux.network.firewall` and a value set to `True` or `False`. When the virtual machine that you want to configure on first boot executes the `firewall` script, it expects this `key:value` pair as input to the script. You use the messaging facility from the Oracle VM Manager to send the necessary `key:value` pairs to your virtual machine for all configuration scripts that are set to execute at first boot.

The steps to implement this first boot configuration are discussed next.

First-Boot Configuration: Before the First Boot

- Create and configure a virtual machine to serve as a template.
- Use the `ovm-chkconfig` command to enable/disable configuration scripts for the `configure` target.
- Prepare a list of all the `key:value` pairs required by each enabled configuration script.
- Unconfigure the OS in your virtual machine by using the `ovmd` command with the `cleanup` target:

  ```
  # ovmd -s cleanup
  ```

- Set up the virtual machine to configure at boot time:

  ```
  # service ovmd enable-initial-config
  ```

- Shut down the virtual machine and convert to template.

List of Steps to Perform a First-Boot Configuration in Your Virtual Machine

Before Rebooting the Virtual Machine

- Make a note of all the `key:value` pairs required by each configuration module.

  Use the `ovm-template-config --human-readable --enumerate configure -script <script name>` command to display these `key:value` pairs.

  Look at the example for this command in the previous slide.
- Use the `ovm-chkconfig` command to enable or disable configuration modules for the `configure` target. By default, all the configuration modules, `firewall`, `network`, `selinux`, `ssh`, `system`, and `user`, are enabled. If enabled, the script for these modules is executed during the first boot configuration.

Note: Before disabling or enabling a module for a particular target, make sure that the associated configuration script supports this operation. For example, the `firewall` script supports only the `configure` target. It does not support the `cleanup` target.

- Unconfigure the OS in your virtual machine by using the `ovmd` command with the `cleanup` target:

  For example:

  ```
  # ovmd -s cleanup
  ```

  This command runs the `cleanup` scripts immediately.

Note: If the `network` module is enabled for the `cleanup` target and you issue this command from a remote session, you lose access to your virtual machine. To avoid this situation, execute all commands from the console of your virtual machine.

- Set up the virtual machine to configure at boot time:

  ```
  # service ovmd enable-initial-config
  ```

  This command directs the `ovmd` service to enter the first-boot wizard at the next boot.
- Shut down the virtual machine.

Testing Your Automated Configuration Setup

The previous steps assume that you are working with a virtual machine that you intend to convert into a template for cloning purposes. However, for testing purposes, you can trigger the execution of the scripts associated with the `configure` target immediately, from the command line, rather than wait for a reboot.

The steps for this type of testing are similar to those described previously:

- Send the `key:value` pairs required for the scripts that have been enabled to execute in your virtual machine for the chosen target. Use the Oracle VM CLI or the Oracle VM Manager UI for this step.
- List these `key:value` pairs from your virtual machine to verify that they are present in the queue:

```
# 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":"mysecret"}
#
```

- Execute the `ovmd` command with the `-s` (or `--script`) option and the `configure` target:

```
# ovmd -s configure
```

Your virtual machine starts running the scripts and uses the `key:value` pairs in the queue as input to the scripts.

First-Boot Configuration: First-Boot Processing

- Clone a virtual machine from the template that is set to run the `configure` scripts at boot time.
- Start the virtual machine.
- Using the Oracle VM messaging facility, send the required `key:value` pairs to your virtual machine.
  - Supply the `key:value` pair for the authentication script last.
- When the configuration scripts have completed, the virtual machine continues its booting process.
  - After the boot process, log in to your virtual machine and verify its configuration.

List of Steps to Perform a First-Boot Configuration in Your Virtual Machine

Booting the Virtual Machine

- Clone the template into a virtual machine and start the virtual machine.
- The virtual machine runs the `ovmd -s configure` command and waits for `key:value` pairs as input to the configuration scripts.
- Using the messaging facility from any of the Oracle VM management interfaces, send the `key:value` pairs needed by each configuration module that is active for the configure target.

  The virtual machine must be running to receive messages.
- `ovmd` waits until all the required parameters are received. The only required parameter is the `root` password to configure the system's `root` password. After sending all the `key:value` pairs needed by the configuration scripts, send the `key:value` pair for the `root` password.

  The required key is: `'com.oracle.linux.root-password'`.

  Note: When you specify `key:value` pairs from the Oracle VM Manager UI, do not use quotation marks for the key.

- After receiving the (only) required parameter, the configuration scripts start executing. Any optional parameter sent after the last required parameter is ignored. All `key:value` pairs that you did not supply and that are needed by the configuration scripts will take default values.

When the configuration scripts have completed, the virtual machine continues its booting process. After the boot process, you can log in to your virtual machine using the `root` user and the password that you provided as part of the configuration process.

First-Boot Processing Using Console Input

When you set a virtual machine to enter the configuration wizard at first boot, the configuration wizard displays messages on the console. You can supply the value for each key from the console, and after you enter the password for `root`, the configuration scripts start executing.

Resources

You can find more information about provisioning virtual machines with the help of the Oracle VM Guest Additions by consulting the white paper titled “Oracle VM Templates - Automated Virtual Machine Provisioning” at http://www.oracle.com/us/technologies/virtualization/ovm-templates-wp-2027191.pdf.

Quiz

With the Oracle VM Guest Additions installed in your virtual machines, you can:

a. Send key/value pairs to your virtual machine
b. Receive key/value pairs from your virtual machine
c. Configure your virtual machine with the first-boot wizard
d. Find the guest IP information from a virtual machine that runs the Oracle VM Guest Additions `ovmd` service
e. All of the above

ORACLE

Answer: e

Summary

In this lesson, you should have learned how to:

- Explain the benefits of the Oracle VM Guest Additions
- Install the Oracle VM Guest Additions in your virtual machines
- Exercise the messaging feature between the Oracle VM Manager and virtual machines
- Enable and disable Oracle VM Template configuration scripts for specific targets
- Configure a virtual machine with the first-boot wizard

Practice Overview

In this practice, you:

- Install the Oracle VM Guest Additions in a virtual machine
- Exercise the messaging feature between the Oracle VM Manager and a virtual machine
- Use the Oracle VM Template configuration scripts for configuring a virtual machine
- Prepare a virtual machine for first-boot configuration
- Test your first-boot setup with the Oracle VM Manager UI and the Oracle VM CLI

5

Operations

Objectives

After completing this lesson, you should be able to:

- Discover, rediscover, add, and remove Oracle VM servers from your environment
- Describe server pool operations
- Rediscover repositories, move repositories between server pools, and reclaim repositories
- Notify your Oracle VM Manager about changes to the storage layer in your Oracle VM environment

Lesson Overview

- In the first section of this lesson, you learn how to manage Oracle VM server resources in your Oracle VM environment, whether the servers belong to a server pool or not.
- In the next section, you learn how to manage your repositories.
- The last topic describes how to refresh the storage layer in your environment.

Operations on Oracle VM servers include:

- Discovering and rediscovering Oracle VM servers
- Adding Oracle VM servers to server pools, or removing them from server pools
- Deleting an Oracle VM server
- Releasing ownership of an Oracle VM server
- Reinstalling the software in the Oracle VM server

Operations on repositories include:

- Moving repositories to other server pools
- Backing up repositories
- Removing repositories from your environment

Discovering Oracle VM Servers

- After installation, the Oracle VM Agent on the Oracle VM server listens for discovery events from Oracle VM Manager.
- During discovery, the Oracle VM Manager contacts the Oracle VM server by using the IP address or host name specified in the discovery request.
  - The Oracle VM Manager also specifies a password, which must match the password that you assigned to the Oracle VM server during installation.
- The Oracle VM Manager performs a series of discovery operations on the Oracle VM server.

Discovery operations are used to inform Oracle VM Manager that new objects have been added to the physical environment. These changes can include adding a new Oracle VM server to the environment, or changes to a particular Oracle VM server (for example, the installation of a new Storage Connect plug-in on that server).

Discovering Oracle VM Servers

You add Oracle VM servers to your environment by discovering them. For this operation, you provide the IP address or host name of one or more Oracle VM servers, on which you previously installed the Oracle VM Server for x86 software. You also provide the OVS Agent password for the server. You specified this password when you installed the Oracle VM Server for x86 software on the Oracle VM server host.

You do not have to specify the same Oracle VM Agent password for all Oracle VM servers that are bound for the same server pool. If the Oracle VM servers to be discovered have different passwords, you have to discover them separately when you are using the Oracle VM Manager UI for the discovery process.

During the discovery process, the Oracle VM Manager contacts the new Oracle VM server, authenticates the server, and performs a series of discovery steps. Discovery information is recorded in the Oracle VM Manager database.

Discovery Steps

- The Oracle VM Manager performs the following types of discovery operations in the Oracle VM server during the discovery process:
  - Basic
  - Hardware
  - Networking
  - Plug-ins
  - Attached LUNs
  - Server database
  - File system mounts
  - Cluster information
- If the Oracle VM server belongs to another Oracle VM Manager, only a partial discovery is performed.

ORACLE

The Oracle VM Manager records the following information for each phase of the discovery process:

- **Basic Discovery:** Oracle VM server UUID, architecture (SPARC/x86), hypervisor type (LDOMs/XEN), host name, software version, and ownership information, which includes the owning Oracle VM Manager and basic server pool information
- **Hardware Discovery:** System family and model, CPU information, memory information, Fibre Channel (FC) HBAs, FC initiators and targets, iSCSI initiators, and targets
- **Network Discovery**: NICs, IP addresses, VLAN interfaces, bonds, and Xen bridges
- **Plug-in Discovery:** All plug-ins installed on the Oracle VM server. This includes the plug-ins that are installed as part of the installation process, and the plug-ins installed separately. For example, the Storage Connect Oracle ZFS Storage Appliance plug-in is installed separately.
- **LUNs Discovery:** All attached physical disks, including local disks, and iSCSI and FC LUNs
- **Oracle VM Server DB Discovery:** Information stored in the Oracle VM server's database, located in `/etc/ovs-agent/db`. This database contains information about server pools and the Oracle VM servers in these server pools, as well as repository information.

- **File system mounts:** The mount points on the Oracle VM server for the pool server file system and all the repositories are stored in the `db` directory, rather than rely on the `/etc/fstab` file.
- **Cluster Discovery:** Cluster information is read and stored in the `db` directory in the server pool file system. This includes information about running virtual machines with high availability enabled.

Discovering Oracle VM Servers from a Previous Installation of the Oracle VM Manager

If you perform a reinstallation of your Oracle VM Manager software by using the UUID that was used in the previous installation, you can discover the Oracle VM servers that belonged to that previous installation. In this case, the discovery process records information from that earlier configuration. You can update this information from the Oracle VM Manager UI or the Oracle VM CLI.

Discovering Oracle VM Servers that Belong to Another Oracle VM Manager

If you discover an Oracle VM server that belongs to another Oracle VM Manager, only a partial discovery is performed. The Oracle VM Manager does not track the status of such Oracle VM servers.

Rediscovering Oracle VM Servers

- You rediscover Oracle VM servers to update their configuration information in the Oracle VM Manager database.
- When you restart the Oracle VM Manager, only a basic rediscovery is performed.
  - To trigger a full rediscovery, use the server rediscovery function from the Oracle VM Manager UI.
- If the certificate-based authentication between the Oracle VM Manager and the Oracle VM server is no longer valid, use the server discovery action.
  - You must specify the Oracle OVS Agent password as part of the discovery process.

You use the Rediscover Server function for your Oracle VM servers to notify Oracle VM of changes to your servers. These changes can include:

- Newly installed Storage Connect plug-ins. For example, an updated version of the plug-in for the Oracle ZFS Storage Appliance.
- Newly attached FC LUNs managed with the generic plug-in.

The rediscovery process is the basic mechanism for synchronizing the state of an Oracle VM server with the information in the Oracle VM Manager database. The rediscovery operation is essentially the same as the initial discovery process. If the information is new, that information is recorded or updated in the Oracle VM Manager database.

Note: If you are using the Oracle VM CLI, execute the `refresh server` command to rediscover your Oracle VM server.

Rediscovery During Restart of Oracle VM Manager

If you want to perform a more exhaustive discovery, use the rediscovery function.

The server rediscovery operation assumes that the certificate-based authentication between the Oracle VM Manager and the Oracle VM server is still in place. If this authentication is no longer valid, use the discovery operation instead. The discovery process will restore the certificate-based authentication as well as perform all the initial discovery steps.

Operations on Oracle VM Servers

After discovering an Oracle VM server, you can:

- Configure file-based (NFS) or block-based (iSCSI or FC) storage
- Create networks and add ports or bonds from this Oracle VM server to the network configuration
- Present the Oracle VM server to NFS-type repositories
- Rediscover the Oracle VM server
- Add the Oracle VM server to a server pool
- Release ownership for the Oracle VM server
- Delete the Oracle VM server

After being discovered, the Oracle VM server is owned by the Oracle VM Manager and appears in the Unassigned Servers folder.

At this point, you can add this Oracle VM server to your storage and network configuration. This includes presenting the Oracle VM server to NFS repositories if they exist, and adding the Oracle VM server to the required storage configuration so that you can add this server to a clustered server pool. The Oracle VM server must be able to access the pool file system associated with the clustered server pool and, if the heartbeat network for the clustered pool is on a network other than the management network, the Oracle VM server must participate in this heartbeat network.

Other operations include creating bonds on Oracle VM servers. You perform this operation before including the bond in a network configuration.

Release Ownership

You release ownership of an Oracle VM server if you want to deploy this server to another Oracle VM environment. This removes the Oracle VM Manager information from the OVS Agent database in the Oracle VM server.

You cannot release ownership of an Oracle VM server that belongs to a server pool, whether it is clustered or not, or if the Oracle VM server is not running.

Deleting Oracle VM Servers

- You can delete an unassigned Oracle VM server at any time.
- If the Oracle VM server is part of a server pool:
  - Migrate running virtual machines from this server
  - Unpresent repositories from this server (optional)
  - Remove the server from the server pool
- You can now delete the server from your Oracle VM environment.

You do not have to unpresent NFS-type repositories to the Oracle VM server before removing the server from the server pool because access to NFS-type repositories does not require the server to be part of a server pool.

When you delete an Oracle VM server from a server pool, the OCFS2-type repositories active on this server are automatically unpresented.

After removing the Oracle VM server from the server pool, the server appears in the Unassigned Servers folder. You can then delete the Oracle VM server and this step removes the server from your Oracle VM environment.

Deleting an Oracle VM Server That Is No Longer Accessible

- Optionally, migrate the virtual machines on the Oracle VM server to the server pool level or to another Oracle VM server in the server pool.
- Delete the Oracle VM server:
  - The unpresent from the repository operations fails.
  - The Oracle VM server is deleted from the server pool.
- The Oracle VM server no longer appears in the environment.

Attempt to remove an Oracle VM server that is no longer available only if you do not intend to bring this Oracle VM server back into your environment.

When you remove a “dead” Oracle VM server from your environment by using the Delete action, the Oracle VM server is removed from the server pool and no longer appears in the list of servers. However, note that this behavior might be different for older releases of Oracle VM. At this point, you might want to perform cleanup operations to remove references to the server in your network and storage configuration.

You use the delete action when you are certain that the server will not be recovered or returned to the environment. If you regain access to the server at this point, you can reinstall the Oracle VM Server for x86 software and bring the server back into the Oracle VM environment. This process is discussed in the next slide.

Bringing the Oracle VM server back into your Oracle VM environment without reinstalling the Oracle VM server for x86 software is not recommended, especially if the environment has gone through several configuration changes since the Oracle VM server was removed. The original Oracle VM server still has active server pool configuration as well as network and storage configuration. At this point, if you regain access to the server, reinstall the Oracle VM Server for x86 software on it. This process is discussed in the next slide.

Reinstalling an Oracle VM Server

- If you reinstall an Oracle VM server that is already part of a server pool, the server reappears in the server pool with an unknown status.
  - The certificate-based authentication is no longer valid.
- At this point, discover the Oracle VM server to re-establish the authentication path.
  - The Oracle VM server is now unassigned.
- To complete the recovery:
  - Configure network and storage for the Oracle VM server
  - Add the Oracle VM server to the server pool
  - Present repositories to the Oracle VM server

If an Oracle VM server is down, a red icon appears on the server in the Oracle VM Manager UI and events are logged against it. If the Oracle VM server is restarted, normal operation resumes. If the Oracle VM server is used for other purposes and later reinstalled with the Oracle VM Server for x86 software, you can bring this server back into its former environment.

When you boot the Oracle VM server after it has been reinstalled, the Oracle VM Manager can no longer communicate with this server because the certificate-based authentication is broken. The Oracle VM Manager knows that the server is up but the server status is unknown.

At this point, rediscover the server. After discovery, the Oracle VM server is now unassigned.

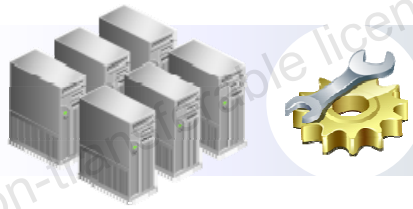
You can now install the Storage Connect vendor plug-ins in the Oracle VM server and add the Oracle VM server to the required networks and storage.

When the Oracle VM server is participating in all the required networks and has access to the required NFS shares and LUNs, you add the server to its former server pool. Next, present the appropriate repositories to this Oracle VM server.

The recovery is now complete and your Oracle VM server is ready to run virtual machines.

Server Pool Configuration

- For clustered server pools, you specify:
  - A cluster timeout value
  - A server pool file system
  - Optionally, a dedicated heartbeat network
- Oracle VM servers in the same clustered server pool must reside in the same physical location.



Clustered server pools use Oracle Cluster File System 2 (OCFS2), a shared-disk cluster file system to implement the cluster.

When you create a clustered server pool, you specify a timeout for the cluster and a server pool file system. The timeout value is the length of time until an Oracle VM server is considered dead. The default value is 120 seconds when using the Oracle VM Manager UI. The range for the timeout value is between 30 and 300 seconds.

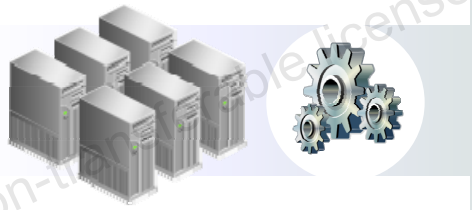
After you have created a server pool, you cannot change the timeout for the cluster from the Oracle VM Manager UI. You can change the timeout from the Oracle VM CLI if there are no Oracle VM servers in the server pool.

Changing the cluster timeout by manually editing the OCFS2 configuration files requires an outage across the entire server pool. Contact My Oracle Support if you must change the timeout configuration for one of your server pools without having to dismantle the server pool.

The pool file system that you specify when creating the server pool becomes the cluster global heartbeat device. Each Oracle VM server in the server pool periodically reads from and writes to this heartbeat device. All Oracle VM servers in the same clustered server pool must reside in the same physical location to minimize latency problems. Latency caused by distance between an Oracle VM server and its server pool file system can result in the Oracle VM server fencing itself and rebooting.

Server Pool Operations

- You can add Oracle VM servers to a server pool at any time.
- If an Oracle VM server is shut down, the server leaves the OCFS2 cluster.
  - The server can rejoin the cluster when it comes up again.
- If an Oracle VM server fails to write to the heartbeat device, it initiates a self-fence operation.
- During the reboot, the server rejoins the cluster.
  - If the Oracle VM server fails to rejoin the cluster, find out why the Oracle VM server cannot access the server pool file system.



The Oracle VM cluster structure of the server pool is based on OCFS2, with a global hearbeat.

The global heartbeat for a clustered server pool starts when the first Oracle VM server is added to the server pool. You can add additional Oracle VM servers to the server pool and these servers are added to the cluster configuration. When you shut down the last Oracle VM server in a clustered server pool, the cluster comes down.

If an Oracle VM server in a clustered server pool fails, the cluster remains functional, but the number of nodes in the cluster goes down by one. When the Oracle VM server comes back up, it rejoins the cluster if it can write to the heartbeat device, which is the pool file system.

When a running Oracle VM server fails to write to the heartbeat device within the timeout value specified for the server pool, the Oracle VM server self-fences by rebooting. This self-fencing avoids the situation where the other servers in the server pool mark the server as dead while it is still alive.

If the Oracle VM server fails to rejoin the server pool cluster when it reboots, it is most likely because the Oracle VM server cannot access the server pool file system.

Oracle VM Repositories

- You can present an NFS-based repository to any Oracle VM server that has access to its NFS share.
- You can present an OCFS2-based repository created on an external LUN only to Oracle VM servers of the server pool specified when creating the repository.
- You can create a local repository on an Oracle VM server:
  - On any unused local disk of the Oracle VM server
  - On the unused portion of the Oracle VM server installation disk if this unused space appears as a physical disk in the local storage volume group for the Oracle VM server

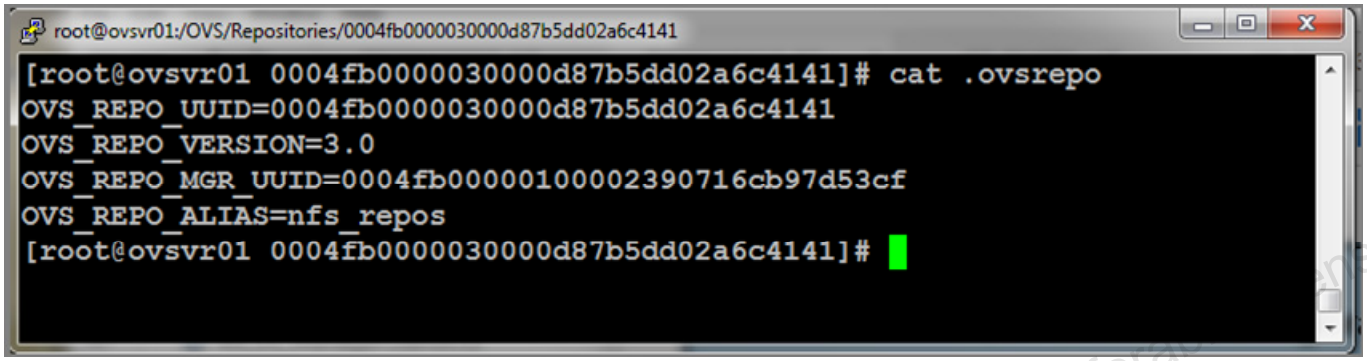
You can create three types of Oracle VM repositories:

- **NFS-type repositories:** You can create an NFS repository on an NFS share without an existing server pool as long as the Oracle VM server used to create the repository can access the NFS share. Therefore, the NFS repository can be accessed across server pools.
- **LUN-based repository:** You must specify a clustered server pool to create a repository on an external LUN. The repository can be presented only to Oracle VM servers that belong to the server pool specified when creating the LUN-based repository.
- **Local repository:** You can create an OCFS2-type repository on the internal disk or local partition of an Oracle VM server in a clustered or an unclustered server pool.

When you create a repository on a local disk of an Oracle VM server, an OCFS2 file system is created on the physical disk, but the file system is not part of the cluster even if the Oracle VM server owning the file system is part of a clustered server pool. This local repository is local to the Oracle VM server that owns it, and you cannot share the resources in the local repository with other Oracle VM servers in the server pool.

Repository Ownership

- For all repositories, the owning Oracle VM Manager UUID is stored in the repository.
- Use the `cat` command to display the UUID information in the `.ovsrepo` file:



When you create a repository, the Oracle VM Manager UUID appears in the `.ovsrepo` hidden file in the top-level directory of the repository. This applies to all repositories.

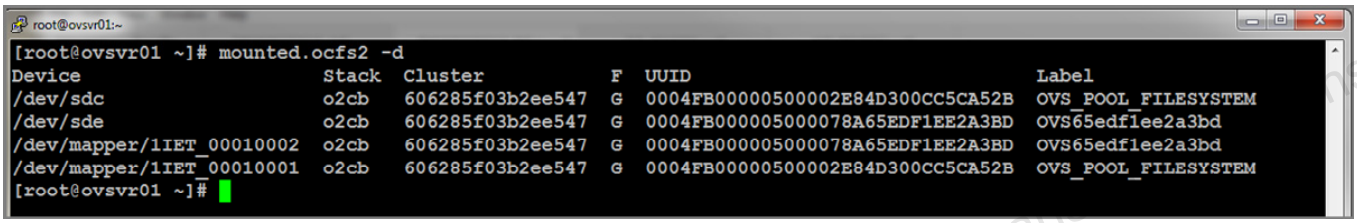
Example:

```
[root@ovsvr01 0004fb0000030000d87b5dd02a6c4141]# cat .ovsrepo
OVS_REPO_UUID=0004fb0000030000d87b5dd02a6c4141
OVS_REPO_VERSION=3.0
OVS_REPO_MGR_UUID=0004fb00000100002390716cb97d53cf
OVS_REPO_ALIAS=nfs_repos
[root@ovsvr01 0004fb0000030000d87b5dd02a6c4141]#
```

Repository with Clustered Server Pools

OCFS2-type repositories on external LUNs always belong to a single server pool.

- The repository and the server pool belong to the same cluster.
- Each OCFS2 file system has two entries: a `/dev/device` entry and a `/dev/mapper` entry.
- Notice that the server pool file system and the two repositories share this same cluster ID.
- Use the `mounted.ocfs2 -d` command to display repository and cluster information.



When you create a repository for a clustered server pool, the repository now belongs to that server pool and cannot be owned by other server pools.

The display shows two repositories, with two entries for each repository: one entry with the disk device and the other entry with the corresponding `/dev/mapper` device.

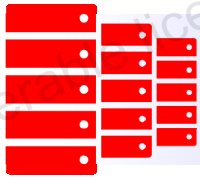
You can display the cluster ID stored in the repository by using the `mounted.ocfs2` command.

Repositories on Local Disk

Repositories on local disk is an OCFS2-type file system but does not belong to a server pool.

Example:

```
# mounted.ocfs2 -d
```



When you create a repository on the local disk of an Oracle VM server, an OCFS2-type repository is created but it does not belong to the cluster.

For example, executing the `mounted.ocfs2 -d` command on the Oracle VM server:

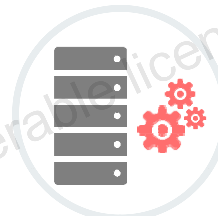
```
# mounted.ocfs2 -d
Device          Stack   Cluster           F  UUID
                                                                  Label
/dev/sdb        None                        0004FB0...00419BA9EE85A5FF5B
                                                               OVSba9ee85a5ff5b
/dev/sdd        o2cb    1cd324510a7639d6  G 0004FB0...0069AE9294CDD5D6BF
                                                                 OVSe9294cdd5d6bf
/dev/mapper/35...1990 None                  0004FB0...00419BA9EE85A5FF5B
                                                                 OVSba9ee85a5ff5b
/dev/mapper/36...053  o2cb 1cd324510a7639d6 G
                                                        0004FB0...0069AE9294CDD5D6BF
                                                                 OVSe9294cdd5d6bf
```

The device named `/dev/sdb`, with corresponding `/dev/mapper/35...1990`, contains an OCFS2-type repository on the local disk of the Oracle VM server, and is not part of the server pool cluster.

Oracle VM Repository Operations

You can perform the following operations on repositories:

- Present and unpresent repositories to Oracle VM servers.
- Refresh a repository to synchronize its contents with the Oracle VM Manager database.
- Create a repository export for OCFS2-type repositories.
- Delete a repository and its contents.
- Release ownership of a repository.
- Reclaim ownership of a repository.



Present and Unpresent a Repository to Oracle VM Servers

You can present NFS-type repositories to Oracle VM servers from different server pools, clustered or not.

You can present only OCSF2-type repositories that reside on external LUNs to Oracle VM servers that belong to the same cluster. When using the Oracle VM Manager UI to present a repository to an Oracle VM server, only candidate Oracle VM servers are listed.

Refresh a Repository

When you refresh a repository, the contents of the repository are synchronized with the repository information in the Oracle VM Manager database:

- The information is updated in the database for all repository objects, such as virtual disks, virtual CD/ROMs, configuration files for virtual machines and templates, and assemblies.
- The content of configuration files for virtual machines and templates is examined and the database is updated if necessary.

Create a Repository Export

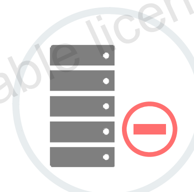
You can export an OCFS2-type repository to allow a backup application to back up its contents.

The delete and release ownership operations are discussed next.

Removing a Repository

You can remove a repository from your Oracle VM environment by:

- Deleting the repository:
  - The repository must be empty before attempting to delete it.
  - Perform the delete operation from any Oracle VM Manager management interface with at least one Oracle VM server still presented to the repository.
- Releasing ownership of the repository:
  - In the Oracle VM Manager UI, edit the repository and select to release the ownership.
    OR
  - In the Oracle VM CLI, use the `releaseOwnership Repository` command.



ORACLE

When you release the ownership of an OCFS2-type repository, the cluster identification and the Oracle VM Manager UUID are removed from the repository. When you release an NFS-type repository, only the Oracle VM Manager UUID is removed.

The following example shows the output of the `mounted.ocfs2` command executed from an Oracle VM server with access to the OCFS2-type repository, after the ownership of the repository was released:

```
# mounted.ocfs2 -d
Device                  Stack  Cluster   F  UUID                Label
/dev/sdh                o2cb   NOTINUSE  G  0004FB000005...B1982B8C51299290
                                                           OVS82b8c51299290
/dev/mapper/360...002   o2cb   NOTINUSE  G  0004FB000005...B1982B8C51299290
                                                           OVS82b8c51299290
```

In the command output, both entries refer to the same disk, which is an OCFS2-type repository. The cluster ID was replaced with `NOTINUSE` when the repository's ownership was released.

You cannot release the ownership of a repository if there are objects in the repository that are in use by other objects in another repository. For example, if there is a virtual disk in the repository and that virtual disk is in use by a virtual machine defined in another repository, you cannot release the ownership of the repository.

When you release the ownership of a repository, it becomes available to other server pools where the Oracle VM servers have access to the storage for the repository.

Reclaiming a Repository After Rebuilding Oracle VM

- After rebuilding your Oracle VM environment with the same Oracle VM Manager UUID, the rebuilt server pools have new cluster IDs.
- OCFS2-type repositories from the previous environment have the cluster ID from their former server pools, if their ownership was never released.
- You can reclaim each repository by updating the cluster ID on the repository by using an OCFS2 utility.

If you rebuild your Oracle VM environment, you can specify the UUID for the Oracle VM Manager that was in use in your previous environment. When your NFS storage is reconfigured, you can refresh the NFS-type repositories and the repository information is added to the Oracle VM Manager database.

If you did not release the ownership of your OCFS2-type repositories before you rebuilt your Oracle VM environment, these repositories still have the cluster ID of their previous clustered server pool.

After reconfiguring the storage arrays that contain the existing OCFS2 repository devices, you can display the OCFS2 information from your Oracle VM servers with the `mounted.ocfs2` command, even if the devices are not currently mounted in the Oracle VM servers.

If you have more than one server pool in your new environment, identify which repositories are to be reclaimed by each server pool.

Perform the following steps for each repository from one of the Oracle VM servers that belong to the new server pool claiming the repository:

- Identify the device ID for the repository LUN to reclaim.

- Run the `tunefs.ocfs2 --update-cluster-stack <device ID>` command.
- From the Oracle VM Manager UI or Oracle VM CLI, refresh the file system on the repository device.

  Note: The file system refresh operation is discussed in the next slide.
- Refresh the repository itself to synchronize its contents with the Oracle VM database.

At this point, you can access and manage the objects in the repository.

The task of reclaiming a repository after rebuilding your Oracle VM environment is discussed further in the lesson titled “Backup and Restore D/R Concepts.”

Storage Refresh Operations

- You perform refresh operations to notify the Oracle VM Manager of changes to the physical environment of your Oracle VM servers.
- Refresh operations include:
  - Oracle VM server rediscovery
  - File server refresh
  - File system refresh
  - SAN server refresh
  - Physical disk refresh

When you configure your storage from one of the Oracle VM management interfaces, many of the actions perform a refresh of the associated object as the last step of the operation. For example, if you delete an object in a repository, the operation triggers a refresh of the repository file system after the object is deleted.

You perform refresh operations when you make changes to your Oracle VM physical environment and you want to inform the Oracle VM Manager of these changes. For example, if you resize a LUN or remove a LUN from a storage array that is managed using a generic plug-in, the Oracle VM Manager is not aware of the changes until you refresh the storage array.

Oracle VM Server Refresh

You refresh an Oracle VM server by rediscovering it. Rediscovery operations were discussed earlier in this lesson.

File Server Refresh

A file server refresh looks for new and missing file systems in the file server. The refresh operation updates file systems and access group information.

Note: File server Access Groups contain the list of Oracle VM servers that are used for refresh operations when the file server offers non-uniform exports. You create Access Groups to make sure that each file system in the file server can be refreshed by at least one Oracle VM server.

File System Refresh

When you refresh a file system, the refresh operation detects the file system size and the presence of a repository or pool file system on the file system. If a repository is found, the information in its `.ovsrepo` file is used to update the Oracle VM Manager information.

Storage Array (SAN Server) Refresh

A SAN server refresh looks for new and missing disks in the storage array. It initiates a SCSI bus rescan operation on all Oracle VM servers that have initiators to that storage array.

Rescan Physical Disks

If you want to refresh the physical disk configuration for a single Oracle VM server, use the Rescan Physical Disks operation for that Oracle VM server from the Oracle VM Manager UI.

The equivalent Oracle VM CLI command is `refreshStorageLayer server name=<`*`Oracle VM server name`*`>` or `refreshStorageLayer server id=<`*`Oracle VM server id`*`>`.

Physical Disk Refresh

A physical disk refresh updates the disk size information if the LUN has been resized. Similar to the file system refresh, the physical disk refresh detects the presence of a repository or a server pool file system on the disk.

Quiz

Which of the following refresh operations can you perform in your Oracle VM environment?

a. File server refresh
b. Repository refresh
c. Oracle VM Manager refresh
d. All of the above

Answer: a, b

You cannot refresh an Oracle VM Manager. You can, however, perform a Refresh All to rediscover all Oracle VM server instances, file servers, and storage arrays in your Oracle VM environment. The Refresh All action is recommended after a restore of the Oracle VM Manager database.

Summary

In this lesson, you should have learned how to:

- Discover, rediscover, add, and remove Oracle VM servers from your environment
- Describe server pool operations
- Rediscover repositories, move repositories between server pools, and reclaim repositories
- Notify your Oracle VM Manager about changes to the storage layer in your Oracle VM environment

ORACLE

Practice Overview

In this practice, you:

- Simulate the hard removal of one Oracle VM server by shutting it down and reinstalling the software on the server
- Perform a clean removal of one Oracle VM server, and add this Oracle VM server to a new server pool
- Perform a migration for the NFS-based virtual machine between the two server pools
- Create a second repository for `Pool1`
- Create dependencies between two repositories owned by one server pool, and release ownership of one of the repositories
- Reclaim ownership of the newly released repository by the other server pool

6

Troubleshooting

Objectives

After completing this lesson, you should be able to:

- Locate the major Oracle VM logs and discuss their usage
- Execute commands to examine the status of the major components of Oracle VM
- Describe the use for the VMPinfo3 and `sosreport` tools
- Configure Kdump for Oracle VM servers
- Configure `netconsole` for Oracle VM servers
- Produce core dumps for virtual machines
- Access the serial console for Linux guests

Lesson Overview

- In this lesson, you learn to use the tools that help you to:
  - Examine your environment when it is running normally
  - Detect problems that disrupt normal operation
- These tools include:
  - Log files located in the Oracle VM Manager host and in the Oracle VM servers
  - Commands to examine the status of various components
  - Tools to gather troubleshooting information or to format log files
  - Procedures that prepare your Oracle VM servers to provide services such as memory dumps and remote logging capability

This lesson also provides some troubleshooting tools for the virtual machines in your Oracle VM environment.

Finding Information About Errors

You can find information to help you solve error conditions in the following locations:

- Oracle VM Manager logs
- Oracle VM Manager's job details
- Oracle VM Manager's events
- Oracle VM Server logs

Oracle VM Manager Job Details

- Example 1

```
OVMRU_002043E Cannot release ownership of Repository: nfs222_repos2. Virtual
Disks/CDroms: [hvm1_vdisk3 on hvm1], are still assigned to VMs/Templates that have
configuration files in another repository. [Wed Jun 25 16:06:09 MDT 2014]
```

- Example 2

```
OVMRU_002021E Cannot perform this operation. The repository: nfs222_repos2, must first be
refreshed. [Wed Jun 25 16:27:28 MDT 2014]
```

When you submit a job, it might fail quickly. For example, if you attempt to delete a repository that is not empty, you get a message informing you of the problem and the job fails.

If a job is running, you can view the job details while it is running. Job details are most useful after the job completes or fails. By viewing the job details, you can often determine why a job failed.

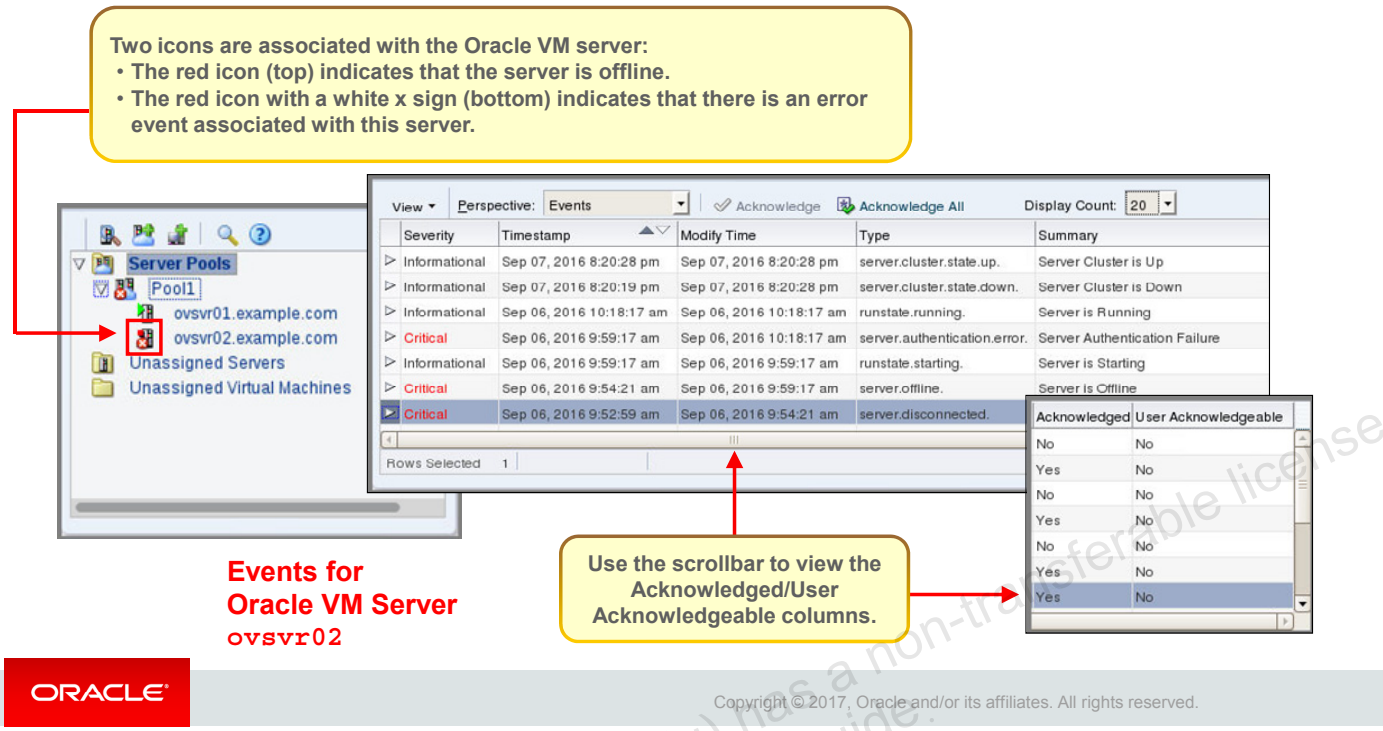
Example 1

The first example in the slide shows an extract of the job details for a job attempting to release ownership of a repository. The job fails because there is a dependency for a virtual disk in the repository for which ownership is being released. The virtual disk is part of a virtual machine configuration that resides in another repository.

Example 2

In the second example, an edit is attempted for a resource in an NFS repository. The repository was newly recovered and had not yet been refreshed. The edit job fails, and in the job details, error `OVMRU_002021E` indicates that a refresh of the repository must be performed before attempting to manipulate the resources in the repository.

Oracle VM Manager Events



Oracle VM flags objects to recognize important milestones (such as object creation) and error conditions associated with the objects. You can examine events for the following objects: Oracle VM servers, virtual machines, repositories, and storage objects.

Most events, such as those flagging object creation, do not need any action on your part.

Error events are generally acknowledged automatically by Oracle VM Manager. Very few error events require that you acknowledge them. To find out if an event is user-acknowledgeable and/or has been acknowledged:

- Oracle VM Manager UI: Examine the Acknowledged/User Acknowledgeable columns from the Events perspective for the object under examination.
- Oracle VM CLI: Use the `getevents` command.

  For example: `getEvents objtype=server objId=ovsvr02 acknowledged=No`

The individual screenshots in the slide show details visible in the different panes of the Servers and VM tab. In the management pane, there are three Critical events shown for Oracle VM server `ovsvr02`. These events were triggered when the Oracle VM server was shut down from the server command line, rather than by using the Stop Server action from the Oracle VM Manager UI.

Two icons appear in the navigation pane on Oracle VM server `ovsvr02`:

- A red icon indicating that the Oracle VM server is offline
- A red octagon with a white x sign in it indicating an error event

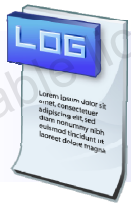
The smaller screenshot in the slide (to the right of the slide) shows the Acknowledged/User Acknowledgeable columns for the events listed in the screenshot of the navigation pane.

If possible, use one of the Oracle VM Manager management interfaces to stop Oracle VM servers. If you use the Oracle VM Manager to stop an Oracle VM server, the Oracle VM Manager logs an informational event against the server and only the red icon appears on the server, indicating that the server is offline.

Always examine the events when your environment experiences a failure. The events yield information about the sequence of events when an object fails. For example, if an Oracle VM server loses access to its server pool file system, you can see what happens in the events pane for the server.

Oracle VM Manager Logs

- **`access.log`**
  - You use this log to track HTTP access to the web interface of the Oracle VM Manager.
- **`AdminServer.log`**
  - This is the most useful log to track activity in the Oracle VM Manager.
  - It also tracks scheduled tasks in the Oracle VM Manager, such as backup file rotation.
- **`AdminServer-diagnostic.log`**
  - This log tracks exceptions in the underlying WebLogic framework.



All Oracle VM Manager log files are located in the `/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs` directory.

`access.log`

This log tracks HTTP access to the web interface of the Oracle VM Manager and to the underlying WebLogic HTTP interface. Because this log tracks access and HTTP operations within the Oracle VM Manager, you can use it to debug access issues and to audit access to the Oracle VM Manager.

`AdminServer.log`

This log tracks events within the underlying Oracle WebLogic framework, including events triggered by Oracle VM Manager. You can use this log to investigate a variety of issues in the Oracle VM Manager including TLS/SSL certificate issues, server availability issues, and any actions performed from the Oracle VM Manager. You can identify these actions by searching for items containing the string "*com.oracle.ovm.mgr".* Login failures resulting from locked accounts (as opposed to incorrect credentials) are also logged to this file.

Example 1: Server Rediscovery during Oracle VM Manager Restart

```
####<2016-09-16T18:41:33.472+0000> <Info>
<com.oracle.ovm.mgr.task.AutoDiscoverTask> <ovmmgr01.example.com>
<AdminServer> <Scheduled Tasks-7> <<anonymous>> <> <4e72859a-8d85-
49ed-8d0f-5439f2388f05-00000004> <1474051293472> <BEA-000000> <(1 of
2) Starting re-discovery of VMs on server: ovsvr01.example.com,
after Manager restart>
####<2016-09-16T18:41:38.149+0000> <Info>
<com.oracle.ovm.mgr.task.AutoDiscoverTask> <ovmmgr01.example.com>
<AdminServer> <Scheduled Tasks-7> <<anonymous>> <> <4e72859a-8d85-
49ed-8d0f-5439f2388f05-00000004> <1474051298149> <BEA-000000> <(2 of
2) Starting re-discovery of VMs on server: ovsvr02.example.com,
after Manager restart>
```

Example 2: MySQL Backup File Rotation

```
####<2016-09-09T20:42:45.881+0000> <Info>
<com.oracle.ovm.mgr.api.manager.ModelManager> <ovmmgr01.example.com>
<AdminServer> <Odof Tcp Client Thread: /127.0.0.1:54321/2393>
<<anonymous>> <> <8674c3c9-5157-4c06-a718-bf24ecbdbabc-00000004>
<1473453765881> <BEA-000000> <Removed 1 oldest files, left 10
files.>
```

AdminServer-diagnostic.log

This log tracks exceptions within the underlying Oracle WebLogic framework.

Manipulating Oracle VM Manager Logs with the `OvmLogTool.py` Tool

- Generating a full formatted listing

```
[root@ovmmgr01 bin]# python OvmLogTool.py -o full
```

- Creating a summary, using the default internal filters

```
[root@ovmmgr01 bin]# python OvmLogTool.py -o summary
```

- Creating a formatted listing, using a filter file

```
[root@ovmmgr01 bin]# python OvmLogTool.py -f ExceptionFilters -o ExceptionListing
```

- Tailing the active log file (Use `Ctrl-C` to exit)

```
[root@ovmmgr01 bin]# python OvmLogTool.py -t
tailing log file: /u01/app/oracle/ovm-manager-3/domains/ovm_domain/
servers/AdminServer/logs/AdminServer.log
```

ORACLE

Because the Oracle VM Manager log file format is determined by Oracle WebLogic, the log files might be difficult to read. You can use the `OvmLogTool.py` Python tool to help you examine the log files. The `OvmLogTool.py` utility is a log parsing tool that can:

- Convert and combine all the `AdminServer.log`<*log#*> log files into one file
- Create a filtered summary log file that lists only certain types of errors
- Tail the `AdminServer.log` file and apply a filter on the fly

The `OvmLogTool.py` utility resides in the `/u01/app/oracle/ovm-manager-3/ovm_tools/bin` directory in the Oracle VM Manager host.

Slide Examples

1. The first example in the slide uses the `OvmLogTool.py` utility to generate a fully formatted listing, including all available `AdminServer.log`<*log#*> log files. A file named `full` is generated in the current directory.
2. In the second example, a summary is generated for all available `AdminServer.log`<*log#*> log files, by using the `LogFilter.properties` default filter file. A file named `summary` is generated in the current directory. The default filter file is located at `/u01/app/oracle/ovm-manager-3/ovm_tools/etc/LogFilter .properties`.

3. The third example uses a custom filter file, which extracts exception type log messages. The filter file, named `ExceptionFilters`, is created by copying the default filter file, `LogFilter.properties`, and removing all filters except those for exception conditions. When running the command in the third example, a file named `ExceptionListing` is generated in the current directory.
4. In the fourth example, the current `AdminServer.log` file is tailed to display incoming log messages. Press Ctrl + C to exit the `tail` operation.

Options

Use the `-h` or `--help` option with the `OvmLogTool.py` utility to display usage information and a list of available options.

Oracle VM Server Logs

- The main log file is called `ovs-agent.log` and is located in the `/var/log/` directory in each Oracle VM server.
- You can easily follow the steps for each operation.

  For example, rediscovering an Oracle VM server:

```
...
[2016-09-16 13:13:03 4210] DEBUG (service:77) call start: discover_server
[2016-09-16 13:13:03 4210] DEBUG (service:77) call complete: discover_server
[2016-09-16 13:13:04 4280] DEBUG (service:77) call start: discover_hardware
[2016-09-16 13:13:04 4280] DEBUG (service:77) call complete: discover_hardware
[2016-09-16 13:13:04 4284] DEBUG (service:77) call start: discover_network
[2016-09-16 13:13:04 4284] DEBUG (service:77) call complete: discover_network
[2016-09-16 13:13:04 4293] DEBUG (service:77) call start: discover_storage_plugins
[2016-09-16 13:13:04 4293] DEBUG (service:77) call complete: discover_storage_plugins
[2016-09-16 13:13:04 4296] DEBUG (service:75) call start: discover_physical_luns('',)
[2016-09-16 13:13:05 4296] DEBUG (service:77) call complete: discover_physical_luns
...
```

ORACLE

You can find the main Oracle VM server log file in the `/var/log` directory. The log file name is `ovs-agent.log`.

The Oracle VM Manager application receives requests from all of the Oracle VM management interfaces and sends the appropriate commands to an Oracle VM server. The Oracle VM server selected depends on many factors, including the server roles assigned to the servers. For example, if you start a virtual machine, in addition to selecting an Oracle VM server located in the appropriate server pool, the Oracle VM Manager takes into account the server roles assigned to the Oracle VM servers in the server pool. When you are starting a virtual machine, the Oracle VM Manager selects only an Oracle VM server that has the VM Server role assigned to it.

Messages that appear in `AdminServer.log` on the Oracle VM Manager host when you start a virtual machine:

```
####<2016-09-13T21:00:28.741+0000> <Info>
<com.oracle.ovm.mgr.api.queuedjob.QueuedJob> <ovmmgr01.example.com> <AdminServer>
<Odof Tcp Client Thread: /127.0.0.1:54321/91> <<anonymous>> <> <8674c3c9-5157-
4c06-a718-bf24ecbdbabc-00000004> <1473800428741> <BEA-000000> <Created queued
child job: 1473800428725/QueuedVmStartDbImpl: OVMJOB_1502J Assign vm: iscsi_pvm3,
to server: ovsvr02.example.com/[InternalJobDbImpl]
QueuedVmStartDbImpl_1473800428724<3839>/t=1473800428725>

...
```

```
####<2016-09-13T21:00:28.828+0000> <Info>
<com.oracle.ovm.mgr.op.virtual.VirtualMachineStart> <ovmmgr01.example.com>
<AdminServer> <Odof Tcp Client Thread: /127.0.0.1:54321/15182>
<<anonymous>> <> <8674c3c9-5157-4c06-a718-bf24ecbdbabc-00000004>
<1473800428828> <BEA-000000> <Starting virtual machine [iscsi_pvm3] on
[ovsvr02.example.com]>
```

The Oracle VM Manager selects `ovsvr02` as the Oracle VM server to run the virtual machine. The messages appearing in the `ovs-agent.log` log file for the `ovsvr02` Oracle VM server are as follows:

```
[2016-09-13 13:11:10 25179] DEBUG (service:75) async call start:
start_vm('0004fb00000300001b76c831770fa6d5',
'0004fb0000060000e32f93ec3d508574')

...

[2016-09-13 13:11:27 2043] DEBUG (notificationserver:192) sent events:
[('{DOMAIN} 0004fb00-0006-0000-e32f-93ec3d508574 {VMAPI} VMAPIEvent
{"VMAPIEvent":{"severity":5,"subsystem":"ovmdSS","process":"ovmd","type":"s
ystem","payload":{"type":"alive","alive":{"hostname":"iscsipvm2.example.com
","domainName":"unknown","osType":"Linux","osVersion":"Oracle Linux Server
release 6.4","kernelVersion":"2.6.39-
400.17.1.el6uek.x86_64","guestDriverVersion":"2.6.39-
400.17.1.el6uek.x86_64","arch":"x86_64","guestType":"PV","vmapiVersion":"10
0"}}}}', {}), ('{DOMAIN} 0004fb00-0006-0000-e32f-93ec3d508574 {VMAPI}
VMAPIEvent
```

In addition to the main log file, `ovs-agent.log`, you can examine the following log files in the `/var/log` directory in each Oracle VM server:

`osc.log` File

This file contains a log for the Oracle VM Storage Connect plug-ins.

`ovm-consoled.log` File

This file contains a log for the Oracle VM virtual machine console.

`ovmwatch.log` File

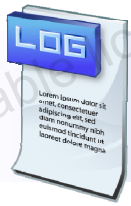
This file contains a log for the Oracle VM watch daemon.

`messages` File

You can find valuable troubleshooting information in the system log file, `/var/log/messages`, on each Oracle VM server. This log file is particularly useful for storage and networking events. For example, if an access path to a LUN fails, this event is reported in the `messages` file before the problem is detected by the Oracle VM Agent software.

Log Entries for Virtual Machines

- The Oracle VM server log, `ovs-agent.log`, contains information about virtual machine startup and error conditions.
- Examine the Xen logs in the `/var/log/xen` directory on the Oracle VM server where the virtual machine is running:
  - `xend.log`
  - `xend-debug.log`
  - `xen-hotplug.log`
  - `qemu-dm.<domain name>.log`



`xend.log` File

The most useful Xen log file for troubleshooting virtual machine problems is `xend.log`, located in the `/var/log/xen` directory on the Oracle VM server where the virtual machine is experiencing problems. This file contains a log of actions by the `xend` daemon, and includes information about starting, stopping and suspending virtual machines. Both normal actions and error conditions are recorded in this log.

`xend-debug.log` File

This log is not as useful because it records information about Xen programs and problems experienced by these programs rather than information about Xen actions and events.

`xen-hotplug.log` File

This log contains a list of hot-plug events which are events about connecting to a device. Devices include virtual interfaces such as VIFs, Xen bridges, and storage devices. Hot-plug events are logged if a device or network script does not start or becomes unavailable.

`qemu-dm.<domain name>.log` File

This log file contains a log for each HVM guest. This log file is created by the `qemu-dm` process. The domain name that is included in each log file is the UUID for the virtual machine. Use the Oracle VM CLI `list vm` command to easily match a virtual machine name to its UUID.

Quiz

Which statements are true about the various logs available in your Oracle VM environment?

a. The Oracle VM Manager provides several logs, all located in the `/u01/app/oracle/ovm-manager-3/domains/ovm_domain/servers/AdminServer/logs` directory.

b. You use the `OvmLogTool.py` tool to examine Oracle VM server log files.

c. The `/var/log/messages` file in each Oracle VM server is an important source of information for storage and networking events.

d. You can find useful troubleshooting information for virtual machines in the Xen logs located in the Oracle VM server where the virtual machine is running.

Answer: a, c, d

Oracle VM Manager Database: MySQL

- For the MySQL database, the database files reside in the `/u01/app/oracle/mysql/data` directory.
- The log file also resides in the `data` directory and is called `mysqld.err`:

```
...
2016-08-25 20:11:00 2582 [Note] InnoDB: Database was not shutdown normally!
2016-08-25 20:11:00 2582 [Note] InnoDB: Starting crash recovery.
2016-08-25 20:11:00 2582 [Note] InnoDB: Reading tablespace information from the .ibd files...
2016-08-25 20:11:05 2582 [Note] InnoDB: Restoring possible half-written data pages
2016-08-25 20:11:05 2582 [Note] InnoDB: from the doublewrite buffer...half-written
                                                        data pages
...
2016-08-25 20:11:09 2582 [Note] Event Scheduler: Loaded 0 events
2016-08-25 20:11:09 2582 [Note] /usr/sbin/mysqld: ready for connections.
```


The database files are located in the `/u01/app/oracle/mysql/data` directory. The error log for the MySQL database also resides in this directory. The name for the error log is `mysqld.err`, as specified in the `my.cnf` file in the same directory.

`mysqld.err` Log File

In this file, you can determine when the MySQL database was started and stopped. If the database was not shut down properly, the error log displays messages about the recovery process. You can also find critical errors that occur while the database is running.

Checking the Status of the Oracle VM Manager Databases

1. Connect to the database (`ovs` or `appfw`):

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

2. Issue the `status` command:

```
mysql> status
--------------
mysql  Ver 14.14 Distrib 5.6.27, for Linux (x86_64) using  EditLine wrapper


Connection id:          8698
Current database:       ovs
...
Server version:         5.6.27-enterprise-commercial-advanced MySQL Enterprise Server -
                          Advanced Edition (Commercial)
...
UNIX socket:            /u01/app/oracle/mysql/data/mysqld.sock
Uptime:                 25 days 17 hours 43 sec
...
mysql>
```


If you are experiencing problems accessing the Oracle VM Manager UI or the Oracle VM CLI, verify that the Oracle VM Manager database is running.

To verify the status of the Oracle VM Manager database, perform the following steps:

1. Access the Oracle VM Manager host.
2. Execute the `mysql` command and provide the database name (`ovs` or `appfw`), the path to the connection socket, and the username for the connection.

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

3. When prompted for the password for the user `root`, provide the password specified during the installation of the Oracle VM Manager software.
4. After connecting to the MySQL database, issue the `status` command.

   If the database is running normally, the command returns information about the current connection and specifies the uptime for the database.
5. Connect to the second database by using the `connect <database name>` command at the `mysql` prompt and repeat the `status` command. For example, if you connected initially to the `ovs` database, use the `connect appfw` command to display the status information for the `appfw` database.
6. Use the `quit` command to exit the `mysql` command.

OCFS2 File Systems in Oracle VM Servers

OCFS2 file systems on Oracle VM servers comprise:

- A server pool file system if the Oracle VM server belongs to a clustered server pool
- All OCFS2-type repositories that are accessible to the Oracle VM server

```
[root@ovm2 ~]# mounted.ocfs2 -d
Device                  Stack  Cluster           F  UUID                              Label
/dev/sdb                o2cb   a32185374d5f9928  G  0004FB0000050000DC1E10A8A63835B7  OVS_POOL_FILESYSTEM
/dev/sdc                o2cb   606285f03b2ee547  G  0004FB00000500002E84D300CC5CA52B  OVS_POOL_FILESYSTEM
/dev/sdd                o2cb   606285f03b2ee547  G  0004FB000005000028E0ACF0A17AD530  OVS0acf0a17ad530
/dev/sde                o2cb   NOTINUSE          G  0004FB000005000078A65EDF1EE2A3BD  OVS65edf1ee2a3bd
/dev/mapper/1IET_00020001  o2cb  a32185374d5f9928  G  0004FB0000050000DC1E10A8A63835B7  OVS_POOL_FILESYSTEM
/dev/mapper/1IET_00020002  o2cb  606285f03b2ee547  G  0004FB000005000028E0ACF0A17AD530  OVS0acf0a17ad530
/dev/mapper/1IET_00010002  o2cb  NOTINUSE          G  0004FB000005000078A65EDF1EE2A3BD  OVS65edf1ee2a3bd
/dev/mapper/1IET_00010001  o2cb  606285f03b2ee547  G  0004FB00000500002E84D300CC5CA52B  OVS_POOL_FILESYSTEM
...
```

Server Pool FS

Same Repository Device

Same Cluster ID

For each Oracle VM server in your environment, you can expect the following OCFS2 file systems to be mounted:

Oracle VM Server Pool File System

If your Oracle VM server belongs to a clustered server pool, the pool server file system for that server pool is mounted in the Oracle VM server. As shown in the slide, this file system is identified by the `OVS_POOL_FILESYSTEM` label.

Storage Repositories

For clustered server pools, you can create repositories on iSCSI or FC external LUNs. The repositories are created as OCFS2 file systems. When an OCFS2-type repository is presented to an Oracle VM server, the OCFS2 file system that makes up the repository is mounted to the server.

In the example in the slide, two OCFS2-type repositories are present on LUNs accessible by the Oracle VM server named `ovm2`. Each repository has two entries in the output of the `mounted.ocfs2` command: One entry for the disk device in `/dev/sd[x]` format, and one entry for its corresponding `/dev/mapper` device. There is only one `/dev/mapper` device per LUN, but a LUN can appear more than once if there is more than one path to the device. The device mapper driver in Linux works with the multipath driver to present a single view of a device to an application.

OCSF2 File System on Repositories and Associated Cluster ID

When you examine the cluster ID information in the output of the `mounted.ocfs2 -d` command in the slide, you find:

- One repository (on `/dev/sdd`), with cluster ID of `98da721c3e78b778`

  This is the repository where the cluster ID of the OCFS2 file system matches the cluster ID for the server pool file system, which is the cluster ID shown for `OVS_POOL_FILESYSTEM`. This repository is currently mounted and active.
- The other repository on `/dev/sdb`, showing “NOT IN USE” for the cluster ID

  “NOT IN USE” indicates that the repository in the OCFS2 file system is not part of the active cluster. You learned in the lesson titled “Operations” that you can reclaim ownership of a repository where the cluster ID is “NOT IN USE” by using the take ownership action on the repository.

A third situation can occur where the cluster ID on an OCFS2 file system reveals a cluster ID other than the cluster ID for the active cluster. This situation indicates that the repository:

- Is owned by another clustered server pool
- Was owned by another server pool and the ownership of the repository was not released properly

In the lesson titled “Backup and Restore D/R Concepts,” you learn how to regain ownership of a repository that is still owned by a different cluster.

OCFS2 Health Check for Oracle VM Servers

```
[root@ovsrv01 ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda2             5.7G  1.8G  3.7G  32% /
tmpfs                 375M     0  375M   0% /dev/shm
/dev/sda1             477M   87M  361M  20% /boot
none                  375M   80K  375M   1% /var/lib/xenstored
/dev/mapper/1IET_00010001
                       13G  263M   13G   3% /poolfsmnt/0004fb00000500002e84d300cc5ca52b
192.0.2.1:/nfsrepos1   20G   18G  904M  96% /OVS/Repositories/0004fb0000030000d87b5dd02a6c4141
/dev/mapper/1IET_00020002
                       20G  4.2G   16G  21% /OVS/Repositories/0004fb00000300004c02ea52a901c68b
[root@ovsrv01 ~]# service o2cb status
...
Checking O2CB cluster "606285f03b2ee547": Online
...
Checking O2CB heartbeat: Active
  0004FB00000500002E84D300CC5CA52B /dev/dm-4
Nodes in O2CB cluster: 0 1
...
```


When an Oracle VM server is up and part of a clustered server pool, you might see several OCFS2 file systems mounted to the server, as described in the previous section on “OCFS2 File Systems in Oracle VM Servers.” You can display the list of active repositories by using the `df` command, as shown in the slide.

The cluster configuration is stored in the `/etc/ocfs2/cluster.conf` file:

```
root@ovm2 ocfs2]# cat cluster.conf    (output formatted to fit on the page)
node:                                                   node:
        name = ovm2                                     name = ovm1
        cluster = 98da721c3e78b778                      cluster = 98da721c3e78b778
        number = 0                                              number = 1
        ip_address = 192.168.131.202                    ip_address = 192.168.131.201
        ip_port = 7777                                  ip_port = 7777
cluster:
        name = 98da721c3e78b778
        heartbeat_mode = global
        node_count = 2
heartbeat:
        cluster = 98da721c3e78b778
        region = 0004FB0000050000008BAA29C845100F
```

To verify that the Oracle VM server is part of the cluster, issue the `service o2cb status` command. The output of this command displays the number of nodes in the cluster and the location of the heartbeat device, which is the server pool file system. This information matches the cluster configuration information in the `cluster.conf` file.

You can also cross-reference the server pool file system information to the output of the `df -h` command (on this slide) or to the output of the `mounted.ocfs2` command (in the previous slide).

If the Oracle VM server fails to mount the server pool file system or any OCFS2- type repository, you might experience one of the following conditions:

- The Oracle VM server cannot access the storage LUN for the server pool file system or the repository.

  Examine and fix the Oracle VM server's access to the external LUNs for the server pool file system or repository.

- The server pool file system or repository file system is corrupted and cannot be mounted.

  With this type of problem, every Oracle VM server in the server pool is affected. You can use the `fsck.ocfs2` command from one of the Oracle VM servers to clean up the server pool file system but you must first ensure that no other Oracle VM server is accessing the file system, that is, no other Oracle VM server has this OCFS2 file system mounted.

  If the server pool file system cannot be recovered, you must restore the file system. You learn how to back up and restore server pool file systems and repositories in the lesson titled “Oracle VM Backup and Restore, and D/R Concepts.”

- In the case of an OCFS2-type repository, a failure to mount the repository file system to the Oracle VM server can indicate that the repository is not owned by the active cluster.

Important Files on Oracle VM Servers

- Release information files:
  - In the `/etc` directory: The `ovs-release` and `ovs-info` files
- Oracle VM server db files:
  - In the `/etc/ovs-agent/db` directory: `exports`, `repository`, and `server` files
  - To display the content of `db` files:

    `# ovs-agent-db dump_db <file name>`, for example:

```
[root@ovsvr01 db]# ovs-agent-db dump_db repository
{'0004fb00000300004c02ea52a901c68b': {'alias': u'iscsi_repos2',
                                      'filesystem': 'ocfs2',
                                      'fs_location': '/dev/mapper/1IET_00020002',
                                      'manager_uuid': u'0004fb0000010000239071 6cb97d53cf',
                                      'mount_point':
                                                   '/OVS/Repositories/0004fb00000300004c02ea52a901c68b',
                                      'version': u'3.0'},
...
```

Repository Name

Repository Mount Point

Oracle VM Server Release Information Files

- The `/etc/ovs-release` contains the Oracle VM server release number.
- The `/etc/ovs-info` file is particularly useful because it contains not only the Oracle VM release, but also the build number, as well as a list of the packages in the build.

Oracle VM Server `db` Files

- **`exports`:** Contains information about repositories that are exported from the Oracle VM server
- **`repository`:** Contains a list of the repositories presented to this Oracle VM server. You can use the contents of this file to match a repository name to its mount point, since the repository name is not displayed in the output of the `df` command.
- **`server`:** Contains information about the Oracle VM server, including its node (cluster) number, and its server roles (VM server and/or Utility server)

You can use the `/usr/sbin/ovs-agent-db dump_db` command to list the information in the `db` files.

Note: You must not modify the information in the `db` files, except when directed by Oracle support.

Important Files on Oracle VM Servers

- Server pool file system `db` files
  - Located in the `db` directory in the top-level directory of the server pool file system—for example, `/poolfsmnt/0004fb00000500002e84d300cc5ca52b/db`. This directory contains the `monitored_vms`, `server_pool`, and `server_pool_servers` files.
  - To display the content of `db` files:

    `# ovs-agent-db dump_db -c <file name>`

Example:

```
[root@ovsvr01 db]# ovs-agent-db dump_db  -c server_pool_servers
{'ovsvr01.example.com': {'node_number': 0,
                         'registered_ip': '192.0.2.101',
                         'roles': set(['utility', 'xen'])}}
[root@ovm2 ~]#
```


Server Pool File System `db` Files

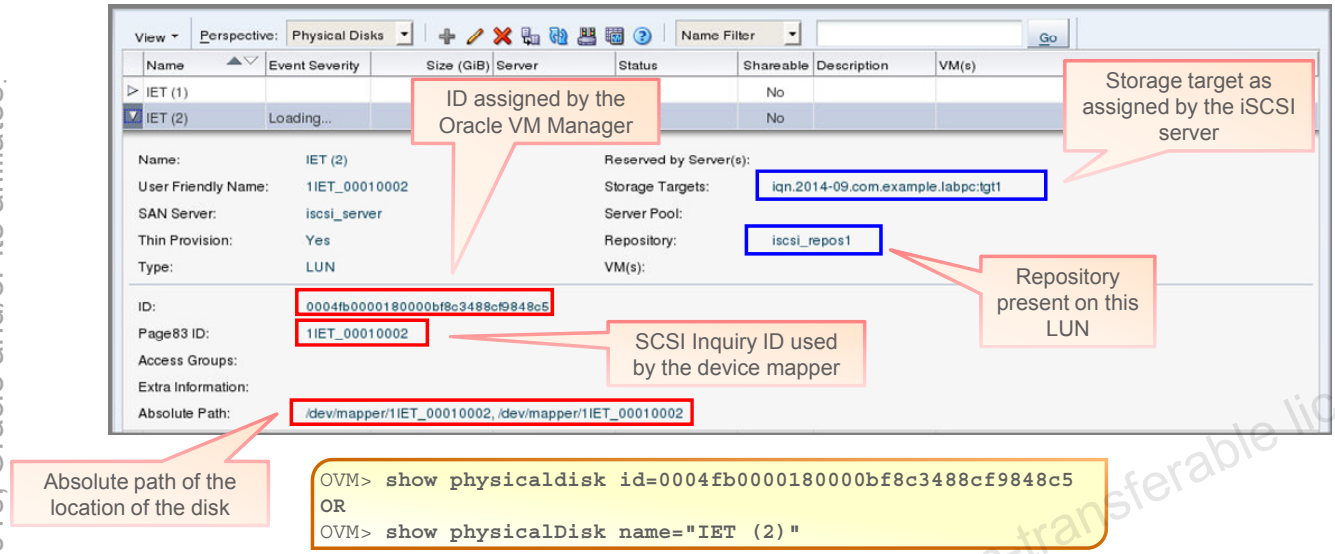
- **`monitored_vms`:** Contains information about the running virtual machines that have high availability enabled. This information is kept on all Oracle VM servers, which allows Oracle VM to restart a virtual machine with high availability enabled even if the Oracle VM Manager is not running.
- **`server_pool`:** Contains information about the server pool to which the Oracle VM server belongs
- **`server_pool_servers`:** Contains the list of Oracle VM servers in the server pool, their node number, their IP address and their server roles

As shown in the example in the slide, use the `-c` option to display information from the server pool file system `db` files.

Note

- Except for the information in the `monitored_vms` file, the information contained in the server pool file system `db` files is not always updated when changes occur. Use the Oracle VM Manager UI or the Oracle VM CLI to obtain the most up-to-date information about the state of your environment.
- You must not modify the information in the `db` files, except when directed by Oracle support.

Storage Information from the Oracle VM Manager UI



You can examine the information about the iSCSI and FC LUNs configured to your SAN servers by selecting the Physical Disks Perspective for each SAN server in your Oracle VM environment.

Physical Disk ID

For each physical disk, you can obtain its ID, as assigned by the Oracle VM Manager. With this ID, you can display the disk information by using the Oracle VM CLI. Using the example in the slide, you get the following output:

```
OVM> show physicaldisk id=0004fb0000180000bf8c3488cf9848c5
Command: show physicaldisk id=0004fb0000180000bf8c3488cf9848c5
Status: Success
Time: 2016-09-21 15:14:53,107 UTC
Data:
  Device Name 1 = /dev/mapper/1IET_00010002
  Device Name 2 = /dev/mapper/1IET_00010002
  Storage Targets 1 = iqn.2014-09.com.example.labpc:tgt1
  Page83 ID = 1IET_00010002
  Absolute Path 1 = /dev/mapper/1IET_00010002
  Absolute Path 2 = /dev/mapper/1IET_00010002
  Server Reserved = No
  Shareable = No
```

```
    Size (GiB) = 56.0
  State = ONLINE
  Status = online
  Thin Provision = Yes
  Type = LUN
  User-Friendly Name = 1IET_00010002
  Vendor = IET
  File System 1 = 0004fb000005000078a65edf1ee2a3bd
[0004fb000005000078a65edf1ee2a3bd]
  Volume Group = 0004fb0000320000c85fc039d394f890
[Generic_iSCSI_Volume_Group]
  Storage Array = 0004fb00000900004a01114bd6edc206  [iscsi_server]
  Id = 0004fb0000180000bf8c3488cf9848c5  [IET (2)]
  Name = IET (2)
  Locked = false
OVM>
```

The ID assigned to a physical disk by the Oracle VM Manager is not used by the OS running in the Oracle VM server (in dom0) to track the physical disk. The Oracle VM server uses the Page83 ID to track the physical disk.

Page83 ID or SCSI Inquiry ID

The SCSI inquiry ID (or Page83 ID) is the unique SCSI identifier for the LUN. This identifier is particularly important when you can access a LUN by more than one path. The device mapper driver uses this Page83 ID to assign a unique `/dev/mapper` name for each LUN. In the next slide, you use commands to display storage information. This information often contains this unique SCSI identifier.

Absolute Path

The Absolute Path is the location of the disk on a Oracle VM server. The `/dev/mapper` LUN is symbolically linked to a block device in the `/dev` directory.

Storage Target

The storage target information shown in the top-right rectangle in the slide lists the iSCSI qualified name (IQN) identifier for the target, as assigned by the SAN server which owns the physical disk.

Storage Initiators

The storage initiator with access to the physical disk shown in the slide is not listed as part of the physical disk information.

Because this physical information is associated with an Oracle VM server, `ovsvr01` in this example, you can assume that the `ovsvr01` Oracle VM server is a storage client for the physical disk shown in the slide. Therefore, `ovsvr01` acts as an initiator to the targets of the SAN server where the physical disk is located.

You can display the initiators for a SAN server by accessing the Access Groups Perspective from the Storage tab of the Oracle VM Manager UI. You can also use the Oracle VM CLI to display initiator information.

Displaying Storage Information on Oracle VM Servers

- `showmount --export <NFS server IP/hostname>`

  Use this command to display a list of exported shares from an NFS server.
- `blkid`

  Use this command to quickly determine the usage for block devices.
- `multipath -ll`

  Use this command to display paths to iSCSI LUNs.
- `iscsiadm -m node`

  Use this command to display the iSCSI targets that have been discovered by the OS in the Oracle VM server.

Use the commands listed in the slide to help you determine the status of your storage configuration. You execute these commands from your Oracle VM servers. The commands shown in the slide do not alter the state of your storage. Do not use other options with these commands without investigating their effect, because other options might trigger new storage operations. You must perform all storage configuration from one of the Oracle VM management interfaces.

Accessible NFS Share Information

The `showmount` command is useful to display the shares available from an NFS server. With the `--export` (or `-e`) option, and providing the NFS server IP address or host name as a parameter, you display the shares that are exported from the NFS server. You use this command to troubleshoot NFS access problems.

If you have access to the NFS server, you can also use the `exportfs` command on the NFS server.

Block Device Information

Use the `blkid` command to display block device information. The output of the command contains:

- The device name as a special file with its driver type, for example, `/dev/sdc`

- If applicable, the device name created by the device mapper driver—for example, `/dev/mapper/0004fb0000180000098507877ff36a45b`
- The UUID for the block device
- If applicable, the type of usage for the block device—for example, file system type or swap
- If available, the label for the file system on the device—for example, `OVS_POOL_FILESYSTEM` or `OVS65edf1ee2a3bd`. The label with the `OVS_POOL_FILESYSTEM` string indicates the server pool file system block device and the label with the OVS string indicates a repository on a block device.

Multipath Information

Use the `multipath -ll` command to display multipathing information for your physical devices, which is your iSCSI and FC LUNs. For each `/dev/mapper` device, the `multipath` command displays its multipathing ID (which is also its Page83 ID), and the paths to this device. If a device has more than one path, you can find out which paths are active. iSCSI multipathing is discussed in the next slide.

Warning: The `multipath` command can be dangerous if executed without options because the command then attempts to configure multipath devices. Always use the **`-l`** or **`-ll`** option when invoking the `multipath` command from an Oracle VM server.

iSCSI Information

Use the `iscsiadm -m node` command to display the iSCSI devices or targets that have been discovered and that have established sessions.

Note: This is a very useful command to troubleshoot iSCSI problems. However, do not use other forms of the command, which could initiate a target discovery or establish/remove sessions to the targets. All discovery and login operations to targets must take place from the Oracle VM Manager as part of the SAN server configuration and refresh operations.

iSCSI Multipathing on Oracle VM Servers

- When configuring an iSCSI storage array, you can specify two access hosts to the storage server.
  - Select two IPs on different subnets.
- If the active path fails, the other path becomes active.
- Use the `multipath -ll` command to display the path status from the Oracle VM server:

```
[root@ovsvr01 ~]# multipath -ll
...
1IET_00010002 dm-3 IET,VIRTUAL-DISK
size=56G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
| `- 2:0:0:2 sde 8:64 failed faulty running      <- Failed Path
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 4:0:0:2 sdh 8:112 active ready running     <- Active Path
```


Configuring iSCSI Multipathing When Creating a SAN Server

When you configure multipathing for iSCSI (during SAN server configuration), you select two IP addresses on different subnets. If you select two IPs in the same subnet, to the same iSCSI server, only the top IP address in the routing table is used unless you set up special routing to direct the traffic. This is due to the way Linux networking works rather than to iSCSI. Therefore, there is no iSCSI multipath operation if the two selected IP addresses are in the same subnet.

Network Path Failure to iSCSI LUNs

If one of the paths to iSCSI LUNs fails, the multipathing daemon fails the path as seen in the `/var/log/messages` log file on the Oracle VM server:

```
Sep 20 18:13:12 ovsvr01 multipathd: 8:32: mark as failed
Sep 20 18:13:12 ovsvr01 multipathd: 1IET_00010001: remaining active paths: 1
...
Sep 20 18:13:13 ovsvr01 multipathd: checker failed path 8:64 in map 1IET_00010002
Sep 20 18:13:13 ovsvr01 multipathd: 1IET_00010002: remaining active paths: 1
```

Depending on the activity on the LUNs, you might see the change in path status by using the `multipath -ll` command:

```
[root@ovsvr01 ~]# multipath -ll
1IET_00010001 dm-4 IET,VIRTUAL-DISK
size=13G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=enabled
| `- 2:0:0:1 sdc 8:32  failed faulty running
`-+- policy='round-robin 0' prio=1 status=active
  `- 4:0:0:1 sdg 8:96  active ready  running
1IET_00020001 dm-0 IET,VIRTUAL-DISK
size=13G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
| `- 3:0:0:1 sdb 8:16  failed faulty running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 5:0:0:1 sdj 8:144 active ready  running
...
1IET_00010002 dm-3 IET,VIRTUAL-DISK
size=56G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
| `- 2:0:0:2 sde 8:64  failed faulty running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 4:0:0:2 sdh 8:112 active ready  running
[root@ovsvr01 ~]#
```

The first numbers associated with a device (for example, `2:0:0:2`) represent `host:channel:id:lun` (where `id` is the target ID). The second set of numbers (for example, `8:64`) represent the major and minor numbers for the device.

When the path to the device is restored, entries appear in the `messages` file:

```
Sep 20 18:45:06 ovsvr01 iscsid: connection1:0 is operational after recovery
(9 attempts)
Sep 20 18:45:14 ovsvr01 multipathd: 1IET_00010002: sde - directio checker
reports path is up
Sep 20 18:45:14 ovsvr01 multipathd: 8:64: reinstated
Sep 20 18:45:10 ovsvr01 multipathd: 1IET_00010001: remaining active paths:
2
Sep 20 18:45:14 ovsvr01 multipathd: 1IET_00010002: sde - directio checker
reports path is up
Sep 20 18:45:14 ovsvr01 multipathd: 8:64: reinstated
Sep 20 18:45:14 ovsvr01 multipathd: 1IET_00010002: remaining active paths: 2
```

Oracle VM Server Health Check

- If the Oracle VM server is part of a clustered server pool, make sure that the pool server file system is mounted.
- Display the `ovs-agent` service status.
- Display the `o2cb` service status (for a clustered server pool).
- Check the list of running virtual machines.
- Run the VMPinfo3 tool.

To Check the Status of the Server Pool File System

Usually, you take this step because of an error condition noted from the Oracle VM Manager. On the Oracle VM server that might be experiencing problems, issue the `df -h` command to ensure that the server pool file system is mounted.

Note: Only clustered server pools have a server pool file system.

Display the `ovs-agent` Service Status

From the Oracle VM server being examined, issue the following command:

```
# service ovs-agent status
log server (pid 2598) is running...
notificationserver server (pid 2925) is running...
remaster server (pid 2936) is running...
monitor server (pid 2937) is running...
ha server (pid 2938) is running...
stats server (pid 2940) is running...
xmlrpc server (pid 2944) is running...
fsstats server (pid 2946) is running...
apparentsize server (pid 2947) is running...#
```

All processes must be running.

Note: You can stop and restart the Oracle VM Agent without affecting the cluster or the running virtual machines on the Oracle VM server.

Check the List of Running Virtual Machines

You can obtain this list from:

- The Oracle VM Manager UI, by selecting a server pool or a specific Oracle VM server in the navigation pane, and selecting the Virtual Machines perspective in the management pane
- The Oracle VM CLI, by using a combination of commands. First, use `show server name=<server name>` to display the list of virtual machines for a specific Oracle VM server, and use the `show vm name=<virtual machine name>` command to display the status of the virtual machines on that server.

  If using the Oracle VM CLI, you might want to write a script to parse the output of each `show vm` command to extract the status information.
- The Oracle VM WS-API.

  The following code extract example uses the REST API and Python:

```
import requests
s=requests.Session()
s.auth=('admin','Welcome1')
s.verify=False
s.headers.update({'Accept': 'application/json', 'Content-Type': \
    'application/json'})
baseUri='https://ovsvr01.example.com:7002/ovm/core/wsapi/rest'
r=s.get(baseUri+'/Vm')
for i in r.json():
    print '{name} is {state}'.format(name=i['name'], \
          state=i['vmRunState'])
```

Note: Replace `ovsvr01.example.com` with the host name of your Oracle VM Manager host.

If virtual machines that are expected to be running are in a stopped state, you might be experiencing memory problems, or some Oracle VM servers might be offline and contributing to the resource shortage.

Running the VMPinfo3 Tool from Oracle VM Manager

You learn about the VMPinfo3 tool in the next slide.

Gather Troubleshooting Information About Your Oracle VM Environment

- Run the VMPinfo3 tool to capture troubleshooting information for Oracle support.
- You run the tool from Oracle VM Manager.
- The tool is located in the `/u01/app/oracle/ovm-manager-3/ovm_tools/support/` directory.
- Examples:
  - To obtain help information:

```
# ./vmpinfo3.sh --help
```

  - To list Oracle VM servers:

```
# ./vmpinfo3.sh --username=admin listservers
```

  - To run the tool for one Oracle VM server:

```
# ./vmpinfo3.sh --username=admin servers=ovsvr01.example.com
```


VMPinfo3 is a tool for capturing logs, configuration information, XML model database dumps, and other information for your Oracle VM Manager host and Oracle VM servers.

The data collected by the VMPinfo3 tool includes:

- An `sosreport` for all the Oracle VM servers selected when running the tool
- An `sosreport` for the Oracle VM Manager
- All of the Oracle VM Manager logs

Note: The `sosreport` utility is part of the Oracle Linux distribution.

The `sosreport` output includes (partial list):

- Configuration information for the target server (Oracle VM Manager or Oracle VM server), such as `ntp.conf`, `iptables`, `xinetd`
- A list of all packages installed
- Networking information
- Hardware information

Running the VMPinfo3 Tool from Oracle VM Manager

The tool is already installed in your Oracle VM Manager host.

The tool has one script, called `vmpinfo3.sh`, that you execute from the Oracle VM Manager host.

Run the following commands to get the troubleshooting information for all Oracle VM servers:

1. Access the directory where the VMPinfo3 tool is located.

```
# cd /u01/app/oracle/ovm-manager-3/ovm_tools/support/
```

2. Run the `vmpinfo3.sh` command with the `listservers` option to get the list of available Oracle VM servers:

```
# ./vmpinfo3.sh --username=admin listservers
Enter OVM Manager Password: MyOracle1
The following server(s) are owned by this manager:
['ovsvr01.example.com', 'ovsvr02.example.com']
```

3. Based on the output from the previous command, run `vmpinfo3.sh` on selected servers.

   Example:

```
# ./vmpinfo3.sh --username=admin
servers=ovsvr01.example.com,ovsvr02.example.com
```

Specify the Oracle VM Manager username, `admin`, as an option to the `vmpinfo3.sh` program. The program prompts you for the password for `admin`.

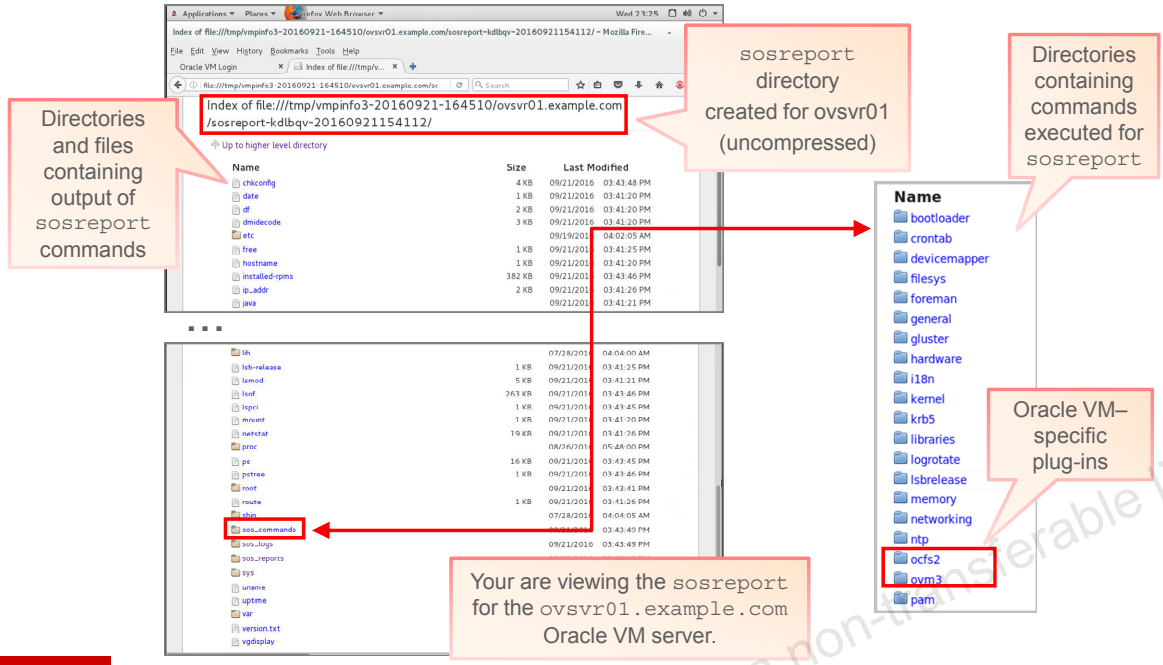
The compressed file created by the VMPinfo3 tool is located in the `/tmp` directory on the Oracle VM Manager host.

Example of running the VMPinfo3 tool for one Oracle VM server only:

```
[root@ovmmgr01 support]# #./vmpinfo3.sh --username=admin
servers=ovm1.example.com
Enter OVM Manager Password: Welcome1 ← You are prompted for the admin's password

Gathering files from servers: ovsvr01.example.com  This process may take
some time.
The following server(s) will get info collected: [ovsvr01.example.com]
Gathering OVM Model Dump files
Gathering sosreport from ovsvr01.example.com
Gathering OVM Manager Logs
Compressing VMPinfo3 20160921-164510
================================================================================
 Please send /tmp/vmpinfo3-3.4.1.1369-20160921-164510.tar.gz to Oracle
support
================================================================================
[root@ovmmgr01 support]#
```

Viewing the VMPinfo3 Report



Oracle support uses the raw snapshot of your Oracle VM configuration produced by the VMPinfo3 tool to troubleshoot problems in your environment.

After you run the VPMinfo3 tool, you can make the resulting compressed TAR file available to Oracle support by:

- Uploading the output file to `sftp.oracle.com` by using the `sftp` command

OR

- Accessing My Oracle Support and uploading the output file by attaching it to your service request

You can, however, use the VMPinfo3 tool to capture a snapshot of your configuration and use it to document the current state of your Oracle VM environment. For example, you can run the VMPinfo3 tool before and after major changes, like an upgrade to the software of your Oracle VM Manager and Oracle VM servers.

You can browse the `sosreport` information created for the Oracle VM Manager and any Oracle VM server selected when you ran the VMPinfo3 tool. To view this information:

- Copy the output file created by the VMPinfo3 tool to a host with a web browser.
- Uncompress the output file from the VMPinfo3 tool.
- Locate the `sosreport` directory created by the `sosreport`.

- Uncompress the `sosreport` file.
- Use the browser to access the `sosreport` directory structure.

The diagrams in the slide show the `sosreport` directory structure for the `ovm1.example.com` Oracle VM server.

`sosreport plugins`

The `sosreport` command uses `plugins` to gather information about the system under examination.

Note the list of directories in the `sos_commands` directory. This list is shown in the smaller screenshot in the slide. The directory names correspond to the `plugins` that are enabled for the `sosreport` command, and each directory contains the files and other information gathered by each `plugin`. Also note the `ovm3` and `ocfs2 plugins` that gather important information about your Oracle VM servers.

You can execute the `sosreport` command from the command line of any of your Oracle VM servers.

VMPinfo3 Resources

For more information about the VMPinfo3 tool, access My Oracle Support and locate Doc ID 1521931.1 titled “VMPinfo3 Diagnostic Tool For Oracle VM 3.2 and 3.3 Troubleshooting”.

For more information about the `sosreport` utility, read the section titled “About `sosreport`” of the “Support Diagnostic Tools” chapter in the *Oracle Linux Administrator’s Solutions Guide*, Part Number E37355-32 or later.

Kdump for Oracle VM Servers

- Kdump is the Linux kernel crash/dump mechanism.
- If your system crashes, Kdump creates a memory image (`vmcore`) of your system.
- You can enable Kdump at installation time.
- When configuring Kdump, you reserve a portion of system memory for use by Kdump.
  - This memory is not available for other uses.
- You can manually configure Kdump and its associated service after installation.

If your Oracle VM server encounters a system-wide problem, such as a crash or unexpected reboots, a system memory dump (or `vmcore`) is a very useful tool for problem analysis. By default, the required packages to enable the Kdump service are included within the Oracle VM server installation, so you choose to enable Kdump and specify an amount of memory to reserve for Kdump operations during the installation of Oracle VM Server for x86. You can, however, manually configure Kdump for your Oracle VM server after installation. The topic “Manually Configuring Kdump for Oracle VM Server” in the *Oracle VM Administration Guide* (E64083-01 or later) describes the actions required to properly configure your Oracle VM server.

The following slides discuss how to manually configure Kdump.

Configuring Kdump on Oracle VM Servers

- Reserve memory in `/boot/grub/grub.conf` for Kdump operations and reboot.
- Set the location for `vmcore` in `/etc/kdump.conf`.
  - For example, specify `/dev/sdb1` or `/dev/mapper/OVM_SYS_REPO_PART_3500000e012e70af0`.
- Prepare the Kdump partition on the selected disk.
- Enable the Kdump service to start at boot time.
  - Also start the service manually.
- Test Kdump:
  - Force a system crash.
  - After reboot, verify that the `vmcore` file has been created.

The Oracle VM server installer provides an option to enable Kdump at installation where many of these steps are performed automatically. The necessary packages for Kdump operations are included in the Oracle VM Server for x86 software. You can verify that the proper packages are installed by using the following command:

```
# rpm -qa | egrep 'kexec-tools|busybox'
kexec-tools-ovs-2.0.7-1.100.8.el6.x86_64
busybox-1.15.1-21.el6_6.x86_64
#
```

Reserve Memory for Kdump

Note: All steps are carried out on the Oracle VM server where Kdump is configured.

- Add "`crashkernel=256M`" on `GRUB_CMDLINE_XEN=` parameter in the Oracle VM server's `/etc/default/grub` file. 256M is the amount of space used in memory to load the crash kernel that will run and generate the dump:

```
...
GRUB_CMDLINE_XEN="dom0_mem=max:800M allowsuperpage crashkernel=256M"
...
```

- Reboot the Oracle VM server.
- Check that the memory has been reserved by examining the boot messages and reserved memory:

```
# xm dmesg | grep -i crashkernel
(XEN) Command line: console=com1,vga com1=57600,8n1 dom0_mem=max:1104M
allowsuperpage crashkernel=256M@16M
# cat /proc/iomem | grep "Crash kernel"
  01000000-08ffffff : Crash kernel
```

Set the Location for the Dump File

In `/etc/kdump.conf`, specify the partition, raw disk, or path to a directory where you want the `vmcore` files to be written.

By default, the `root` file system installed on the Oracle VM server is too small to contain a `vmcore` file. You must specify a different location for storing dump files.

For example, specify a second internal disk in the Oracle VM server, or specify the device created on the installation disk with the space not used during the installation.

Spare Partition

During the installation of the Oracle VM Server for x86 software, four partitions are defined: a root, boot, swap, and spare partition. You can use the spare partition as:

- An Oracle VM repository
- A dedicated disk for a virtual machine
- A space to write `vmcore` dumps

A device is created automatically for the spare partition. This device has a name containing the `OVM_SYS_REPO_PART` string. In this example, the `/dev/mapper/OVM_SYS_REPO_PART_3500000e012e70af0` device is used as space for storing dumps.

Create the ext4 File System

The spare partition is created automatically if there is enough leftover space on the installation disk (starting with Oracle VM 3.3.x releases).

If you are using another internal disk to store the `vmcore` files, create a partition on the disk. You can use the raw partition as a dump location or you can create a file system on the partition.

For example, to use a file system on partition `/dev/sdb1` as a dump location, create an ext4 file system on it:

```
# mkfs.ext4 /dev/sdb1
```

Identify the UUID for the File System

Use the `blkid` command to display the UUID for the file system created on the partition. For example:

```
# blkid /dev/mapper/OVM_SYS_REPO_PART_3500000e012e70af0
/dev/mapper/OVM_SYS_REPO_PART_3500000e012e70af0: UUID="9023fea0-
a0ca-4829-9f01-c5718ec2e934" TYPE="ext4"
```

Update the `kdump.conf` File

You must identify the location where the `vmcore` files are to be written during a dump operation. For example, add the information for the file system on the spare partition as the location to store `vmcore` dumps.

```
...
ext4 UUID=9023fea0-a0ca-4829-9f01-c5718ec2e934
#raw /dev/sda5
#ext4 /dev/sda3
...
```

Enable the Kdump Service

- Enable the `kdump` service by using the `chkconfig` command:

```
# chkconfig kdump on
```

- Restart the `kdump` service now:

```
# service kdump restart
Stopping kdump:                                            [  OK  ]
Detected change(s) the following file(s):

  /etc/kdump.conf
Rebuilding /boot/initrd-3.8.13-26.4.2.el6uek.x86_64kdump.img
Starting kdump:                                            [  OK  ]
```

Test Kdump Operations

- Trigger a system crash:

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

- After the Oracle VM server has rebooted, mount `/dev/mapper/OVM_SYS_REPO_PART _3500000e012e70af0` on a temporary mount point and check for the presence of a `vmcore` file in the `var/crash/<`*`dump directory name`*`>` directory.

netconsole for Capturing Panic Messages

- netconsole is a kernel module that forwards kernel log messages over the network to a remote host.
  - The netconsole service activates the netconsole module.
- Use the netconsole service to debug kernel panics.
- To set up netconsole:
  - Configure and activate the netconsole service on your Oracle VM server
  - Configure logging on a remote host to accept messages from the netconsole module on your Oracle VM server

netconsole is a kernel module that, when loaded, is active even when user space is experiencing problems. The netconsole service loads the netconsole module. For normal operations, you do not have to start the netconsole service on your Oracle VM servers. Use netconsole only to debug kernel panics, when the panic messages do not appear in the messages log of your Oracle VM server.

To set up netconsole, you configure and activate the netconsole service on your Oracle VM server and configure a remote host to receive the messages sent by the netconsole service on your Oracle VM server. You can send the netconsole messages from several Oracle VM servers to one central remote host.

Steps for Configuring `netconsole` for Oracle VM Servers

On your Oracle VM server:

- Configure an appropriate log level for kernel messages
- Edit the `/etc/sysconfig/netconsole` file
  - Select the network interface for sending log messages.
  - Specify an IP and a MAC address for the remote host.
- Set the `netconsole` service to start at boot time by using the `chkconfig` command

On the Oracle VM server, perform the following tasks:

- Confirm that you have a free `eth(x)` device that is not part of a bond or a Xen bridge on Dom0.
- Increase the logging level for kernel messages:
  - Set the console log level for kernel messages in `/etc/sysctl.conf`

    ```
    # echo 'kernel.printk = 9 4 1 7'  >> /etc/sysctl.conf
    ```

    The first number is the console log level. A high number corresponds to a low priority. All messages with a priority higher than this number are logged to the console.
  - Apply the change

    ```
    # sysctl -p
    ```
- Edit the `/etc/sysconfig/netconsole` file:
  - Select the network interface through which the kernel messages will be sent.
  - You can select the `bond0` (management) interface if there is no bridge configured on the bond. A bridge is present if the management network has the Virtual Machine network channel defined.

- Specify the IP address and the MAC address for the remote host, and the local and remote port numbers to use for the transfer of messages.

  For example:

```
# LOCALPORT=6666   ← default
DEV=bond0
SYSLOGADDR=10.150.36.209
# SYSLOGPORT=514   ← default
SYSLOGMACADDR=00:14:4F:78:55:68
```

Note: In some cases, the `DEV` and `SYSLOGMACADDR` parameters are not required.

- Start the `netconsole` service:

```
# service netconsole start
```

- Set the `netconsole` service to start at boot time:

```
# chkconfig netconsole on
```

Steps for Configuring `netconsole` on a Remote Host

On the host at the remote location:

- If the firewall is enabled on the remote host, open port 514 (UDP and TCP) in the firewall
- Configure `rsyslog` (in `/etc/rsyslog.conf`) to receive messages from remote locations
- Restart the `rsyslog` service
- Test by sending kernel messages from the Oracle VM server to the remote host

Example of Remote Host Configuration for Oracle Linux 6

On the remote host, configure the `rsyslog` service to accept incoming messages from one or more of your Oracle VM servers.

- Edit the `/etc/rsyslog.conf` file and uncomment the following lines:

```
#$ModLoad imudp
#$UDPServerRun 514
#$ModLoad imtcp
#$InputTCPServerRun 514
```

- Restart the `rsyslog` service:

```
# service rsyslog restart
```

After setting up the Oracle VM server and the remote host, verify that `netconsole` is working properly.

Test 1:

- On the Oracle VM server, send a message to the `/dev/kmsg` device:

```
# echo "This is message 1 using /dev/kmsg" > /dev/kmsg
```

- On the remote server, display the log messages:

```
# tail -f /var/log/messages
...
Aug 21 12:47:50 This is message 1 using /dev/kmsg
...
```

Test 2:

- On the Oracle VM server, send a request to the `/proc/sysrq-trigger` file:

```
# echo m > /proc/sysrq-trigger
```

The preceding system request (`m`) causes memory information to be displayed to the console and does not produce any change to the Oracle VM server.

- On the remote host, display the log messages from `ovm4`, the Oracle VM server used in this example:

```
# tail -f /var/log/messages
...
Aug 21 14:34:59 SysRq :
Aug 21 14:34:59 Show Memory
Aug 21 14:34:59 ovm4 Mem-Info:
Aug 21 14:34:59 Node 0
Aug 21 14:34:59 DMA per-cpu:
Aug 21 14:34:59 CPU    0: hi:    0, btch:   1 usd:   0
Aug 21 14:34:59 CPU    1: hi:    0, btch:   1 usd:   0
Aug 21 14:34:59 CPU    2: hi:    0, btch:   1 usd:   0
Aug 21 14:34:59 CPU    3: hi:    0, btch:   1 usd:   0
...
Aug 21 14:34:59 ovm4 = 681888kB
Aug 21 14:34:59 29921 total pagecache pages
Aug 21 14:34:59 0 pages in swap cache
Aug 21 14:34:59 Swap cache stats: add 0, delete 0, find 0/0
Aug 21 14:34:59 Free swap  = 3850236kB
Aug 21 14:34:59 Total swap = 3850236kB
Aug 21 14:34:59 282709 pages RAM
Aug 21 14:34:59 25813 pages reserved
Aug 21 14:34:59 1089061 pages shared
Aug 21 14:34:59 65211 pages non-shared
```

Core Dumps for Virtual Machines

- For HVM or PVHVM guests running UEK kernels or for any PVM guest:
  - Edit the virtual machine's `vm.cfg` file and change the `on_crash` setting:

```
on_crash='coredump-destroy'
on_crash='coredump-restart'
```

  - Restart the guest
    - If a crash occurs, a `vmcore` file is created in the `/var/xen/dump/<UUID of virtual machine>` directory.
- For guests running the Red Hat Compatible Kernel (RHCK) with Oracle Linux 6.3 or newer:
  - Configure Kdump in the guest

Earlier in this lesson, you learned how to configure the Kdump feature on your Oracle VM servers. The Kdump feature is also available with Oracle Linux guests running a Red Hat Compatible Kernel (RHCK).

For your particular Linux distribution, consult the documentation for information about obtaining a `vmcore` (core dump) file for your virtual machines.

For HVM and PVHVM guests running Oracle Linux with a UEK kernel, and for any PVM guest, you can obtain a dump of a virtual machine's memory content by modifying the `on_crash` option in the `vm.cfg` of the virtual machine to one of the following setting:

- `on_crash='coredump-restart'`: This setting generates a core dump and the virtual machine is automatically restarted.
- `on_crash='coredump-destroy'`: This setting generates a core dump but the virtual machine is not restarted.

The following options are also supported for the `on_crash` setting, but they do not trigger a core dump when the virtual machine crashes:

- `on_crash='preserve'`: With this setting, the virtual machine freezes when crashing. You might still be able to access the last console messages for troubleshooting purposes.

- `on_crash='restart'`: This setting causes the virtual machine to reboot automatically when terminating abnormally.

Note

- If you edit the virtual machine from the Oracle VM Manager after making changes to the `on_crash` setting, the updated `on_crash` setting is lost.
- The `on_crash` setting is ignored for HVM and PVHVM guests using the Red Hat Compatible Kernel (RHCK).

Where Are Core Dump Files Stored?

Core dump files are stored in the `/var/xen/dump` directory on the Oracle VM server where the virtual machine is running. The `vmcore` file name contains the UUID of the virtual machine and a time stamp. This file is stored in a subdirectory in `/var/xen/dump` whose name matches the UUID of the virtual machine.

Make sure that there is enough space in the `/var/xen/dump` directory. The `vmcore` files can be large, up to the size of the memory assigned to the virtual machine. The `vmcore` files are never deleted automatically.

You cannot reconfigure the `/var/xen/dump` directory. However, you can create an NFS share and mount it under `/var/xen/dump`.

Transferring core dump files over the network requires adequate bandwidth. Use a dedicated network; otherwise, the transfer might interfere with cluster operation in the server pool to which the Oracle VM server belongs.

Triggering a Core Dump Manually

- You can trigger a core dump for a virtual machine that is running on any Oracle VM server.
- For all guests types, use the `xm dump-core <domain ID>` command from the Oracle VM server where the virtual machine is running.
- The core dump file is created by default in the `/var/xen/dump` directory on the Oracle VM server where the virtual machine is running.
  - You can specify an alternate location for the core dump file. Example:

```
# xm dump-core 0004fb00000600009c727c757e759faa \ /tmp/myCoreDump
```


To obtain a core dump for any guest, use the `xm dump-core` command from the Oracle VM server where the virtual machine is running.

Example:

1. From the Oracle VM Manager UI or Oracle VM CLI, find the virtual machine UUID. The Xen domain ID matches the virtual machine UUID.
2. You can verify that the guest is running on the target Oracle VM server (`ovm1` in this example) by using the `xm list` command:

```
[root@ovm1 ~]# xm list
Name                                        ID   Mem  VCPUs   State    Time(s)
0004fb0000060000210e604975b14eb3             6  3771    2     -b----      83.7
0004fb00000600009c727c757e759faa             9  1024    1     -b----    1108.3
0004fb0000060000e32f93ec3d508574             1  1024    2     -b----     759.2
Domain-0                                     0   672    4     r-----  434553.3
[root@ovm1 ~]#
```

3. Trigger a core dump with the `xm dump-core` command and specify the *domain ID* or *domain name* for the virtual machine:

Example 1: Using the default location for the core dump file

```
[root@ovm1 ~]# xm dump-core 0004fb00000600009c727c757e759faa
Dumping core of domain: 0004fb00000600009c727c757e759faa ...
```

Example 2: Using an alternate location for the core dump file

```
[root@ovm1 ~]# xm dump-core 0004fb00000600009c727c757e759faa
/tmp/myCoreDump
Dumping core of domain: 0004fb00000600009c727c757e759faa ...
```

Note: Manually triggering a core dump for a virtual machine does not stop or crash the virtual machine.

Location for Core Dump Files

You can find the core dump file for the first example in a subdirectory of the `/var/xen/dump` directory:

```
[root@ovm1 ~]# ls -l /var/xen/dump/
total 4
drwx------ 2 root root 4096 Aug 21 18:15 0004fb00000600009c727c757e759faa
[root@ovm1 ~]# ls -l /var/xen/dump/0004fb00000600009c727c757e759faa
total 1054708
-rw------- 1 root root 1080016896 Aug 21 18:15 2014-0821-1815.21-
0004fb00000600009c727c757e759faa.9.core
[root@ovm1 ~]#
```

The core file for the second example is located in the `/tmp` directory:

```
[root@ovm1 ~]# ls -l /tmp/my*
-rw------- 1 root root 1080016896 Aug 21 18:23 /tmp/myCoreDump
[root@ovm1 ~]#
```

Console Access for Linux Guests

- If the serial console access for your virtual machine is enabled, you can use the Oracle VM Manager serial console feature.
- If you can no longer access your virtual machine by using the Oracle VM Manager console feature, use the `xl console` command for troubleshooting purposes.
- You must configure serial console access in your virtual machine before you can access its console with the `xl console` command.

If your virtual machine is experiencing problems, you might no longer be able to access the virtual machine by using the Oracle VM Manager VNC-type console feature. Similarly, if the Oracle VM Manager is not running, you cannot use the Oracle VM Manager console feature at all.

If you have configured your virtual machine to support serial console access, you can use the `xl console <domain ID>` command to access your virtual machine's console. You issue this command from the Oracle VM server where the virtual machine is currently running.

After connecting to your virtual machine, you can use a variety of tools for troubleshooting purposes. For example, you might want to bring your virtual machine to single-user mode to use the `fsck` command to clean up a corrupted file system that keeps the virtual machine from booting normally.

One advantage of configuring your virtual machines for serial console access is that you can use the serial console feature with the Oracle VM Manager UI in addition to the VNC console feature.

When you are using the serial console feature with the `xl console` command from the Oracle VM server, you can use the cut and paste feature to capture information from the virtual machine's console.

Configuring Serial Console Access for Oracle Linux Guests

- Find out if a serial console is already active in your virtual machine:

  `# dmesg | grep`
  - For HVM guests, the serial console device has the form `/dev/ttySx` where *x* is usually `0`.
  - For PVM guests, the serial console device is `/dev/hvc0`.
- Steps to configure the console access might include:
  - Adding a `console` parameter to the kernel command line in the GRUB configuration
  - Configuring `getty` to set up a serial terminal connection
  - Editing `/etc/securetty` to allow the root user to log in to the serial console

The steps to configure serial console access are different for HVM and PVM guests. Moreover, the steps to configure the serial console often changes from one release of Oracle Linux to the next. For other Linux flavors, consult the document for your particular Linux distribution.

Serial Console Configuration for HVM Guests (for Oracle Linux 6 guests)

- Modify the `/boot/grub/grub.conf` file in your virtual machine.

  Append `console=ttyS0,9600n8` to the kernel line of the default boot title:

```
...
 kernel /boot/vmlinuz-2.6.9-78.0.8.2.1.EL ro root=LABEL=/
console=ttyS0,9600n8
```

- Create the `/etc/init/serial-ttyS0.conf` file with the following content:

```
start on stopped rc RUNLEVEL=[2345]
stop on runlevel [016]
instance ttyS0
respawn
pre-start exec /sbin/securetty ttyS0
exec /sbin/agetty /dev/ttyS0 9600 vt100-nav
```

- Modify the `/etc/securetty` file by adding an entry for the `ttyS0` device:

```
console
ttyS0
vc/1
vc/2
...
```

- Verify that the `serial='pty'` parameter/value is present in the guest's configuration file (`vm.cfg`):

```
...
kernel = '/usrlib/xen/boot/hvmloader'
maxmem = 2048
memory = 2048
name = 'HVM_OEL6U4'
on_crash = 'restart'
on_reboot = 'restart'
pae = 1
serial = 'pty'
...
```

Serial Console Configuration for PVM Guests (for Oracle Linux 6 guests)

- Modify the `/boot/grub/grub.conf` file:
  - By adding a `terminal and serial` lines for the serial console
  - By appending `console=hvc0` to the kernel line of the default boot title

Example with Oracle Linux 6.4:

```
default=0
timeout=5
serial --unit=0 --speed=9600
terminal --timeout=5 serial console
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Oracle Linux Server Unbreakable Enterprise Kernel (2.6.39-
400.17.1.el6uek.x86_64)
            root (hd0,0)
      kernel /vmlinuz-2.6.39-400.17.1.el6uek.x86_64 ro
root=/dev/mapper/vg_iscsipvm1-lv_root rd_NO_LUKS LANG=en_US.UTF-8
rd_LVM_LV=vg_iscsipvm1/lv_swap rd_NO_MD SYSFONT=latarcyrheb-sun16
rd_LVM_LV=vg_iscsipvm1/lv_root  KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM
rhgb quiet console=hvc0
      initrd /initramfs-2.6.39-400.17.1.el6uek.x86_64.img
title Oracle Linux Server Red Hat Compatible Kernel (2.6.32-
358.el6.x86_64)
    root (hd0,0)
...
```

- Modify the `/etc/securetty` file by adding an entry for the `hvc0` device:

```
...
tty7
tty8
tty9
tty10
tty11
hvc0
```

After making modifications to your virtual machine, reboot it. You can now access it by using the `xl console <domain ID>` from the Oracle VM server where the virtual machine is running. Find the domain ID for your virtual machine by issuing the `xl list` command.

To exit the serial console, press the Crt +] key combination.

For more information about the procedure to configure the serial console access for your virtual machines, including instructions for older releases of Oracle Linux, consult Doc ID 579413.1, titled “Oracle VM: How to Configure ‘xm console’ Access for Guests” in the My Oracle Support site. This document includes instructions for configuring `xm console` access for Oracle Linux 7 guests.

Quiz

Which statements are true about the VMPinfo3 tool?

a. This tool captures troubleshooting information for the Oracle VM Manager and selected Oracle VM servers.

b. You obtain this tool from My Oracle Support.

c. You must install special plug-ins before running this tool.

d. The output of this tool is valuable as a troubleshooting tool for Oracle support and for the Oracle VM environment administrators.

Answer: a, d

Summary

In this lesson, you should have learned how to:

- Locate the major Oracle VM logs and discuss their usage
- Execute commands to examine the status of the major components of Oracle VM
- Describe the use for the `vmpinfo` and `sosreport` tools
- Configure Kdump for Oracle VM servers
- Configure `netconsole` for Oracle VM servers
- Produce core dumps for virtual machines
- Access the serial console for Linux guests

Practice Overview

In this practice, you:

- Locate and view important files and logs
- Run commands to assess the status of storage, iSCSI connections, and OCFS2 clusters
- Run the VMPinfo3 tool from the Oracle VM Manager host and examine the tool's output
- Return your environment to a single server pool configuration
- Configure the serial console for a PVM guest and access the serial console by using the `xl console` command

7

Backup and Restore
D/R Concepts

Objectives

After completing this lesson, you should be able to:

- Identify the Oracle VM components to be backed up
- Discuss the guidelines for backing up the components
- Perform Oracle VM Manager database backups and restores
- Recover your Oracle VM environment
- Discuss Disaster/Recovery (D/R) concepts

Lesson Overview

- The first section of this lesson introduces the guidelines for backing up and restoring various components of your Oracle VM environment.
- The topics in this first section include:
  - What components must be part of your backup strategy
  - What elements to back up for each component and their backup frequency
  - How to back up and restore these components
- The second section presents recovery scenarios for your Oracle VM environment.

Oracle VM Environment

The Oracle VM environment comprises:

- Multiple software components
- A networking infrastructure
- External storage
- Virtual machines that depend on the underlying components, networking, and storage resources

The Oracle VM environment is a complex mix of software components, networking, and storage infrastructure supporting multiple virtual machines running across multiple Oracle VM servers that are grouped into server pools.

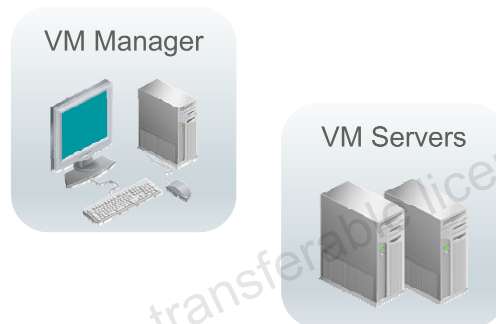
When creating a backup strategy, the logical starting point is to back up the various components of the Oracle VM environment, but this approach is not enough. You must also anticipate the situations where a restore is required. The restore operation might involve one or more components of your Oracle VM environment. Therefore, your backup strategy must encompass not only the protection for the components of your environment, but also a process to evaluate which components must be restored under various failure scenarios.

For example, if a storage array fails, what elements in your environment are affected? What if a single disk in the storage array fails? What happens if the server pool file system for a server pool fails?

There is no single solution that is suitable for all sites. In this lesson, you learn the concepts that help you devise, test, and implement your own backup and restore strategies.

Oracle VM Components: Oracle VM Manager and Oracle VM Servers

- Oracle VM Manager:
  - Oracle VM Manager software
  - Configuration files
  - Database objects
- Oracle VM servers:
  - Oracle VM Server for x86 software
  - Configuration files
  - Server pool file systems
  - Repositories



Oracle VM Manager

The Oracle VM Manager is typically installed on a stand-alone host, either a physical or virtual server. Nearly all of the Oracle VM software and configuration files reside in the `/u01` directory. The MySQL database files reside in the `/u01` directory and the MySQL Enterprise Backup software resides in the `/opt/mysql/meb-x.y` directory, where `x.y` represents the version of the software. You need to only back up the Oracle VM Manager database to recover the Oracle VM Manager; however, backing up the entire host might speed up recovery of your Oracle VM Manager if the entire platform becomes unusable.

Oracle VM Servers

Oracle VM servers do not contain any critical information on their system disks except for Oracle VM servers with repositories created on local storage. These repositories are not external to the Oracle VM servers that owned them and they must be backed up directly from the Oracle VM server.

Custom configuration is not supported on Oracle VM servers, except for a few exceptions. For example, if you have a custom NTP configuration, or have added DNS information beyond what you provided during software installation, you must include these configuration files in your backup strategy.

The server pool file system is accessed by all Oracle VM servers in the server pool. The server pool file system contains a small database that contains cluster data. It also contains the Oracle VM server's "heartbeat" to this device, that is, each Oracle VM server writes data to its own region on the device and reads information from the other Oracle VM servers' regions. As such the pool server file system is an important operational element and you must plan for its backup and recovery.

Repositories

Repositories contain virtual machine resources. If a repository becomes inaccessible because the storage has failed or because of an access problem, the Oracle VM Manager and Oracle VM servers continue functioning; however, virtual machines with resources in the repository might experience problems with their applications or might even fail.

Repositories, except for those created on Oracle VM server's local storage, are located on external storage. NFS-type repositories can be accessed over the network, but OCFS2-type repositories are accessed directly by the Oracle VM servers by using the FC or iSCSI protocol. Repositories must be backed up. Their resources are wholly contained in the repository. The Oracle VM Manager database contains only information about the resources stored in the repositories. Backup and restore strategies are discussed later in this lesson.

Oracle VM Components: Oracle VM Guests

- Oracle VM guests:
  - Virtual machine configuration file
  - Virtual or physical disks for the system image (OS)
  - Virtual or physical disks for business applications and application data
  - Ancillary virtual machine resources (ISO files, virtual appliances, and templates)

Virtual Machine Components

All virtual machines, also called guests, have one configuration file that always resides in a repository.

You can choose to create the system and application disks for your guests on virtual disks or physical disks. Many virtual machines have a mix of virtual and physical disks. Virtual disks reside in repositories and physical disks are LUNs assigned directly to a virtual machine, by using iSCSI or FC.

If you created your virtual machine by cloning a template, the virtual machine's disks are virtual disks in a repository. You can add virtual disks and physical disks to a virtual machine at any time.

Virtual Machine Structure and Backup/Restore Operations

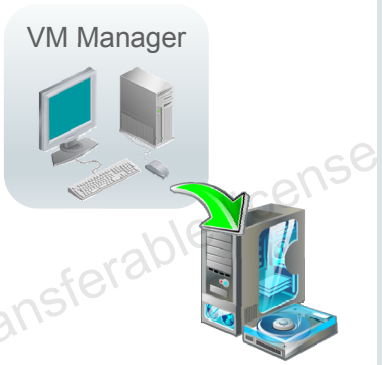
If your virtual machine contains resources spread over several repositories, and the virtual machine fails, then you might have to initiate several restore operations to recover its content.

You must include the backup of resources other than the configuration and disk resources of these virtual machines. For example, if you have highly customized templates that you use to quickly provision virtual machines, include these templates in your backup strategy.

The methodology to back up virtual machines is discussed later in this lesson.

Backing Up the Oracle VM Manager Components

- What to back up (in order of importance)
  - The Oracle VM Manager database
  - The entire `/u01/app/oracle` directory, which includes most configuration files and the keystores containing authentication certificates
  - The system disks
- Backup frequency
  - Oracle recommends daily full backups.
  - An automatic backup of the MySQL database is performed every day.
  - Always perform additional full backups before any major changes.



The components that you must consider when backing up your Oracle VM Manager environment are listed in order of importance.

Oracle VM Manager Database

At a minimum, you must back up the Oracle VM Manager database.

Note: If you are using an older release of Oracle VM with an external Oracle Database for your Oracle VM Manager data repository, use facilities such as `expdb`/`impdb`, `rman`, or Oracle Secure Backup to back up the database.

`/u01/app/oracle` Directory

Most of the Oracle VM Manager software is contained in the `/u01/app/oracle` directory. When you back up the `/u01/app/oracle` directory, you back up all the software products for Oracle VM Manager, (including WebLogic) and all the configuration files, except for the `/etc/sysconfig/ovmm` file and the MySQL Enterprise Backup software.

There are other files outside of `/u01/app/oracle` that are needed for Oracle VM Manager operations:

- The `/etc/init.d/ovmm*` files that are linked from `/etc/rc3.d`
- The `ovm_shell.sh` and `ovmwsh` scripts. The `ovmwsh` script is a link to `/u01/app/oracle/ovm-manager-3/ovmwsh/bin/ovmwsh`.

MySQL Database

The MySQL database files are also backed up when you back up `/u01/app/oracle`, but your database backup might not be valid except if you quiesce the database before the backup. Stopping the `ovmm` service does not quiesce the MySQL database.

For this reason, you can rely on the automatic MySQL database backups that occur every day, or you can perform your own database backups. MySQL database backup and restore are discussed in the next slide.

System Disks

If you include the system disks (in addition to your database backup) as part of your Oracle VM Manager backup strategy, you can recover from a complete failure of the host where you installed the Oracle VM Manager software without having to reinstall and reconfigure the operating system in the host. When you back up the system disks, you also capture a copy of those Oracle VM Manager files that reside outside of the `/u01/app/oracle` directory.

Backing Up the Oracle VM Manager MySQL Database

- A backup of the local MySQL database is performed automatically every 24 hours.
- To initiate a manual backup, use the `/u01/app/oracle/ovm-manager-3/ovm_tools/ bin/BackupDatabase` application.
- Backups are stored in `/u01/app/oracle/mysql/dbbackup`.
- Backup files are rotated to keep 21 backup events.

MySQL Database Backup Notes

- Backups are stored in the `/u01/app/oracle/mysql/dbbackup` directory. You can modify this setting by editing the `DBBACKUP` parameter in the `/etc/sysconfig/ovmm` file.
- You can initiate a backup manually. Oracle recommends that you perform a manual backup before upgrading your Oracle VM Manager.
- The backup script also backs up the keystores where the authentication certificates are stored.
- By default, a manual backup directory acquires a name that avoids the rotation rules.

  Example of manual backup:

```
# /u01/app/oracle/ovm-manager-3/ovm_tools/bin/BackupDatabase -w
```

The `-w` command-line switch forces the backup script to wait until the backup job is complete to end the script execution.

Restoring the Oracle VM Manager MySQL Database

- As user `root`, stop the Oracle VM Manager services (`ovmm` and `ovmcli`) and the MySQL database (`ovmm_mysql`) services.
- As user `oracle`, use the `RestoreDatabase.sh` command, and specify the name of a backup directory, for example, `AutoFullBackup-20160928_184240`.
- Use the Oracle VM Manager "Refresh All" feature.

MySQL Database Restore Notes

- You must stop the `ovmm`, `ovmcli`, and the `ovmm_mysql` services before you restore the database.
- Delete the database files that already exist in the `/u01/app/oracle/mysql/data` directory. Do not delete the following files: `auto.cnf`, `my.cnf`, `mysqlconfig` and `mysql_upgrade_info`.
- Perform the restore by executing the `RestoreDatabase.sh` script.

  Example:

```
# su - oracle
$ bash /u01/app/oracle/ovm-manager-3/ovm_tools/bin/ RestoreDatabase.sh \
MyBackup-20160928_184240
```

- Restart the `ovmm_mysql`, `ovmm`, and the `ovmcli` services in this order.
- After you have completed the restore of the database, you must use the Refresh All feature from the Oracle VM Manager UI or the Oracle VM CLI.
  - From the Oracle VM Manager UI: Click the Servers and VMs tab, click Server Pools in the navigation pane, and click the Refresh All icon on the toolbar of the management pane.

- From the Oracle VM CLI: Run the `refreshAll` command from the Oracle VM CLI prompt.

Note: The Refresh All operation can take a long time. At the completion of the operation, the configuration information about all Oracle VM components is updated in the Oracle VM Manager database.

Additional Post-Restoration Steps

The preceding steps are sufficient when you plan to restore the MySQL database but the rest of the Oracle VM Manager environment is intact. For example, if you experience a database corruption problem, you restore the database only.

However, if you restore the MySQL database as part of an effort to restore the entire Oracle VM Manager environment, additional steps are necessary to re-establish access to the SSL certificates that are restored as part of the MySQL database restore operation. This additional task is discussed in the next topic, which covers the steps to recover all Oracle VM Manager components.

Recovering the Oracle VM Manager Components

- Restore the host environment and the Oracle VM Manager software by using one of the following:
  - Recover all the system disks and the `/u01` directory.
  - Reinstall the OS and the Oracle VM Manager software.
- Quiesce the `ovmm`, `ovmcli`, and `ovmm_mysql` services.
- Restore the Oracle VM Manager MySQL database and restart services.
- If you reinstalled the Oracle VM Manager software:
  - Reregister the restored certificate information for the new WebLogic `admin` user with the `ovmkeytool.sh` command.
  - Regenerate the certificates for the Oracle VM Manager UI and Oracle VM CLI with the `configure_client_cert_login.sh` script.
- Refresh all objects in the database.

What you must recover depends on what failed.

- If the entire Oracle VM Manager host is lost, then you restore all components.
- If you choose to reinstall the Oracle VM Manager software, specify the UUID of the previous Oracle VM environment as a parameter of the `runInstaller.sh` script.
- If only the database has failed, then restore only the MySQL database from a database backup.

Note that the loss of the Oracle VM Manager database has no impact on running Oracle VM servers and guests. However, some facilities are not available when the Oracle VM Manager is not available. For example, DRS and DPM are not operational. Similarly, you cannot make changes to your Oracle VM environment if the database is not available.

If you do not have backups for your Oracle VM Manager components, you can still recover your environment. This scenario is described later in this lesson.

Oracle VM Manager Recovery Notes

If you have restored the OS and `/u01` directory as part of the recovery effort and rebooted the Oracle VM Manager host, the MySQL database and Oracle VM Manager services attempt to restart.

In the same manner, if you have reinstalled the OS and performed a clean installation of the Oracle VM Manager software, the MySQL database and Oracle VM Manager services are running.

Stop these services before proceeding with the restore of the MySQL database.

At this point, you restore the MySQL database by using your latest backup.

The process to restore the MySQL database was described in the previous slide.

After completing the MySQL database restore operation, restart the MySQL database and Oracle VM Manager services.

Post Database Restore Considerations

The backup script for the Oracle VM Manager database backs up the MySQL database, and the SSL keystores and truststore. If you have reinstalled the Oracle VM Manager software, the certificates that are reinstated as part of the keystore and truststore restore operation are for the previous WebLogic `admin` user. Because the new Oracle VM Manager installation creates a new WebLogic `admin` user with a different GUID, you must run the following scripts:

- The `ovmkeytool.sh` script with the `setupWebLogic` parameter. This parameter causes the script to configure the restored keystore settings in WebLogic. The script is located in the `/u01/app/oracle/ovm-manager-3/ovm_upgrade/bin` directory.

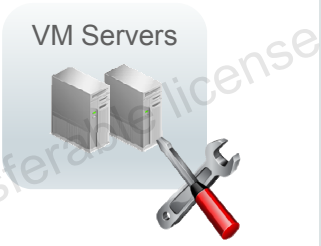
```
# export MW_HOME=/u01/app/oracle/Middleware
# /u01/app/oracle/ovm-manager-3/ovm_upgrade/bin/ovmkeytool.sh \
setupWebLogic
# /sbin/service ovmm restart
```

- The `configure_client_cert_login.sh` script that is located in the `/u01/app/oracle.ovm-manager-3/bin` directory. This script regenerates the certificates to allow the Oracle VM Manager UI and Oracle VM CLI to communicate to the Oracle VM Manager core application.

```
# /u01/app/oracle/ovm-manager-3/bin/configure_client_cert_login.sh
# /sbin/service ovmm restart
```

Recovering the Oracle VM Server

- The Oracle VM server is considered a commodity in the Oracle VM environment.
- If the Oracle VM server fails:
  - Reinstall the software on the server
  - Reapply custom configurations
  - Discover the Oracle VM server by using one of the management interfaces offered by the Oracle VM Manager
  - Add the rebuilt Oracle VM server to networks, file servers, and SAN servers as needed
  - Add the Oracle VM server to its server pool



Recovering from the Loss of an Oracle VM Server

Oracle does not recommend that you back up the Oracle VM servers. Oracle VM servers are a replaceable commodity in the overall Oracle VM infrastructure.

If you lose your Oracle VM server, perform the following actions:

- Reinstall the Oracle VM Server for x86 software on the physical server.
- After installing the software and rebooting the Oracle VM server, the server appears in its previous server pool, but with an unknown state.
- Discover the Oracle VM server: The Oracle VM server moves to the Unassigned Servers folder in the Oracle VM Manager UI.
- Reapply any custom configuration on your server. This might involve updating the `/etc/hosts` file, configuring the LDAP client, and/or configuring NTP.
- Add the Oracle VM server to the networks and storage to which it previously belonged.
- Add the Oracle VM server to the server pool to which it previously belonged.
- Present repositories to the Oracle VM server.

Swapping Hardware for an Oracle VM Server

The recovery process when reinstalling on a new hardware platform is similar to reinstalling on the same hardware, except that the discover operation does not work. This is due to the fact that the Oracle VM Manager remembers the SMBIOS UUID of this server and this UUID has changed with the new hardware.

To fix this problem, instead of discovering the newly reinstalled Oracle VM server, delete it from the server pool. Then delete it from the environment. The Oracle VM server can now be discovered and its new UUID is stored in the Oracle VM Manager database.

Backup and Recovery of the Server Pool File System

- The server pool file system is an important component of your clustered server pools.
  - Without it, the server pool is no longer operational.
- Back up all server pool file systems in your environment:
  - After the server pool is created
  - After adding or removing an Oracle VM server from the server pool
- For an NFS-based server pool file system, make a copy of `.ovspoolfs` and `ovspoolfs.img` in the NFS share.
- For a LUN-based server pool file system, use the utility provided by your storage array to create a backup of the server pool file system.

The server pool file systems contain cluster information for the server pool. Each clustered server pool requires a server pool file system.

Backing Up an NFS-Based Server Pool File System

Go to the NFS server that exposes the NFS share or any other server with access to this share, and make a copy of:

- The `.ovspoolfs` file, which contains configuration information including the UUID of the Oracle VM Manager for the environment
- The `ovspoolfs.img`, which is a disk image containing the OCFS2 file system

Backing Up a LUN-Based Server Pool File System

If available, use the snapshot technology provided by your storage subsystem.

Recovering the Server Pool File System

If the server pool file system fails, cluster operations stop, but the Oracle VM servers and the virtual machines running on these Oracle VM servers continue to run; however, you can no longer perform operations such as moving resources within the cluster.

To recover the server pool file system, perform the restore operation outside of the Oracle VM server by using a recovery process suitable for your storage array. After the restore operation is complete, make the restored server pool file system available to the Oracle VM servers in your server pool.

Strategies for Virtual Machine Availability and Recovery

- Oracle VM-centric
  - Enable high availability for virtual machines.
  - Back up the external storage used by virtual machines.
  - Back up the repositories containing the virtual disks and configuration files of virtual machines.
- Guest-centric
  - Back up your virtual machines as if they were physical hosts.
  - Implement the recovery and availability features that are offered by the applications.

When designing your backup and recovery strategy for your Oracle VM virtual machines, you must consider the type of failures that you want to recover from:

- Loss of a file or loss of a disk, either virtual or physical
- Failure of a database or other application running in the guest
- Failure of the Oracle VM server where the virtual machine is running

Oracle VM provides several ways to protect the resources in the virtual environment. These techniques include:

- Backing up the repositories where virtual resources reside
- Backing up NFS shares, iSCSI, and FC LUNs by using the utilities provided by your disk subsystem
- Enabling high availability for your virtual machines

The types of backups listed in the preceding paragraph are an essential part of a Disaster/Recovery (D/R) strategy for Oracle VM, but the backups remain external to your virtual machines. For example, when you back up the virtual disks or the physical LUNs assigned to your guests, the backup process is unaware of the applications running in the guests. For this reason, you must implement in your guests the same recovery and availability techniques that you use for physical hosts.

As part of this topic, you explore the backup and recovery strategies listed in the slide.

Backup Strategy for Repositories

- Repositories contain resources that can be divided into two categories:
  - Direct virtual machine resources, such as configuration files and virtual disks
  - Indirect resources, such as ISO files and virtual appliances
- You can store indirect resources into separate repositories, but beware of cross-repository dependencies.

Repository Resources

Oracle VM repositories contain resources for installing, cloning, and running virtual machines. These resources include virtual machine configuration files, virtual disks, ISO files, and assembly parts. Templates are a special case of a virtual machine because they are stored as configuration files and virtual disks, but they never run directly. You must clone a template to create a virtual machine from its resources.

You can store resources that are not used to run virtual machines in separate repositories. This is useful for templates and virtual appliances. Perform daily backups of these repositories during the initial implementation period, when there is a lot of activity around virtual machine creation. Perform weekly backups when production is more stable.

When you design your repositories, avoid creating dependencies between repositories. You cannot release ownership of a repository if a virtual machine in another repository is assigned a resource in the repository that you are trying to release.

Repository Backup as Part of Guests Backup Strategy

You create virtual machines with storage resources that reside in repositories (virtual disks) or on LUNs directly assigned to your virtual machines (physical disks). Your backup and recovery strategy for your virtual machines must include the repositories where virtual machine resources reside.

Repository-Level Backup

- For NFS-type repositories:
  - Provide NFS access from the NFS server to the host executing the backup.
- For OCFS2-type repositories:
  - Create a repository export from the Oracle VM Manager UI or Oracle VM CLI.
  - Provide the resulting repository path to the host executing the backup.
- For this type of backup, quiesce your virtual machines if possible.

When performing repository-level backups, your goal is to provide an external server with access to the repositories in your Oracle VM environment.

NFS-Type Repositories

With NFS-type repositories, the NFS access methodology is built in.

OCFS2-Type Repositories

With OCFS2-type repositories, use the Oracle VM Manager facility to create an NFS export for your repositories:

- From the Oracle VM Manager UI, highlight an Oracle VM server with access to the repository that you want to back up.
- Select Repository Exports from the Perspective drop-down list.
- Click the Create Repository Export icon and provide the IP or host name of the backup server and the repository name.

Provide the resulting path under Repository Path and the Oracle VM server IP/hostname to your backup server.

Do not perform backups directly from the Oracle VM servers. Oracle VM servers are dedicated to running virtual machines and other types of workload are not supported.

With repository-level backups, you can perform single file restore operations easily.

Storage-Level Backup

For virtual machines with physical disks residing on storage arrays

- Advantages
  - Same backup procedure for all storage elements residing on the same type of storage array.
  - You can use snapshot technology when available to reduce backup window.
- Disadvantages
  - It might be difficult to identify which virtual machines must be quiesced before the backups.
  - File-level restore might be challenging.

ORACLE

If your virtual machines storage resources are mostly physical disks residing on storage arrays, use the backup and restore facility provided by the storage array to protect your virtual machines. You can also use storage-level backups to protect your repositories and server pool file systems.

Backup techniques for storage arrays include remote backups from a backup server and serverless backups. Serverless backups use disk-to-disk or disk-to-tape technology, and bypass the backup server entirely. This technology manages communication using a protocol, such as Network Data Management Protocol (NDMP).

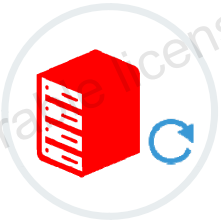
You can also use storage-level backups for Oracle VM repositories and server pool file systems, because these elements can also reside on LUNs.

Storage-level backups are a good choice to protect against large-scale failure. Include this type of backup into your disaster/recovery (D/R) plan.

Restoring single files might be challenging—for example, restoring the `vm.cfg` file for a particular virtual machine.

Guest-Centric Backup

- Your backup and recovery strategy for your guests might include several techniques.
- You can back up your Oracle VM guest as if it were a physical host by using a backup agent.
- Consider system and application data separately for backup purposes.
- Set up continuous availability in your guest by using Oracle's Maximum Availability Architecture (MAA).



Backing Up the Guest as a Physical Host

You can back up the resources of an Oracle VM guest in much the same way as you back up the resources of a "bare metal" server.

For example, you can install a standard backup agent into the virtual machine. Backup agents include the agent for Oracle Secure Backup or Symantec Netbackup.

Note: At this time, Oracle does not support converting your virtual machine into a media server. A media server has direct access to disk or tape devices used during the backup.

By using a backup agent installed in your virtual machine, you can back up your entire virtual machine.

Backing Up System and Applications Data Separately

Consider your Oracle VM guests as made up of:

- System resources, such as the system disk
- Logical resources, such as applications and associated data

Do not mix system and application software on the same disk. Mixing system and application software makes it difficult to devise a sound backup (and restore) strategy for your virtual machines.

Application-Specific Backups

Back up your application software and associated data by using the backup program supplied with the application. For example, use the RMAN backup solution for the Oracle Database.

With these specialized backup programs, you have the ability to perform hot backups.

Continuous Availability

To provide continuous availability for the applications running in your guests, you can use the wide range of Oracle products certified for Oracle's Maximum Availability Architecture (MAA).

MAA is discussed in the next slide.

Conclusion

There is no "one size fits all" methodology. Your final backup and recovery strategy will be a mix of the approaches discussed in this lesson. You must also coordinate your backup and recovery strategy with your D/R plan.

Do validate your backup and recovery process during the initial implementation of Oracle VM when this exercise is the least intrusive.

Revisit your backup strategy at regular intervals to ensure that it still provides the recovery capabilities that you expect.

Protecting Guest Applications with Maximum Availability Architecture

- Continuous application availability must not rely on the high availability and recovery features of the virtual environment.
- The Maximum Availability Architecture (MAA) is a framework for integrating various Oracle products into a cohesive, fault-tolerant business system built for continuous availability of Oracle and non-Oracle applications.
- Using MAA to implement a continuously available business system is the preferred solution for disaster recovery with Oracle VM.

ORACLE

Oracle VM provides several availability features such as maintenance mode, anti-affinity groups, and virtual machine high availability. However, these features are not aware of the complexities of the applications running in your virtual machines.

High Availability

Availability is a measure of the ability to access an application at any time. Availability can be described in terms of reliability, recoverability, error detection and correction, and continuous operations.

To implement high availability for your business applications, you must implement:

- Application-aware high availability solutions
- Operational practices that support these high-availability solutions

Oracle Maximum Availability Architecture (MAA)

The Maximum Availability Architecture (MAA) is a framework for integrating various Oracle products, such as databases, middleware, applications, and web services into a cohesive, fault-tolerant mission-critical business system built for high availability and disaster recovery. Oracle VM disaster recovery relies on the HA and D/R capabilities of the Oracle products running on virtual machines rather than features or capabilities of Oracle VM.

Example of High Availability Solution

If you are running an Oracle Database with multiple instances across several virtual servers, Oracle Real Applications Clusters (RAC) in conjunction with Oracle Clusterware can provide high availability and redundancy for your database environment. If one virtual machine in the cluster fails, the Oracle Database continues to run on the surviving virtual machines that make up the cluster. If more processing power is needed, you can add another virtual machine without service interruption. You can use the Oracle VM anti-affinity group feature to make sure that your virtual servers are not running on the same Oracle VM server.

By adding products such as Oracle Automatic Storage Management (ASM) and Oracle Data Guard to your environment, you can address other aspects of the high-availability spectrum.

Quiz

Which statements are true about the backup and restoration of the MySQL database on the Oracle VM Manager host?

a. You must create a cron job to trigger a daily backup of the MySQL database.
b. Backup files are rotated to keep 21 backup events.
c. You must stop the `ovmm` and `ovmm_mysql` services before you restore the MySQL database.
d. A refresh of all components in the environment is automatically triggered when the restore of the MySQL database completes.

Answer: b, c

Recovery Scenarios

- The first section of this lesson presented information about backing up and recovering elements of your Oracle VM environment.
- The next topic illustrates recovery options for two scenarios:
  - The Oracle VM Manager host is no longer available.
  - The Oracle VM Manager host and the Oracle VM servers are no longer available.
- These recovery scenarios introduce concepts that you can use when designing your D/R strategy.

There are many backup and recovery scenarios and each depends on the type of failure experienced in the Oracle VM environment.

You might lose the LUN that houses a server pool file system for one of your Oracle VM server pools. Or, you might experience the loss of an entire storage array, with elements spanning many server pools and many virtual machines.

In this new topic, you learn to recover from two types of failures:

- In the first scenario, you learn to recover from the failure of the Oracle VM Manager environment.
- In the second scenario, you recover most of the components in your environment, but this scenario assumes that you have backups of the storage elements in your environment.

These scenarios illustrate the need for solid documentation of your Oracle VM environment. Examples of situations where documentation is essential:

- What is the UUID of your Oracle VM Manager installation?
- What LUN is used for a particular server pool file system? Can you identify this LUN by its `/dev/mapper` name?

- Do you know which virtual machine networks are needed by each virtual machine in your environment?
- Is the range of MAC addresses used by virtual machines documented in your backup and recovery plan?

These are the type of questions that you must be able to answer when you are recovering one or more components of your Oracle VM or when recovering the entire environment.

Scenario 1: Recovering the Oracle VM Manager

- If you lose the Oracle VM Manager database, restore it from the latest backup.
- If you do not have a valid backup of the Oracle VM Manager database, you can repopulate the information in the Oracle VM Manager database if:
  - The Oracle VM servers are operational
  - The server pool file systems are mounted to the Oracle VM servers in the server pools
  - The Oracle VM servers storage and networking infrastructure is intact
  - You can obtain the UUID of the current Oracle VM Manager installation

Earlier in this lesson, you learned how to restore the Oracle VM MySQL database.

If you don't have a valid backup of your Oracle VM database, you can recover your environment by:

- Reinstalling the Oracle VM Manager software on your Oracle VM Manager host
- Rediscovering the components in your environment

Note: If your Oracle VM Manager host is intact and only the Oracle VM Manager database is unusable, contact My Oracle Support for guidance on how to wipe out the Oracle VM Manager database only. You can then proceed with the recovery of your Oracle VM Manager environment.

Steps to Rebuild the Oracle VM Manager Environment

- Reinstall the Oracle VM Manager software by using the UUID of the previous installation.
- From the Oracle VM Manager UI or Oracle VM CLI:
  - Discover one Oracle VM server from each server pool
  - Verify the networking configuration
  - Discover the file servers and SAN servers and add the discovered Oracle VM servers to the Admin Servers and Access Groups
  - Refresh the LUNs with repositories on them
  - Present repositories to Oracle VM servers and refresh the repositories to record their contents in the database
  - Discover the remaining Oracle VM servers
  - Add the remaining Oracle VM servers to your storage and network configuration

Rebuilding the Oracle VM Manager

In this scenario, you reinstall the Oracle VM Manager software on a rebuilt host, or you wipe out the Oracle VM Manager database.

If reinstalling the Oracle VM Manager software, reuse the same UUID. Locate the UUID of the current Oracle VM Manager by examining the contents of the `.ovspoolfs` file in the top-level directory of any of your server pool file systems.

Steps to Rebuild the Database Information in Your Oracle VM Manager

- Discover one Oracle VM server from each server pool: You recover a lot of information when you discover Oracle VM servers from your newly installed Oracle VM Manager software. The Oracle VM Manager examines the configuration of your Oracle VM servers and automatically recovers the networking information and the server pools associated with these servers.

  You must examine the recovered networking information and make changes as necessary.
- Reconfigure the NFS file servers, and the FC or iSCSI storage arrays.
- Add the Oracle VM servers to the list of Admin Servers and to the appropriate Access Groups.

- Refresh the LUNs housing the server pool file systems.
- Refresh the LUNs housing the repositories to rediscover these repositories. You cannot see the repositories from the Oracle VM Manager until you have performed this step.
- Present the repositories to the Oracle VM servers.
- After presenting the repositories to the Oracle VM servers, refresh the repositories. After this operation, you can access the contents of the repositories.
- Add the remaining Oracle VM servers to your environment and add these servers to your network and storage configuration.
- Configure the MAC address range that was in place in the previous environment.

  If you have a backup of the Oracle VM Manager file systems, you can use the latest `OVMModelExport_<date and time stamp>.xml` file that is stored in `/u01/app/oracle/mysql/dbbackup` to view the objects in your environment. These `.xml` files contain a list and description of all objects in your Oracle VM Manager database.
- Examine your environment including the location of your virtual machines. Restart a virtual machine to verify that the components are in place.
- Make a manual backup of your Oracle VM Manager database.
- Restart the remaining virtual machines.

Scenario 2: Recovering the Oracle VM Environment

- In this scenario, you rebuild the Oracle VM Manager and Oracle VM servers and recover the virtual machine resources.
- This scenario assumes that you have access to the storage elements of your previous Oracle VM environment:
  - By reaccessing the storage elements (if your storage is intact)
  - By restoring virtual machine resources from backups. These resources include repositories and physical disks assigned to virtual machines.
  - Through replication (or snapshot) of storage elements to a new site

In the second scenario, you recover most of your environment, including the Oracle VM Manager host and the Oracle VM servers.

This scenario is similar to the type of rebuild process that you must consider as part of your D/R design. For example, a simple D/R plan might be:

- **Protect your storage components:** These components include repositories and LUNs assigned directly to virtual machines (physical disks).

  There are several ways to protect your storage: storage arrays that implement RAID, replication and snapshot technology, and regular backups.
- **Recover your Oracle VM environment:** At your recovery site, reinstall the Oracle VM Server for x86 and Oracle VM Manager software on new physical servers.

  Re-establish access to the networks and storage and recover your environment.

Recovery steps are discussed in the next slide.

Steps to Rebuild the Whole Environment

- Prepare a new host and reinstall the Oracle VM Manager software on that host by using the previous UUID.
- Reinstall the Oracle VM Server for x86 software on new physical servers.
- Restore access to storage components: This operation re-establishes access to the repositories and to the LUNs that are assigned directly to your virtual machines.
- Discover the Oracle VM servers.
- Reconfigure your networks: *New network IDs are created*.
- Reconfigure the storage infrastructure.
- Create the server pools: *New cluster IDs are assigned*.
- Reclaim the repositories and restore friendly names.
- Verify all virtual machines resources.

The steps listed in the slide represent a recovery scenario where you rebuild the environment from scratch at a new site, but it is assumed that the storage elements of the previous environment are available, either because they were not affected or because they were restored or replicated.

This scenario is different from the previous scenario where only the Oracle VM Manager is rebuilt.

Two variances are particularly important:

- The network information is lost: When you re-create the networking infrastructure, your new networks acquire network IDs that are different from the network IDs of the previous installation even if you use the same names for your networks.
- The server pools are re-created: When you create new server pools, they acquire new cluster IDs, which are different from the cluster IDs of the previous installation.

Your recovery plan must handle these two changes.

The Importance of Network IDs

- Each network is assigned a network ID:

```
OVM> list network
...
Data:
  id:10f9178ca8  name:storage_net
  id:0a962400  name:10.150.36.0
  id:104e9c33c6  name:vm_net
  id:106ba6ff6c  name:hb_net
OVM>
  ...
```

  - In the virtual machine configuration file, the virtual NIC network assignment references a network ID:

```
# cat vm.cfg
vif = ['mac=00:21:f6:6d:36:d6,bridge=104e9c33c6']
OVM_simple_name = 'iscsi_hvm1'
vnclisten = '127.0.0.1'
serial = 'pty'
...
```


When you add a virtual NIC (VNIC) to a virtual machine, you assign a virtual machine network to this VNIC. This association is made by using the virtual machine network ID.

In the example in the slide:

- The command at the top lists the networks available in the Oracle VM environment. The `vm_net` network is assigned the virtual machine role.
- The command at the bottom lists the virtual machine information for `iscsi_hvm1`. The VNIC for virtual machine `iscsi_hvm1` is assigned to the `vm_net` network, but references the network ID for this network.

When you rebuild your Oracle VM environment, and create new networks, you reuse the names for the networks, but new network IDs are assigned to your new networks. When you restore your virtual machine resources, by restoring your repositories, `vm.cfg` of your virtual machines references network IDs that no longer exist.

Recovering Network Information for Virtual Machines

- Solution 1: Retain the former network IDs when recovering your environment:
  - Save the network information stored in the `/etc/sysconfig/network-scripts/meta-*` files on your Oracle VM servers.
  - Restore the `meta-*` files information on your Oracle VM servers before discovering them.
- Solution 2: Update the `vm.cfg` information for your virtual machines to match the new network IDs.

Solution 1: Retaining and Restoring Network ID Information

You must back up the network ID information of your Oracle VM servers if you plan to reclaim the network IDs during a recovery of your Oracle VM environment.

The network IDs in your Oracle VM servers are saved in files in the `/etc/sysconfig/network-scripts` directory. These files are prefixed with the string `meta` and contain the interface name that belongs to the network.

For example, on Oracle VM server `ovm4.example.com`:

```
[root@ovsvr01 network-scripts]# ls -l meta*
-rw------- 1 root root 132 Aug  1 03:57 meta-bond0
-rw------- 1 root root 123 Aug  1 04:05 meta-eth1
-rw------- 1 root root 125 Aug  1 04:02 meta-eth2
-rw------- 1 root root 120 Aug  1 04:07 meta-eth3
-rw------- 1 root root 121 Aug  1 04:10 meta-eth4
[root@ovsvr01 network-scripts]#
```

The `meta-eth3` file contains:

```
[root@ovm4 network-scripts]# cat meta-eth3
#This file was dynamically created by OVM manager. Please Do not edit
METADATA=ethernet:108162377e{vm_net}:VIRTUAL_MACHINE
```

When you discover your Oracle VM servers, the Oracle VM Manager uses the information in the `meta-*` files to create networks, and uses the same network ID for each network.

Note: The process of recovering network IDs is most important for networks with the Virtual Machine role. Though you can extend this recovery effort to other types of networks, it is not necessary for recovering network information for virtual machines.

After the discovery process, you must add the ports and provide IP configuration information for all Oracle VM servers participating in the networks.

Recovering Network IDs with Networks on VLANs

You must perform additional steps when recovering the Network ID on your alternate Oracle VM site for each virtual machine network on a VLAN segment:

1. Copy the appropriate metafiles to a newly installed Oracle VM server that participates in this network.
2. Additional step: Using one of the Oracle VM Manager management interfaces, create the VLAN port for the network.
3. Rediscover the Oracle VM server. The network appears in Oracle VM Manager.
4. Additional step: Remove and add again the Virtual Machine channel for the new network to force the creation of a bridge in the Oracle VM server.
5. Add the other Oracle VM servers to the network.

Solution 2: Updating the `vm.cfg` Information in Your Virtual Machines

If you did not restore the network IDs from your previous Oracle VM environment, you must update the network information in your virtual machines' configuration files to match the new network IDs. You can do this manually for each virtual machine or you can use a script.

Cluster ID Discrepancy After Recovery

- Before recovery, each OCFS2-type repository has a cluster ID that matches the cluster ID for the server pool.

```
[root@ovsvr01 ~]# mounted.ocfs2 -d
Device                    Stack  Cluster           F
                              UUID                                Label
...
/dev/mapper/1IET_00010002  o2cb   606285f03b2ee547 G   <-- Repository
                              0004FB000005000078A65EDF1EE2A3BD  OVS65edf1ee2a3bd
/dev/mapper/1IET_00010001  o2cb   606285f03b2ee547 G   <-- Server Pool File System
                              0004FB00000500002E84D300CC5CA52B  OVS_POOL_FILESYSTEM
```

- After recovery, the cluster ID for the repository no longer matches the cluster ID for the server pool.

```
[root@ovsvr01 ~]# mounted.ocfs2 -d
Device                    Stack  Cluster           F

                                UUID                                Label
...
/dev/mapper/1IET_00010002  o2cb   13b6177fd903edf0  G   <-- Repository

                                0004FB000005000078A65EDF1EE2A3BD  OVS65edf1ee2a3bd
/dev/mapper/1IET_00010001  o2cb   92090cfd07fc4881  G   <-- Server Pool File System

                                0004FB00000500000FB52A01AFDD57FC  OVS_POOL_FILESYSTEM
```


Before Rebuilding the Oracle VM Environment

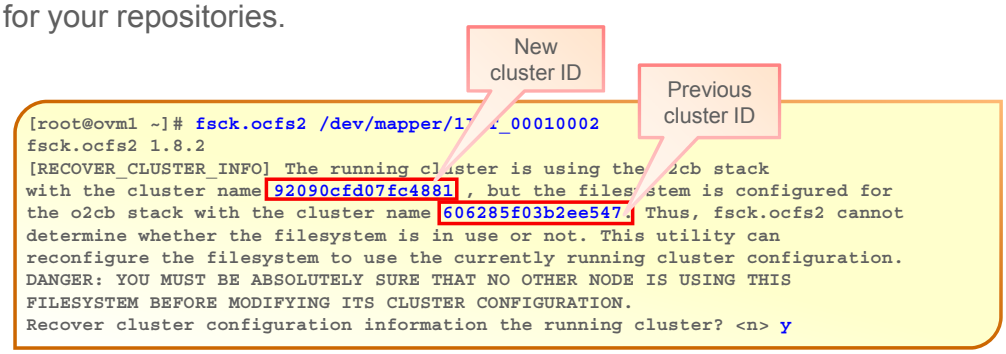
The top portion in the slide shows that the repository on the `/dev/mapper/1IET_00010002` device has cluster ID `606285f03b2ee547`, which matches the cluster ID of the server pool file system.

After Rebuilding the Oracle VM Environment

The bottom portion of the slide shows the new cluster ID for the server pool file system in the recovered environment. This cluster ID, `92090cfd07fc4881`, no longer matches the cluster ID of the repository.

Reclaiming Repositories with Different Cluster IDs

After creating a new server pool as part of the recovery process, update the cluster ID information for your repositories.



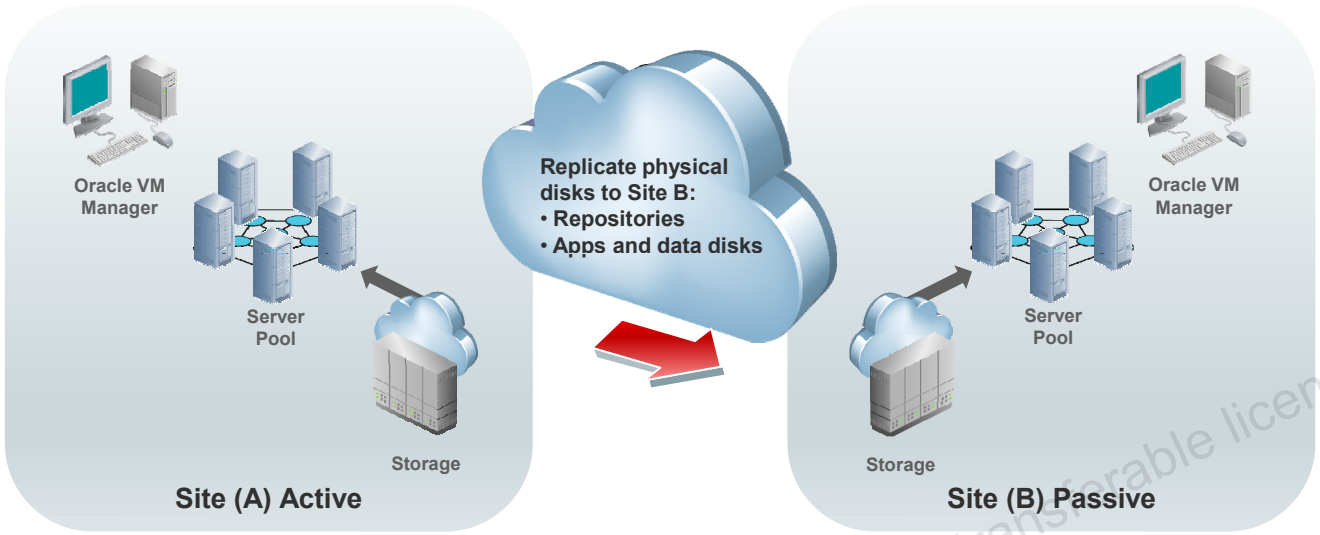
The `fsck.ocfs2` command shown in the slide shows one way to update the cluster ID information after creating new server pools as part of the recovery effort. When you are using the `fsck.ocfs2` command on the repository device, the `fsck` process notifies you that there is a discrepancy between the cluster ID in the OCFS2 repository file system and the cluster ID for the `o2cb` stack running on the Oracle VM server. The `fsck` process asks you if you want to recover the cluster information from the running cluster, that is the cluster that is currently active on the server. Reply `y` to update the cluster ID for the repository. Repeat this process for all repositories that you want to reclaim.

You perform this operation from a single Oracle VM server that is part of the server pool. You must ensure that the OCFS2 file system for the repository is not mounted on any Oracle VM server in the pool.

Note: You can also use the `tunefs.ocfs2` command to update the cluster information on your repositories:

```
# tunefs.ocfs2 --update-cluster-stack /dev/mapper/1IET_00010002
```

D/R Concepts



There are many different DR solutions possible.

Active/Passive D/R Solution

The diagram in the slide illustrates an active/passive approach, where Site A is the active site, and Site B is readied in case a failover is needed:

- The Oracle VM Manager installed at Site B has a UUID that does not match the UUID of the Oracle VM Manager installed at Site A. The distinct UUID at site B allows you to configure this passive site to Enterprise Manager Cloud Control without collision with Site A, the active site.
- The network and storage configurations for Site B mirrors the configuration for Site A. In addition, the network IDs for Site B are the same as those from Site A.
- The server pools in Site B are preconfigured, but they have their own cluster ID and their own Oracle VM servers.
- The following storage components at Site A are replicated to Site B on a regular basis:
  - Storage repositories, NFS or LUN based
  - LUNs directly assigned to Oracle VM guests

The Oracle VM environment at site A is actively running Oracle VM guests but there are no running guests at Site B. The Oracle VM servers at Site B are sitting idle. Both sites have independent server pools, that is, the server pools have different cluster IDs.

Active/Active D/R Solution

You can also prepare a D/R plan by using an active/active solution where the recovery site (Site B) is running its own Oracle VM guests. This solution ensures that the Oracle VM servers at Site B are not sitting idle. However, you must ensure that the standby server pools at the recovery site have enough idle resource to run their own Oracle VM guests plus the Oracle VM guests from the primary site if a failover occurs at the primary site.

For any D/R solution that you implement, you must create automated processes for the recovery process in order to meet your Recovery Time Objective (RTO).

Reclaiming Repositories at Recovery Site

- The recovery site might not have the Oracle VM Manager UUID used at the primary site.
  - After a failover, the repositories are still owned by the primary site.
- In this situation, you must perform additional steps to reclaim all repositories.
  - Refresh the cluster ID.
  - Mount the repository file system and change the UUID in the repository's `.ovsrepo` file to `NONE`.
  - In the Oracle VM CLI, execute the `takeownership` command and specify the repository ID and a server pool.

Recovery Consideration with Active/Passive and Active/Active D/R Solutions

If you configure your recovery site with a new Oracle VM Manager UUID, there are additional steps that you must perform to reclaim your repositories at the recovery site.

Note: For OCFS2-type repositories, you perform these steps after refreshing the cluster ID.

For each repository, you must:

- Mount the repository on one Oracle VM server only and update the `.ovsrepo` file in the top-level directory of the repository by replacing the UUID with the string `NONE`.
- Unmount the repository.
- Use the `takeownership` command in Oracle VM CLI, and specify the repository ID and the server pool (name or ID) to which the repository belongs.

The `takeownership` command forces the Oracle VM Manager to take ownership of a repository when the information in its database indicates that it was owned at one time by a different Oracle VM Manager.

If you are reusing the UUID for the Oracle VM Manager when rebuilding your Oracle VM environment, you do not have to perform these additional recovery steps to reclaim your repositories.

Quiz

Which statements are true about networking information in your Oracle VM environment?

a. When you create a network in your Oracle VM environment, the network is assigned a name and a unique network ID.

b. When you create a network with the same name in a different Oracle VM environment, you get a different ID for the network.

c. When you assign a virtual machine VNIC to a network with the Virtual Machine role, the VNIC is associated with the name of the network.

d. All of the above

Answer: a, b

Summary

In this lesson, you should have learned how to:

- Identify the Oracle VM components to be backed up
- Discuss guidelines for backing up the components
- Perform Oracle VM Manager database backups and restores
- Recover your Oracle VM environment
- Discuss D/R concepts by examining an active/passive D/R solution

Practice Overview

In this practice, you:

- Back up and restore the Oracle VM Manager database
- Recover from the loss of the Oracle VM Manager database
- Recover the whole environment and reclaim the repositories