

Oracle Linux Virtualization Manager Architecture and Planning Guide



F52193-11
December 2023



Oracle Linux Virtualization Manager Architecture and Planning Guide,
F52193-11

Copyright © 2022, 2023, Oracle and/or its affiliates.

Contents

1 About the Docs

Documentation License	1-2
Conventions	1-2
Documentation Accessibility	1-2
Access to Oracle Support for Accessibility	1-2
Diversity and Inclusion	1-2

2 Architecture

Engine	2-2
Host Architecture	2-3
Self-Hosted Engine	2-6
Data Warehouse and Databases	2-7
Access Portals	2-7
Directory Services	2-8
Consoles	2-9

3 Requirements and Scalability Limits

Engine Host Requirements	3-1
KVM Host Requirements	3-3
Firewall Requirements	3-4
Engine Host Firewall Requirements	3-4
Remote Component Firewall Requirements	3-6
KVM Host Firewall Requirements	3-6
Storage Requirements	3-8
Scalability Limits	3-8
Guest Operating System Requirements	3-9

4 Planning Your Environment

Data Centers	4-1
Clusters	4-2

Hosts	4-4
Virtual Machines	4-5
Considerations When Using Snapshots	4-6
Virtual Machine Consoles	4-6
High Availability and Optimization	4-6
Networks	4-9
Logical Networks	4-9
VLANs	4-12
Virtual NICs	4-14
Bonds	4-15
MAC Address Pools	4-16
Storage	4-17
Storage Domains	4-17
Storage Pool Manager	4-18
Virtual Machine Storage	4-18
Storage Leases	4-19
Local Storage	4-19
System Backup and Recovery	4-20
Users, Roles, and Permissions	4-20
System State and History	4-21
Event Logging and Notifications	4-21
Data Visualization with Grafana	4-22
Default Grafana Dashboards	4-22

1

About the Docs

Oracle Linux Virtualization Manager Release 4.5 is based on [oVirt](#), which is a free, open-source virtualization solution. The product documentation comprises:

- **Release Notes** - A summary of the new features, changes, fixed bugs, and known issues in the Oracle Linux Virtualization Manager. It contains last-minute information, which might not be included in the main body of documentation.
- **Architecture and Planning Guide** - An architectural overview of Oracle Linux Virtualization Manager, prerequisites, and planning information for your environment.
- **Getting Started Guide** - How to install, configure, and get started with the Oracle Linux Virtualization Manager using standard or self-hosted configuration. It also provides information for configuring KVM hosts and deploying GlusterFS storage.
- **Administration Guide** - Provides common administrative tasks for Oracle Linux Virtualization Manager such as:
 - setting up users and groups
 - creating data centers, clusters, and virtual machines
 - using virtual machine templates and snapshots
 - migrating virtual machines
 - configuring logical and virtual networks
 - using local, NFS, iSCSI and FC storage
 - backing up and restoring
 - configuring high-availability, vCPUs, and virtual memory
 - monitoring with event notifications and Grafana dashboards
 - upgrading and updating your environment
 - active-active and active-passive disaster recovery solutions

You can also refer to:

- REST API Guide, which you can access from the Welcome Dashboard or directly through its URL <https://manager-fqdn/ovirt-engine/apidoc>.
- Upstream [oVirt Documentation](#).

If you want to provide feedback about this documentation, please complete the [Oracle Help Center feedback form](#).

To access Oracle Linux Virtualization Manager Release 4.4 documentation, PDFs are available at:

- [Release Notes](#)
- [Getting Started Guide](#)
- [Architecture and Planning Guide](#)
- [Administration Guide](#)

Documentation License

The content in this document is licensed under the [Creative Commons Attribution–Share Alike 4.0](#) (CC-BY-SA) license. In accordance with CC-BY-SA, if you distribute this content or an adaptation of it, you must provide attribution to Oracle and retain the original copyright notices.

Conventions

The following text conventions are used in this document:

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

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the [Oracle Accessibility Program website](#).

For information about the accessibility of the Oracle Help Center, see the [Oracle Accessibility Conformance Report](#).

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through [Oracle Accessibility Learning and Support](#).

Diversity and Inclusion

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

2

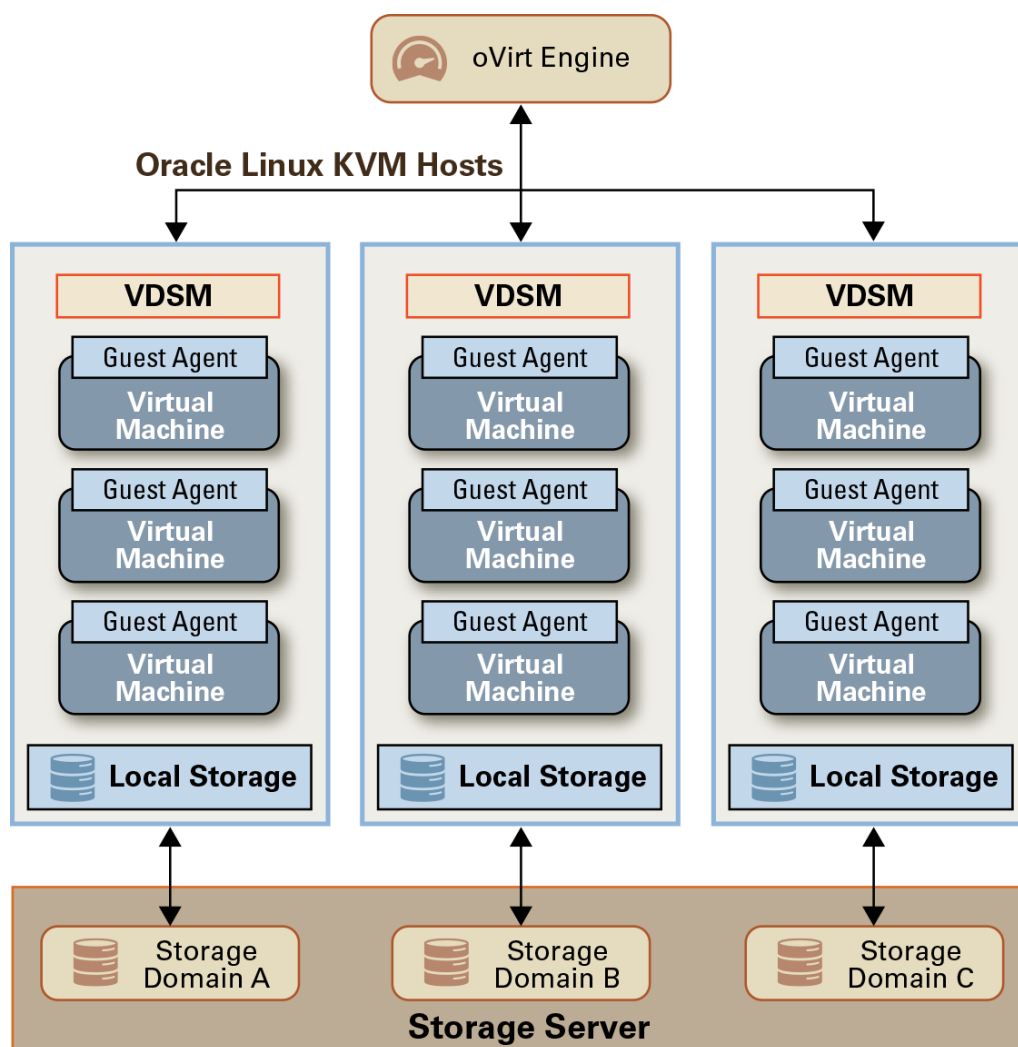
Architecture

Oracle Linux Virtualization Manager is a server virtualization management platform based on the open source oVirt project. You can use it to configure, monitor, and manage an Oracle Linux Kernel-based Virtual Machine (KVM) environment, including hosts, virtual machines, storage, networks, and users. You access the Manager through the Administration Portal or VM Portal.

Oracle Linux Virtualization Manager also provides a Representational State Transfer (REST) Application Programming Interface (API) for managing your KVM infrastructure, allowing you to integrate the Manager with other management systems or to automate repetitive tasks with scripts.

For more information, see [Planning Your Environment](#).

Figure 2-1 Architectural Overview



Engine

The workhorse of Oracle Linux Virtualization Manager is the oVirt engine (engine), which is a WildFly-based Java application that runs as a web service and provides centralized management for server and desktop virtualization. The engine provides many features including:

- Managing the Oracle Linux KVM hosts
- Creating, deploying, starting, stopping, migrating, and monitoring virtual machines
- Adding and managing logical networks
- Adding and managing storage domains and virtual disks
- Configuring and managing cluster, host, and virtual machine high availability
- Migrating and editing live virtual machines
- Continuously balancing loads on virtual machines based on resource usage and policies

- Monitoring all objects in the environment such as virtual machines, hosts, storage, networks

The engine communicates with the Virtual Desktop and Server Manager (VDSM) service which is a host agent that runs as a daemon on the KVM hosts. The engine communicates directly with the VDSM service on Oracle Linux KVM hosts to perform tasks such as managing virtual machines and creating new images from templates.

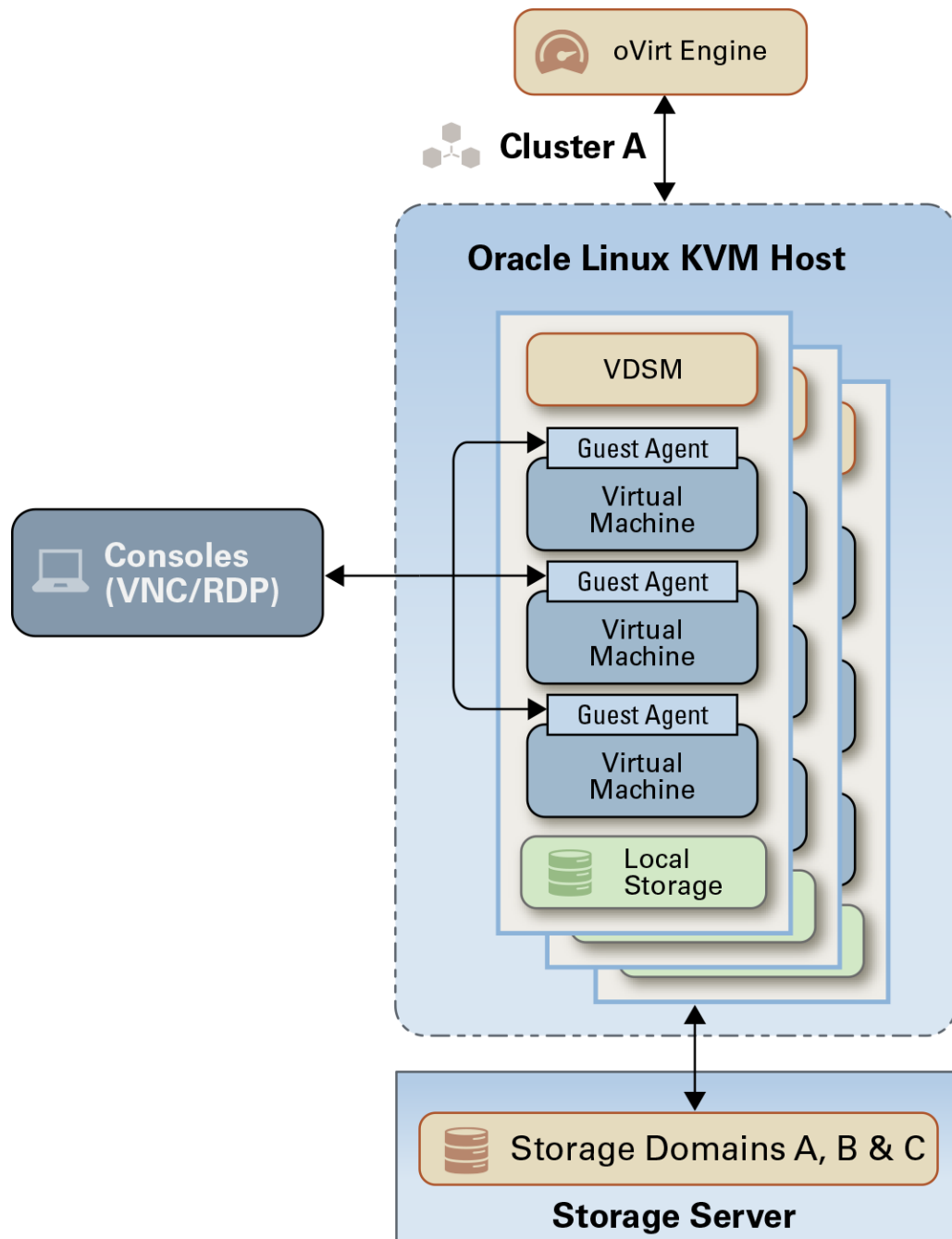
The majority of tasks you can do through the Administration Portal. Additionally, you can perform a subset of tasks using the VM Portal or Cockpit.

Host Architecture

The engine runs on an Oracle Linux server and provides the administration tools for managing the Oracle Linux Virtualization Manager environment. Oracle Linux KVM hosts provide the compute resources for running virtual machines.

For more information, see [Hosts](#).

Figure 2-2 Basic Host Architecture



Kernel-based Virtual Machine (KVM) and Quick Emulator (QEMU)

As a loadable kernel module, KVM

- provides full virtualization through the use of hardware extensions.
- allows a host to make its physical hardware available to virtual machines.
- runs in the kernel space and the virtual machines running on it run as individual QEMU processes in the user space.

QEMU enables KVM to become a complete hypervisor by emulating the hardware for the virtual machines, such as the CPU, memory, network, and disk devices. KVM enables QEMU to execute code in the virtual machine directly on the host CPU. This allows a virtual machine's operating system direct access to the host's resources without any modification.

Host Agent and libvirt

The Virtual Desktop and Server Manager (VDSM) service is a host agent that covers all functionality required by the engine for managing hosts, virtual machines, networks and storage. All communication between the engine and the KVM hosts is handled by the VDSM service that runs on the KVM hosts.

The **libvirt daemon** runs as a service (`libvirtd`) on Oracle Linux KVM hosts and it provides an application programming interface (API) for managing various hypervisors, including Oracle Linux KVM. VDSM uses `libvirt` to manage the complete life cycle of virtual machines and their virtual devices on the host, and to collect statistics about them.

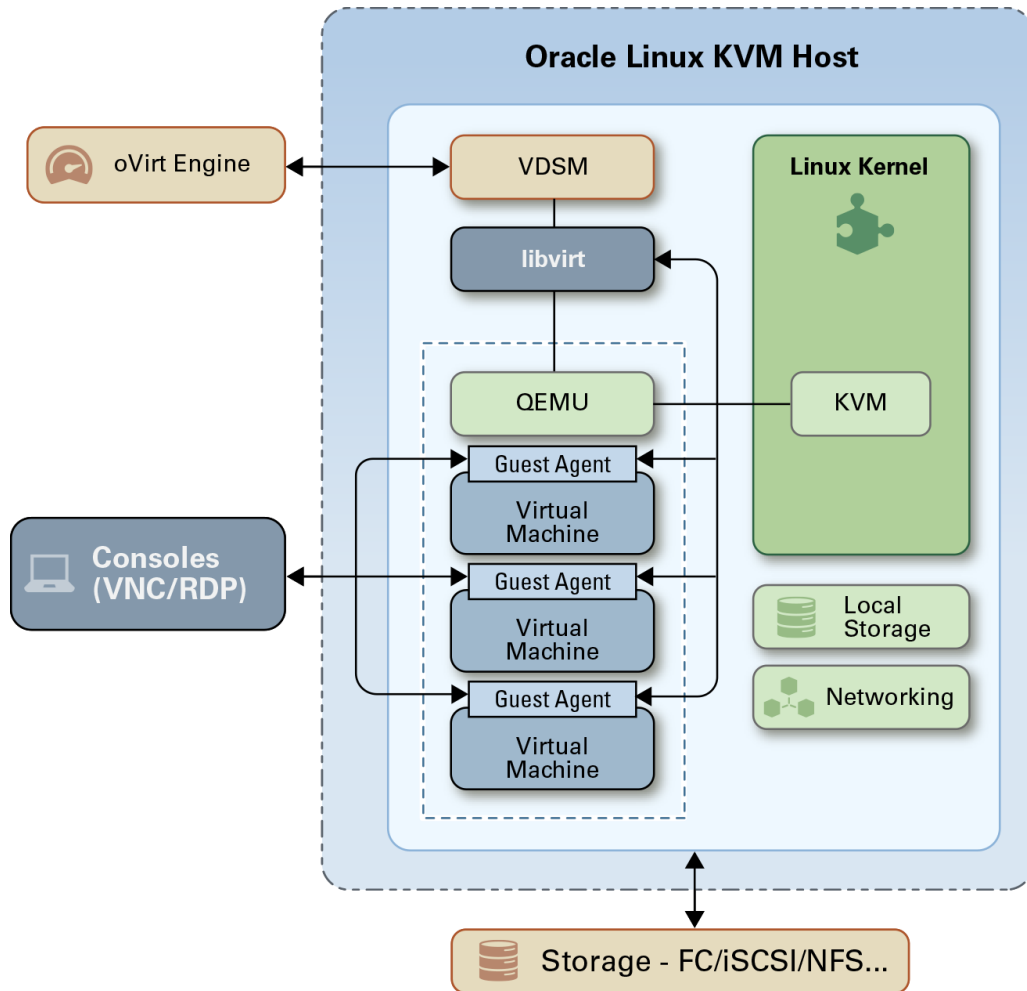
Guest Agent

The guest agent runs inside the virtual machine, and provides information on resource usage to the engine. Communication between the guest agent and engine is done over a virtualized serial connection.

The guest agent provides:

- information, notifications, and actions between the engine and the guest.
- the guest machine name, guest operating system, and other details to the engine, including associated IP addresses, installed applications, and network and RAM usage.
- a single sign-on so an authenticated user to the engine does not need to authenticate again when connected to a virtual machine.

Figure 2-3 Agents, QEMU, and libvirt



Self-Hosted Engine

In Oracle Linux Virtualization Manager, a self-hosted engine is a virtualized environment where the engine runs inside a virtual machine on the hosts in the environment. The virtual machine for the engine is created as part of the host configuration process. And, the engine is installed and configured in parallel to the host configuration.

Since the engine runs as a virtual machine and not on physical hardware, a self-hosted engine requires less physical resources. Additionally, since the engine is configured to be highly available, if the host running the Engine virtual machine goes into maintenance mode or fails unexpectedly the virtual machine is migrated automatically to another host in the environment. A minimum of two self-hosted Engine hosts are required to support the high availability.

You use the oVirt Engine Virtual Appliance to install the engine virtual machine. The appliance is installed during the deployment process; however, you can install the appliance on the host before starting the deployment if required:

```
# dnf install ovirt-engine-appliance
```

If you plan to use bonded interfaces for high availability or VLANs to separate different types of traffic (for example, for storage or management connections), you should configure these interfaces before deployment.

If you want to customize the engine virtual machine, you can use a custom cloud-init script with the appliance. You can generate a default cloud-init script during deployment and customize as needed.

To deploy a self-hosted engine, see Self-Hosted Engine Deployment in the [Oracle Linux Virtualization Manager: Getting Started Guide](#).

 **Note:**

To review conceptual information, troubleshooting, and administration tasks, see the oVirt Self-Hosted Engine Guide in [oVirt Documentation](#).

Data Warehouse and Databases

There are two PostgreSQL databases in Oracle Linux Virtualization Manager. The engine configuration creates a database called `engine`. If you elect to install the `ovirt-engine-dwh` package, a second database called `ovirt_engine_history` is created:

- The engine database (`engine`) stores persistent information about the state of the Oracle Linux Virtualization Manager environment, its configuration, and its performance. The historical configuration information and statistical metrics are collected every minute.
- The data warehouse database is a management history database (`ovirt_engine_history`) that can be used by any application to retrieve historical configuration information and statistical metrics for data centers, clusters, and hosts.

The data warehouse service (`ovirt-engine-dwd`):

- Extracts data from the `engine` database, performs ETL, and inserts it into the `ovirt_engine_history` database.
- Tracks three types of changes:
 - When new entity is added to the `engine` database, `ovirt-engine-dwd` service replicates the change to the `ovirt_engine_history` database.
 - When an existing entity is updated, `ovirt-engine-dwd` service replicates the change to the `ovirt_engine_history` database.
 - When an entity is removed from the `engine` database, a new entry in the `ovirt_engine_history` database flags the corresponding entity as removed.

Both the history and engine databases can run on a remote host to reduce the load on the engine host. Running these databases on a remote host is a technology preview feature. For more information, see Technology Preview in the [Oracle Linux Virtualization Manager: Release Notes](#).

Access Portals

Oracle Linux Virtualization Manager provides three portals you can use to configure, manage, and monitor your environment: Administration Portal, VM Portal, and Monitoring Portal.

The Administration Portal is the graphical administration interface of the oVirt Engine server. Administrators can monitor, create, and maintain all elements of the virtualized environment from web browsers. Tasks that can be performed from the Administration Portal include:

- Creation and management of virtual infrastructure (networks, storage domains)
- Installation and management of hosts
- Creation and management of logical entities (data centers, clusters)
- Creation and management of virtual machines
- User and permission management

The **Cockpit web interface** enables you to monitor a KVM host's resources and to perform administrative tasks. Cockpit must be installed and enabled separately. You can access a host's Cockpit web interface from the Administration Portal or by connecting directly to the host.

The VM Portal presents a comprehensive view of a virtual machine and allows the user to start, stop, edit, and view details of a virtual machine. The actions available to a user in the VM Portal are set by a system administrator who can delegate additional administration tasks to a user, such as:

- Create, edit, and remove virtual machines
- Manage virtual disks and network interfaces
- Create and use snapshots to restore virtual machines to previous states

Direct connection to virtual machines is facilitated with VNC clients. Both protocols provide the user with an environment similar to a locally installed desktop. The administrator specifies the protocol used to connect to a virtual machine at the time of the virtual machine's creation.

For more information on the VM Portal, see [oVirt Documentation](#).

The Monitoring Portal opens Grafana where you can see the built-in Grafana dashboards: Executive, Inventory, Service Level, and Trend. You can create customized dashboards or copy and modify existing dashboards according to your reporting needs.

Grafana integration is enabled and installed by default when you run the `engine-setup` in a stand alone Manager or Self-Hosted engine installation. You might need to install Grafana manually under some scenarios such as performing an upgrade, restoring a backup, or when the data warehouse is migrated to a separate machine.

For more information on the Monitoring Portal, see [oVirt Documentation](#) and [Grafana Documentation](#).

Directory Services

You can use Active Directory, OpenLDAP, and 389DS as an external directory server to provide user account and authentication services. If an external directory server is being used, the oVirt engine uses these directory services to receive user and group information when assigning permissions for roles.

Consoles

You can use either Virtual Network Computing (VNC) or Remote Desktop Protocol (RDP) to provide graphical consoles for virtual machines. From the console, you can work and interact directly with your virtual machines as you would with physical machines.

VNC

When using VNC, either use the Remote Viewer application or a VNC client to open a console to a virtual machine.

If you want to use a locally installed remote-viewer application, you can install the application using your package manager (`yum` or `dnf install virt-viewer`) or download it from [Virtual Machine Manager](#).

If you want to use a browser-based console clients, the certificate authority must be imported in your browser since the communication is secured. You can download the certificate authority by navigating to `https://<your engine address>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA`.

RDP (Windows only)

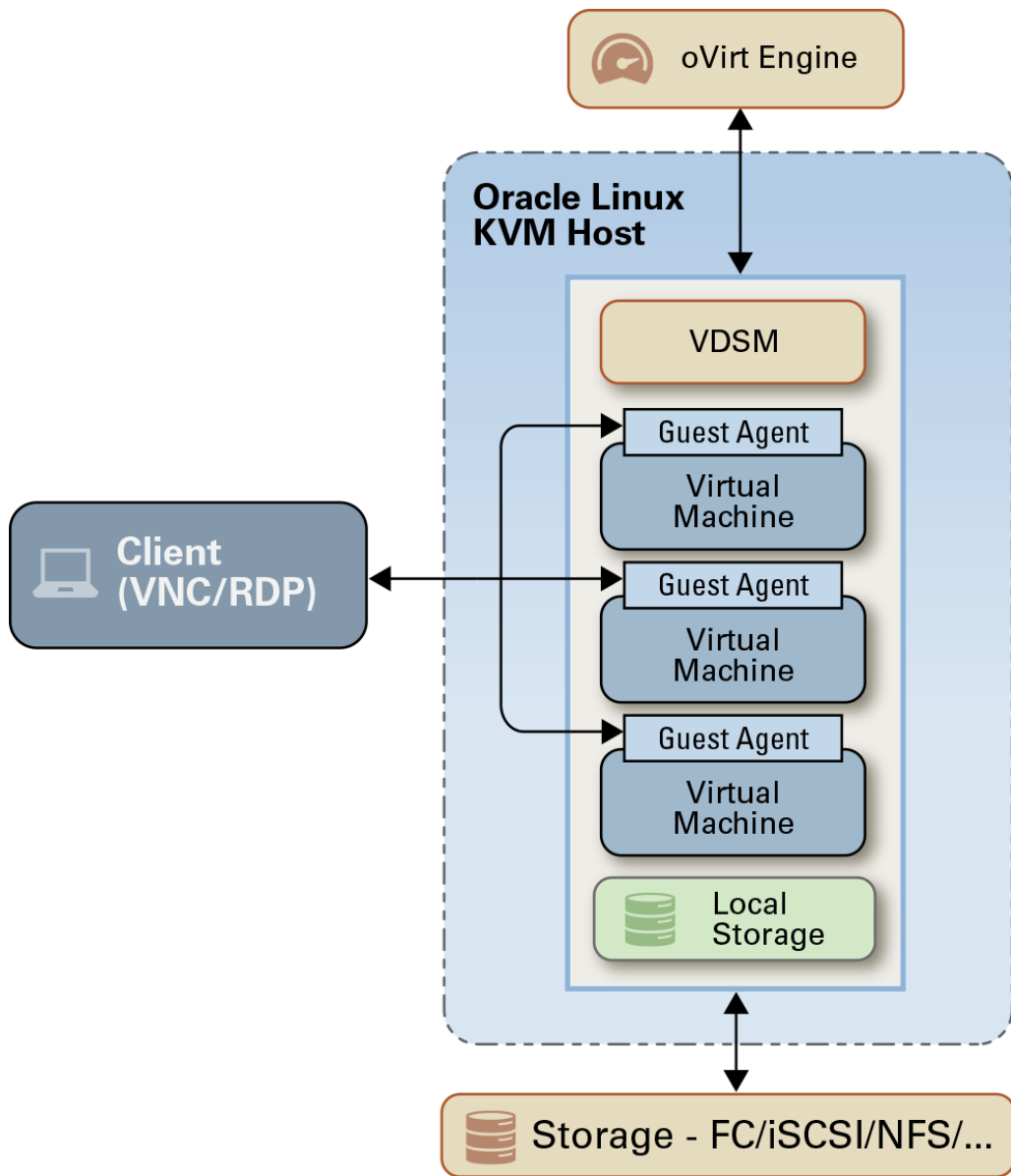
RDP is only available when you access virtual machines from a Windows machine on which the Microsoft Remote Desktop application has been installed. You must also set up remote sharing on the virtual machine and ensure the firewall is configured to allow remote desktop connections before you can connect to a Windows virtual machine using RDP.

Important:

See Windows Virtual Machines Lose Functionality Due To Deprecated Guest Agent in the Known Issues section of the [Oracle Linux Virtualization Manager: Release Notes](#).

For more information see Installing Remote Viewer on Client Machine in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Figure 2-4 Client Consoles Connected to Linux KVM Host



3

Requirements and Scalability Limits

The following sections provide detailed requirements for a Oracle Linux Virtualization Manager Release 4.5 environment as well as the scalability limitations.

Engine Host Requirements

The engine host system requirements are:

- Host must be on-premise.
- Oracle Linux 8.8 (or later Oracle Linux 8 release) with **Minimal Install** selected as the base environment for the installation.
- Unbreakable Enterprise Kernel Release 6, Unbreakable Enterprise Kernel Release 7, or Red Hat Compatible Kernel.



Note:

Do not configure the same host as a standalone engine and a KVM host.

Refer to the following tables for the minimum and recommended hardware requirements for the engine host system within the following deployment sizes:

- **Small deployment** (up to 128 KVM hosts and 1,250 VMs)
- **Medium deployment** (up to 512 KVM hosts and 5,000 VMs)
- **Large deployment** (up to 1024 KVM hosts and 10,000 VMs)



Important:

For medium and large deployments, you should run the `engine-vacuum` command on a regular basis to maintain the databases by updating tables and removing dead rows, allowing disk space to be reused. See Reclaiming Database Storage in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Table 3-1 Engine Host System Hardware Requirements - Small Deployment

Resource	Minimum	Recommended
CPU	64-bit two (2) core CPU	64-bit four (4) core or greater CPU
Memory	4 GB of available system RAM	16 GB or greater of available system RAM
Hard disk	25 GB local writable hard disk	50 GB or greater of local writable hard disk

Table 3-2 Engine Host System Hardware Requirements - Medium Deployment

Resource	Minimum	Recommended
CPU	64-bit four (4) core or greater CPU	64-bit eight (8) core or greater CPU
Memory	16 GB of available system RAM	32 GB or greater of available system RAM
Hard disk	50 GB local writable hard disk	100 GB or greater of local writable hard disk

Table 3-3 Engine Host System Hardware Requirements - Large Deployment

Resource	Minimum	Recommended
CPU	64-bit eight (8) core CPU	64-bit sixteen (16) core or greater CPU
Memory	32 GB of available system RAM	64 GB or greater of available system RAM
Hard disk	75 GB local writable hard disk	150 GB or greater of local writable hard disk

 **Note:**

If Data Warehouse is installed and if memory is being consumed by existing processes, consider using the recommended amount of system memory based on deployment size.

Table 3-4 Engine Host System Network Interface Requirements - All Deployment Sizes

Minimum	Recommended
One network interface card (NIC) with bandwidth of at least 1 Gbps	Two or more NICs with bandwidth of at least 1 Gbps

For information about x86-based servers that are certified for Oracle Linux with UEK, see the [Hardware Certification List for Oracle Linux and Virtualization](#).

For more details about installation, system requirements and known issues, see:

- [Oracle® Linux 8: Release Notes for Oracle Linux 8.8](#).
- [Unbreakable Enterprise Kernel Documentation](#).
- [Oracle® Linux 8: Installing Oracle Linux](#).

! Important:

Oracle does not support Oracle Linux Virtualization Manager on systems where the `ol8_developer`, `ol8_developer_EPEL`, `ol8_codeready_builder`, `ol8_distro_builder`, `ol8_developer_UEKR6`, or `ol8_developer_UEKR7` repositories are enabled, or where software from these repositories is currently installed on the systems where the Manager will run. Even if you follow the instructions in this document, you may render your platform unsupported if these repositories or channels are enabled or software from these channels or repositories is installed on your system.

KVM Host Requirements

Minimum system requirements for Oracle Linux KVM hosts:

- Host must be on-premise,
- Oracle Linux 8.8 (or later Oracle Linux 8 release) with:
 - **Minimal Install** selected as the base environment for the installation
 - Unbreakable Enterprise Kernel Release 6, Unbreakable Enterprise Kernel Release 7, or Red Hat Compatible Kernel.

- 64-bit dual-core CPU

Recommended: Multiple CPUs

The CPUs must support either the Intel VT-x or the AMD AMD-V hardware virtualization extensions and the extensions must be enabled in the host's BIOS. The CPUs must also support the No eXecute flag (NX).

- 2 GB RAM

Maximum Tested: 6 TB

The amount of RAM required varies depending on guest operating system requirements, guest application requirements, and guest memory activity and usage.

- 1 network interface card (NIC) with bandwidth of at least 1 Gbps

Recommended: 2 or more NICs with bandwidth of at least 1 Gbps

Multiple NICs are recommended so that NICs can be dedicated for network intensive activities, such as virtual machine migration.

- 60 GB of locally accessibly, writable disk space dedicated to Oracle Linux Virtualization Manager, allocated as follows:

Allocation	Size
/ (root)	30 GB
/boot	1 GB
/var	29 GB

For information about x86-based servers that are certified for Oracle Linux with UEK, see the [Hardware Certification List for Oracle Linux and Virtualization](#).

NOT_SUPPORTED:

Do not install any third-party watchdogs on your Oracle Linux KVM hosts, as they can interfere with the watchdog daemon provided by VDSM.

Do not install any other applications on the Oracle Linux KVM hosts as they may interfere with the operation of the KVM hypervisor.

For more details about installation, system requirements and known issues, see:

- [Oracle® Linux Documentation](#).
- [Unbreakable Enterprise Kernel Documentation](#).

Firewall Requirements

Before you install and configure the Oracle Linux Virtualization Manager engine or any KVM hosts ensure you review the following firewall requirements.

**Note:**

Oracle Linux Virtualization Manager requires IPv6 to remain enabled on the computer or virtual machine where you are running the Manager. Do not disable IPv6 on the Manager machine, even if your systems do not use it.

Engine Host Firewall Requirements

When you run the `engine-setup` command to configure Oracle Linux Virtualization Manager, you can have the Setup program automatically configure the firewall ports on the host. Use the following information if you want to manually configure firewalls.

The following table shows the ports that are configured by default. The Setup program enables you to choose different ports for some of the configuration options, see Engine Configuration Options in the [Oracle Linux Virtualization Manager: Getting Started Guide](#).

Port	Protocol	Source	Destination	Purpose	Encrypted by default
Not applicable	ICMP	Oracle Linux KVM hosts	Manager host	(Optional) Diagnostics	No
22	TCP	External systems	Manager host	(Optional) SSH access to the Manager host for administration and maintenance	Yes

Port	Protocol	Source	Destination	Purpose	Encrypted by default
80, 443	TCP	Administrati on Portal clients VM Portal clients Oracle Linux KVM hosts REST API clients	Manager host	HTTP access to the Manager	Yes
2222	TCP	Clients	Manager host	SSH access to virtual machine serial consoles	Yes
5432	TCP,UDP	Manager host Data Warehouse Service External systems	Manager host	(Optional) Connections to PostgreSQL database server Only required if the Engine database or the Data Warehouse database run on the Manager host	No
6100	TCP	Administrati on Portal clients VM Portal clients	Manager host	(Optional) WebSocket proxy access to the noVNC or HTML 5 virtual machine consoles Only required if the WebSocket proxy runs on the Manager host	No
7410	UDP	Oracle Linux KVM hosts	Manager host	(Optional) Kdump notifications Only required if Kdump is enabled	No

Port	Protocol	Source	Destination	Purpose	Encrypted by default
54323	TCP	Administration Portal clients	Manager host	Image I/O Proxy access to upload images Only required if the Image I/O Proxy runs on the Manager host	Yes

Remote Component Firewall Requirements

Some Oracle Linux Virtualization Manager components can run on separate remote hosts. Refer to the following table for information to help you configure the firewall on remote hosts.

Port	Protocol	Source	Destination	Purpose	Encrypted by default
5432	TCP,UDP	Manager host Data Warehouse Service External systems	PostgreSQL database server	Connections to PostgreSQL database server Required if the Engine database or the Data Warehouse database run on a remote host	No
6100	TCP	Administration Portal clients VM Portal clients	WebSocket proxy host	WebSocket proxy access to the noVNC or HTML 5 virtual machine consoles Required if the WebSocket proxy runs on a remote host	No

KVM Host Firewall Requirements

When you add an Oracle Linux KVM host to Oracle Linux Virtualization Manager, the existing firewall configuration on the host is overwritten and the required firewall ports are configured automatically.

To disable automatic firewall configuration when adding a KVM host, clear the **Automatically configure host firewall** check box under **Advanced Parameters**. Then, refer to the following table for information to help you manually configure the firewall.

Port	Protocol	Source	Destination	Purpose	Encrypted by default
22	TCP	Manager host	KVM hosts	(Optional) SSH access to KVM hosts	Yes
111	TCP	NFS storage server	KVM hosts	(Optional) NFS connections Only required if you use NFS storage	No
161	UDP	KVM hosts	Manager host	(Optional) Simple network management protocol (SNMP) Only required if you want to send SNMP traps to external SNMP managers	No
2223	TCP	Manager host	KVM hosts	SSH access to virtual machine serial consoles	Yes
5900 to 6923	TCP	Administration Portal clients VM Portal clients	KVM hosts	Access to virtual machine consoles using VNC or RDP protocols	Yes
5989	TCP,UDP	Common Information Model Object Manager (CIMOM)	KVM hosts	(Optional) CIMOM connections Only required if you use CIMOM to monitor virtual machines running on the host	No
6081	UDP	KVM hosts	KVM hosts	(Optional) Open Virtual Network (OVN) connections Only required if the OVN network provider is enabled	No
9090	TCP	Manager host Client machines	KVM hosts	(Optional) Cockpit connections	Yes
16514	TCP	KVM hosts	KVM hosts	Virtual machine migration using libvirt	Yes

Port	Protocol	Source	Destination	Purpose	Encrypted by default
49152 to 49216	TCP	KVM hosts	KVM hosts	Automated and manual virtual machine migration and fencing using VDSM	Yes
54321	TCP	Manager host KVM hosts	KVM hosts	VDSM communication with the Oracle Linux Virtualization Manager and other KVM hosts	Yes
54322	TCP	Manager host Image I/O Proxy host	KVM hosts	Communication with the Image I/O Proxy to upload images Only required if the Image I/O Proxy runs on the Manager host or a separate host	Yes

Storage Requirements

Before you can create virtual machines, you must provision and attach storage to a data center. You can use Network File System (NFS), Internet Small Computer System Interface (iSCSI), Fibre Channel Protocol (FCP), or Gluster storage. You can also configure local storage attached directly to hosts.

Storage devices in Oracle Linux Virtualization Manager are referred to as **data domains**, which are used to store virtual hard disks, snapshots, ISO files, and templates. Every data center must have at least one data domain. Data domains cannot be shared between data centers.

For more information, see:

- Storage in the [Oracle Linux Virtualization Manager: Architecture and Planning Guide](#)
- Storage in the [Oracle Linux Virtualization Manager: Administration Guide](#)

Scalability Limits

The following tables contain the scalability limits for the Oracle Linux Virtualization Manager host, Oracle Linux KVM hosts, virtual machines and storage.

Table 3-5 Manager Host Limits

Component	Small Deployment	Medium Deployment	Large Deployment
Servers managed by one engine	128	512	1024
Concurrently running virtual machines	1250	5000	10,000
VLANs managed by one engine	1024 (for all size deployments)		
Servers managed per data center	250 (for all size deployments)		

Table 3-6 Oracle Linux KVM Host Limits

Component	Maximum
Physical CPUs (cores)	384
Memory	6 TB
Concurrently running virtual machines on a single host	600, depending on the performance of the host

Table 3-7 Virtual Machine Limits

Component	Maximum
Virtual CPUs	256
Virtual RAM	2 TB
Virtual NICs	10

Table 3-8 Storage Limits

Component	Maximum
Domains	50
Hosts per domain	Unlimited
Logical volumes per block domain	1500
<ul style="list-style-type: none"> • LUNs per storage domain • LUNs per Oracle Linux Virtualization Manager 	<ul style="list-style-type: none"> • 400 • 2000
Disk size	500 TB (limited to 8 TB by default)

Guest Operating System Requirements

There are several guest operating systems you can use to configure a KVM host for use with Oracle Linux Virtualization Manager. For information on the supported guest operating systems, see the [Oracle® Linux: KVM User's Guide](#).

 **Note:**

The Administration User Interface offers a number of additional guest operating systems that are untested with Oracle Linux KVM and therefore not supported.

4

Planning Your Environment

Before installing Oracle Linux Virtualization Manager, review this section to help plan your deployment. For more information about the virtualization management platform, see [Architecture](#).

! Important:

The engine server and KVM hosts can be configured on a single NIC, a Bond, or a VLAN interface, but *all hosts must be connected to the same network segment*.

Data Centers

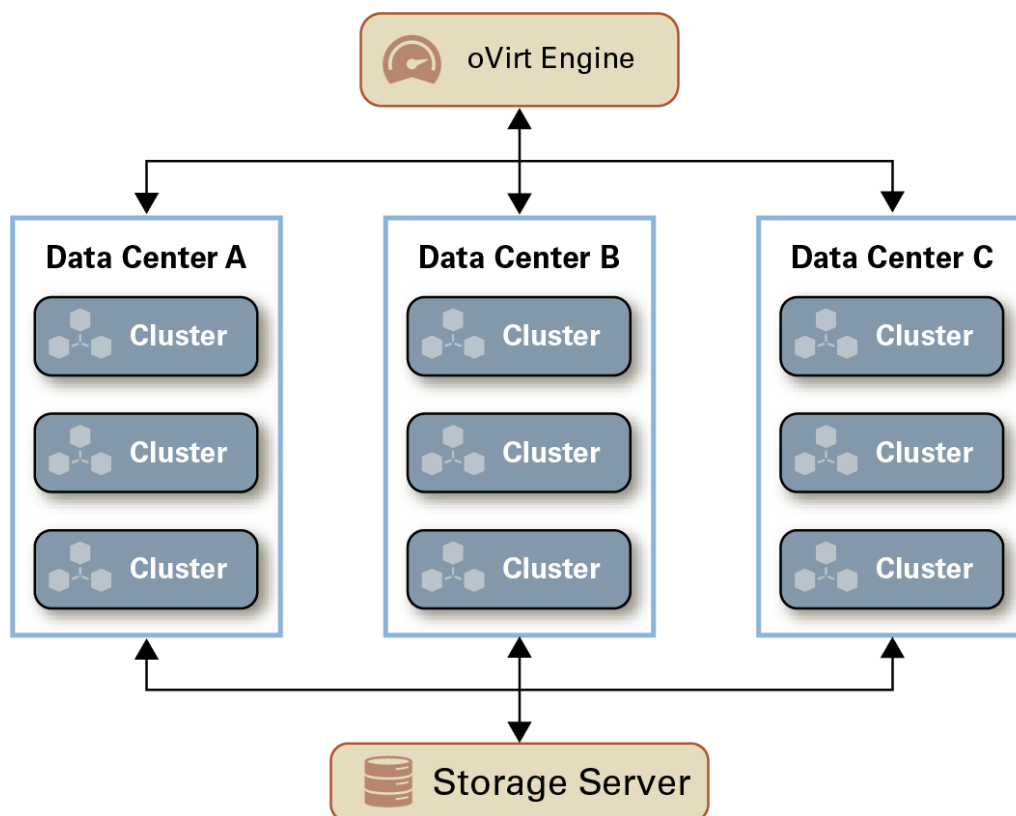
A data center is a high-level logical entity for all physical and logical resources in the environment. You can have multiple data centers and all the data centers are controlled from a single Administration Portal. For more information, see Data Centers in the [Oracle Linux Virtualization Manager: Administration Guide](#).

When you install Oracle Linux Virtualization Manager, a default data center (`Default`), which you can rename and configure. You can also create and configure additional data centers. To initialize any data center, you must add a cluster, a host, and a storage domain:

- **Cluster** - A cluster is an association of physical hosts sharing the same storage domains and having compatible processors. Every cluster belongs to a data center; every host belongs to a cluster. A cluster has to have a minimum of one host, and at least one active host is required to connect the system to a storage pool.
- **KVM Host** - Hosts, or hypervisors, are the physical servers that run virtual machines. You must have at least one host in a cluster. KVM hosts in a datacenter must have access to the same storage domains.
- **Storage Domain** - Data centers must have at least one data storage domain. Set up the data storage domain of the type required for the data center: NFS, iSCSI, FCP or Local.

Logical networks are not required to initialize a data center, but are required for Oracle Linux Virtualization Manager to communicate with all components of a data center. Logical networks are also used for the virtual machines to communicate with hosts and storage, for connecting clients to virtual machine resources, and for migrating virtual machines between the hosts in a cluster.

Figure 4-1 Data Center



Clusters

A cluster consists of one or more logical grouping of Oracle Linux KVM hosts on which a collection of virtual machines can run. The KVM hosts in a cluster must have the same type of CPU (either Intel or AMD).

Each cluster in the environment must belong to a data center and each KVM host must belong to a cluster. During installation, a default cluster is created in the `Default` data center. For more information, see Clusters in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Virtual machines are dynamically allocated to any KVM host in the cluster and can be migrated between them, according to policies defined on the cluster and settings on the virtual machines. The cluster is the highest level at which power and load-sharing policies can be defined. Since virtual machines are not bound to any specific host in the cluster, virtual machines always start even if one or more of the hosts are unavailable.

Figure 4-2 Single Cluster

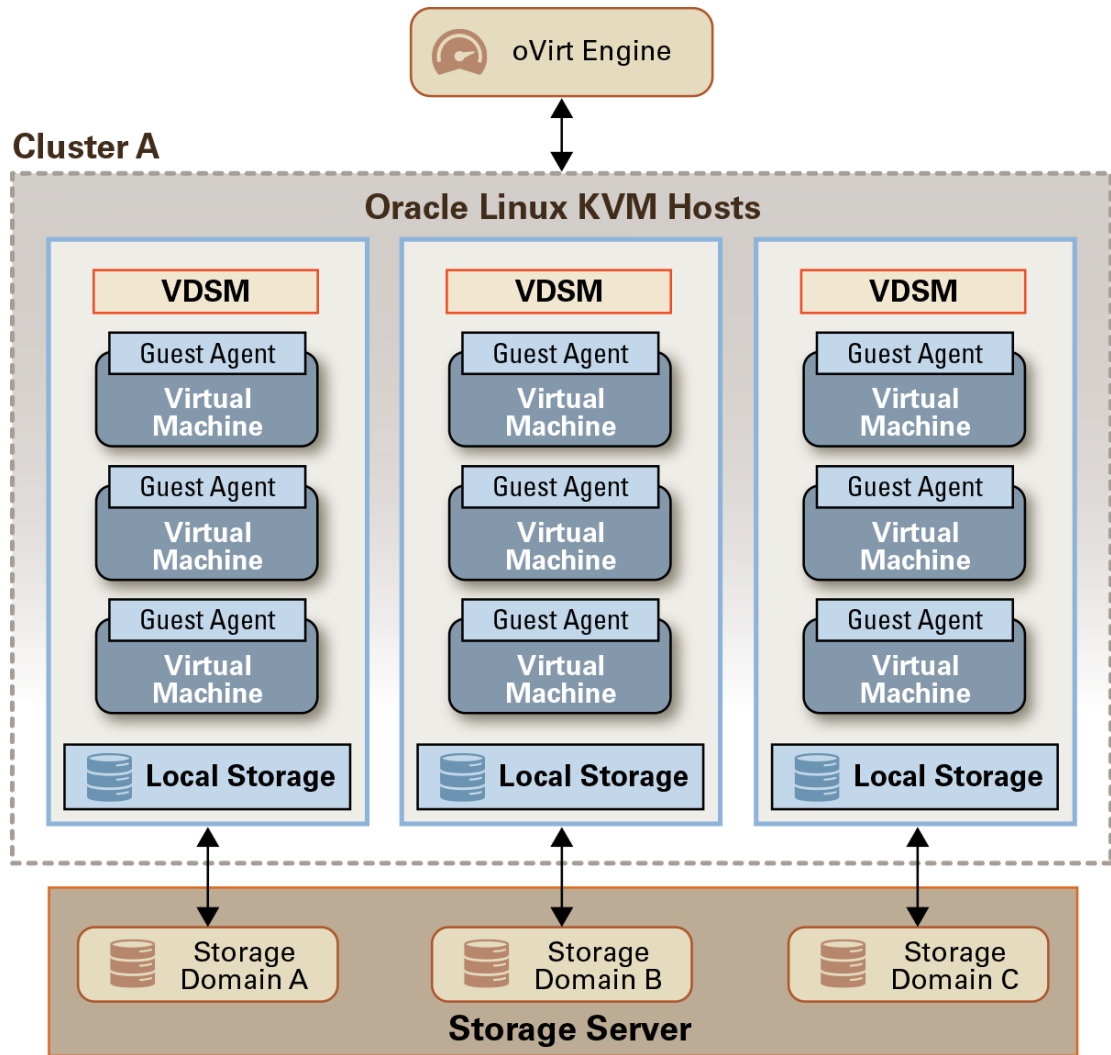
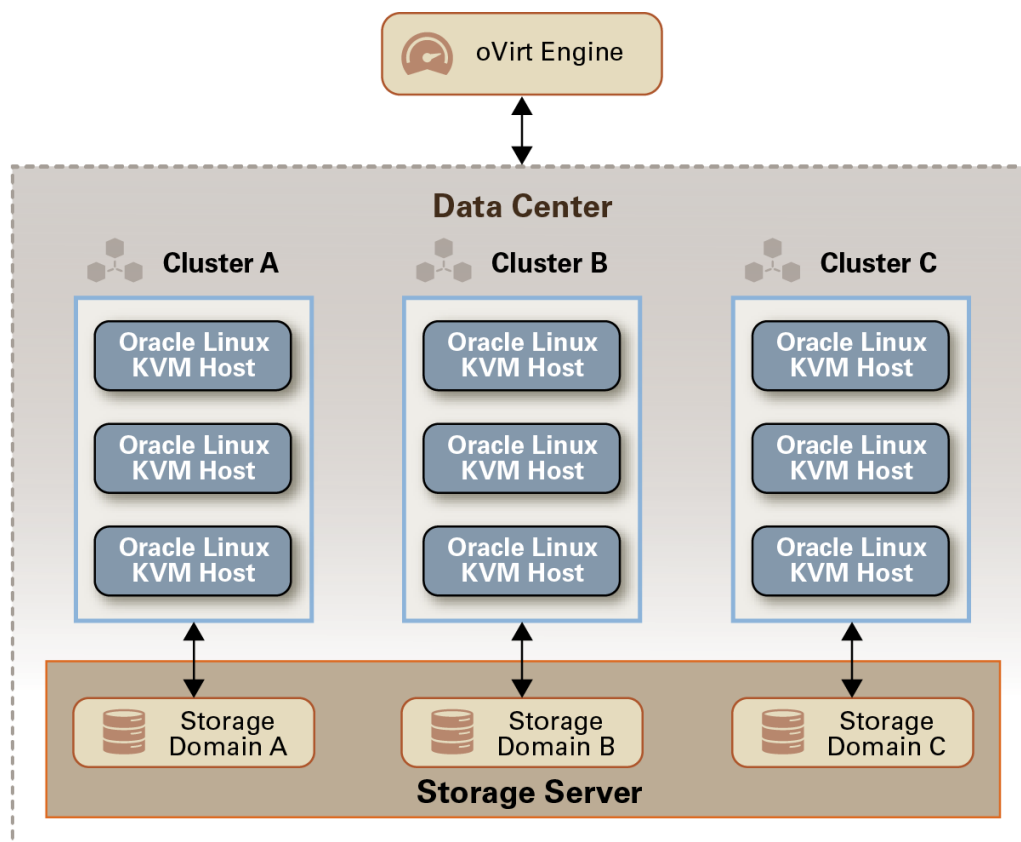


Figure 4-3 Multiple Clusters



Hosts

In Oracle Linux Virtualization Manager, you install Oracle Linux on a bare metal (physical) server and leverage the Unbreakable Enterprise Kernel, which allows the server to be used as a KVM hypervisor. When you are running a hypervisor on a server it is referred to as a **host** meaning it is capable of hosting virtual machines.

The engine host is a separate physical host and provides the administration tools for managing the Oracle Linux Virtualization Manager environment. All hosts in your environment must be Oracle Linux KVM hosts, except for the host running the engine which is an Oracle Linux hosts.

Oracle Linux Virtualization Manager can manage many Oracle Linux KVM hosts, each of which can run multiple virtual machines concurrently. Each virtual machine runs as individual Linux processes and threads on the KVM host.

Using the Administration Portal you can install, configure and manage your KVM hosts. You can also use the **Cockpit web interface** to monitor a KVM host's resources and perform administrative tasks. The Cockpit feature must be installed and enabled separately. You can access a host's Cockpit web interface from the Administration Portal or by connecting directly to the host.

The Virtual Desktop and Server Manager (VDSM) service is a host agent that runs as a daemon on the KVM hosts and communicates with the engine to:

- Manage and monitor physical resources, including storage, memory, and networks.
- Manage and monitor the virtual machines running on a host.
- Gather statistics and collect logs.

For more information on engine host and virtual machine requirements, see [Requirements and Scalability Limits](#) Requirements and Scalability Limits.

For more information, see [Host Architecture](#) and Adding a KVM Host in the [Oracle Linux Virtualization Manager: Getting Started Guide](#).

Virtual Machines

Virtual machines can be created to a certain specification or cloned from an existing template in the virtual machine pools. For more information, see [Creating a New Virtual Machine and Creating a Template](#) in the [Oracle Linux Virtualization Manager: Administration Guide](#). You can also import an Open Virtual Appliance (OVA) file into your environment from any host in the data center. For more information, see [oVirt Virtual Machine Management Guide](#) in [oVirt Documentation](#).

- A **virtual machine pool** is a group of on-demand virtual machines that are all clones of the same template. They are available to any user in a given group.

When accessed from the VM Portal, virtual machines in a pool are stateless, meaning that data is not persistent across reboots. Each virtual machine in a pool uses the same backing read-only image, and uses a temporary copy-on-write image to hold changed and newly generated data. Each time a virtual machine is assigned from a pool, it is allocated in its base state. Users who have been granted permission to access and use virtual machines from a pool receive an available virtual machine based on their position in a queue of requests.

When accessed from the Administration Portal, virtual machines in a pool are not stateless so that administrators can make changes to the disk if needed.

- **Guest agents and drivers** provide functionality for virtual machines such as the ability to monitor resource usage, shutdown and reboot the virtual machines from the Administration Portal.

! Important:

See *Windows Virtual Machines Lose Functionality Due To Deprecated Guest Agent* in the *Known Issues* section of the [Oracle Linux Virtualization Manager: Release Notes](#).

- A **snapshot** captures a virtual machine's operating system and applications on all available disks at a given point in time. Use a snapshot to restore a virtual machine to its previous state. For more information, see [Snapshots](#).
- A **template** is a copy of a virtual machine that you can use to simplify the subsequent, repeated creation of similar virtual machines. Templates capture the configuration of software, the configuration of hardware, and the software installed on the virtual machine on which the template is based, which is known as the source virtual machine.

Virtual machines that are created based on a template use the same NIC type and driver as the original virtual machine but are assigned separate, unique MAC addresses.

Considerations When Using Snapshots

A snapshot is a picture of a virtual machine's state and should not be used as a primary backup process. You take snapshots so you can revert the virtual machine to a specific point if and when required. Before creating snapshots, consider the following:

- Do not shutdown or start a virtual machine that displays an illegal status in the Administration Portal as this might cause data corruption or the virtual machine might fail to start.
- As soon as you revert to a snapshot and it is no longer required, delete the snapshot.
- Taking several snapshots in a row without any cleanup can affect virtual machine and host performance.
- When taking a snapshot, it creates a new copy of the virtual machine disk; so the more data you write to the snapshot, the longer it takes when using it.
- For an I/O intensive virtual machine, when deleting a snapshot remove it while the virtual machine is shut down (cold merge) instead of deleting the snapshot while the virtual machine is up (live merge).
- Ensure you have installed the latest guest agent package on your virtual machine before taking snapshots.

Virtual Machine Consoles

You access virtual machine consoles using the **Remote Viewer application** (`virt-viewer`) on Enterprise Linux and Microsoft Windows clients. Remote Viewer allows you to interact with a virtual machine in a similar way to a physical machine. For more information, see [Consoles](#).

To download Remote Viewer, see *Installing a Remote Viewer on Client Machine* in the [Oracle Linux Virtualization Manager: Administration](#). You must have Administrator privileges to install the Remote Viewer application.

High Availability and Optimization

You can configure Oracle Linux Virtualization Manager so that your cluster is optimized and your hosts and virtual machine are highly available. You can also enable or disable devices (hot plug) while a virtual machine is running.

For more information about high availability and optimization, see Deployment Optimization in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Clusters

Using the **Optimization** tab when creating or editing a cluster, you can select the memory page sharing threshold for the cluster, and optionally enable CPU thread handling and memory ballooning on the hosts in the cluster. Some of the benefits are:

- Virtual machines run on hosts up to the specified overcommit threshold. Higher values conserve memory at the expense of great CPU usage.
- Hosts can run virtual machines with a total number of CPU cores greater than the number of cores in the host.

- Memory overcommitment on virtual machines running on the hosts in the cluster.
- Memory Overcommitment Manager (MoM) runs Kernel Same-page Merging (KSM) when it can yield a memory saving benefit. To use KSM, you have to explicitly enable it at the cluster level.

You can set cluster optimization for the MoM to start ballooning where and when possible, with a limitation of the guaranteed memory size of every virtual machine. To have a ballooning running, a virtual machine needs to have a balloon device with relevant drivers. Each virtual machine includes a balloon device unless specifically removed. Each host in the cluster receives a balloon policy update when its status changes to **Up**. If necessary, you can manually update the balloon policy on a KVM host without having to change the status.

Hosts

If you want a cluster to be responsive when unexpected host failures happen, you should configure fencing. Fencing keeps hosts in a cluster highly available by enforcing any associated policies for power saving, load balancing, and virtual machine availability. If you want highly available virtual machines on a particular host:

- You must also enable and configure power management for the host
- The host must have access to the Power Management interface via the `ovirtmgmt` network

! Important:

For power management operations, you need at least two KVM hosts in a cluster or data center that are in *Up* or *Maintenance* status.

The Manager does not communicate directly with fence agents. Instead, the Engine uses a proxy to send power management commands to a host power management device. The Engine uses VDSM to execute power management device actions, so another host in the environment is used as a fencing proxy. You can select between:

- Any host in the same cluster as the host requiring fencing.
- Any host in the same data center as the host requiring fencing.

Each KVM host in a cluster has limited resources. If a KVM host becomes overutilized, there is an adverse impact on the virtual machines that are running on the host. To avoid or mitigate overutilization, you can use scheduling, load balancing, and migration policies to ensure the performance of virtual machines. If a KVM host becomes overutilized, virtual machines are migrated to another KVM host in the cluster.

Virtual Machines

A highly available virtual machine automatically migrates to and restarts on another host in the cluster if the host crashes or becomes non-operational. If a virtual machine is not configured for high availability it will not restart on another available host. If a virtual machine's host is manually shut down, the virtual machine does not automatically migrate to another host.

 **Note:**

Virtual machines do not live migrate unless you are using shared storage and have explicitly configured your environment for live migration in the event of host failures. Policies, such as power saving or distribution, as well as maintenance events trigger live migrations of virtual machines.

Using the Resource Allocation tab when creating or editing a virtual machine, you can:

- Set the maximum amount of processing capability a virtual machine can access on its host.
- Pin a virtual CPU to a specific physical CPU.
- Guarantee an amount of memory for the virtual machine.
- Enable the memory balloon device for the virtual machine. For this feature to work, memory balloon optimization must also be enabled for the cluster.
- Improve the speed of disks that have a VirtIO interface by pinning them to a thread separate from the virtual machine's other functions.

When a KVM host goes into maintenance mode, all virtual machines are migrated to other servers in the cluster. This means there is no downtime for virtual machines during planned maintenance windows.

If a virtual machine is unexpectedly terminated, it is automatically restarted, either on the same KVM host or another host in the cluster. This is achieved through monitoring of the hosts and storage to detect any hardware failures. If you configure a virtual machine for high availability and its host fails, the virtual machine automatically restarts on another KVM host in the cluster.

Policies

Load balancing, scheduling, and resiliency policies, enable critical virtual machines to be restarted on another KVM host in the event of hardware failure with three levels of priority.

Scheduling policies enable you to specify the usage and distribution of virtual machines between available hosts. You can define the scheduling policy to enable automatic load balancing across the hosts in a cluster. Regardless of the scheduling policy, a virtual machine does not start on a host with an overloaded CPU. By default, a host's CPU is considered overloaded if it has a load of more than 80% for 5 minutes, but these values can be changed using scheduling policies.

There are five default scheduling policies:

- `Evenly_Distributed` - evenly distributes the memory and CPU processing load across all hosts in a cluster.

 **Note:**

All virtual machines must have the latest `qemu-guest-agent` installed and its service running.

- `Cluster_Maintenance` - during maintenance tasks activity in a cluster is limited.

- **Power_Saving** - reduces power consumption on underutilized hosts by distributing memory and CPU processing load across a subset of available hosts.
- **VM_Evenly_Distributed** - evenly distributes virtual machines between hosts.
- **None**

Migration policies enable you to define the conditions for live migrating virtual machines in the event of KVM host failure. These conditions include how long a virtual machine can be down during migration, how much network bandwidth is used, and how the virtual machines are prioritized.

Resilience policies enable you to define how the virtual machines are prioritized in migration. You can configure the policy so that all or no virtual machines migrate, or that only highly available virtual machines migrate which helps to prevent overloading hosts.

For more information on policies, refer to the *Administration Guide* in [oVirt Documentation](#).

Networks

The following are general, high-level networking recommendations.

- Use bond network interfaces, especially on production hosts
- Use VLANs to separate different traffic types
- Use 1 GbE networks for management traffic
- Use 10 GbE, 25 GbE, 40 GbE, or 100 GbE for virtual machines and Ethernet-based storage
- When adding physical interfaces to a host for storage use, uncheck **VM network** so that the VLAN is assigned directly to the physical interface

The Oracle Linux Virtualization Manager host and all Oracle Linux KVM hosts must have a fully-qualified domain name (FQDN) as well as forward and reverse name resolution. Oracle recommend using DNS. Alternatively, you can use the `/etc/hosts` file for name resolution, however, this requires more work and is error-prone.

All DNS services used for name resolution must be hosted outside of the environment.

Logical Networks

In Oracle Linux Virtualization Manager, you configure logical networks to represent the resources required to ensure the network connectivity of the Oracle Linux KVM hosts for a specific purpose, for example to indicate that a network interface controller (NIC) is on a management network.

You define a logical network for a data center, apply the network to one or more clusters, and then configure the hosts by assigning the logical networks to the hosts physical interfaces. Once you implement the network on all the hosts in a cluster, the network becomes operational. You perform all these operations from the Administration Portal.

At the cluster level, you can assign one or more network roles to a logical network to specify its purpose:

- A **management network** is used for communication between Oracle Linux Virtualization Manager and the hosts.

- A **VM network** is used for virtual machine communication, a virtual machine's virtual NIC is attached to a VM network. For more information, see Creating a Logical Network in the [Oracle Linux Virtualization Manager: Administration Guide](#).
- A **display network** is used to connect clients to virtual machine graphical consoles, using either the VNC or RDP protocols.
- A **migration network** is used to migrate virtual machines between the hosts in a cluster.

By default a single logical network named **ovirtmgmt** is created and this is used for all network communication in a data center. You separate the network traffic according to your needs by defining and applying additional logical networks.

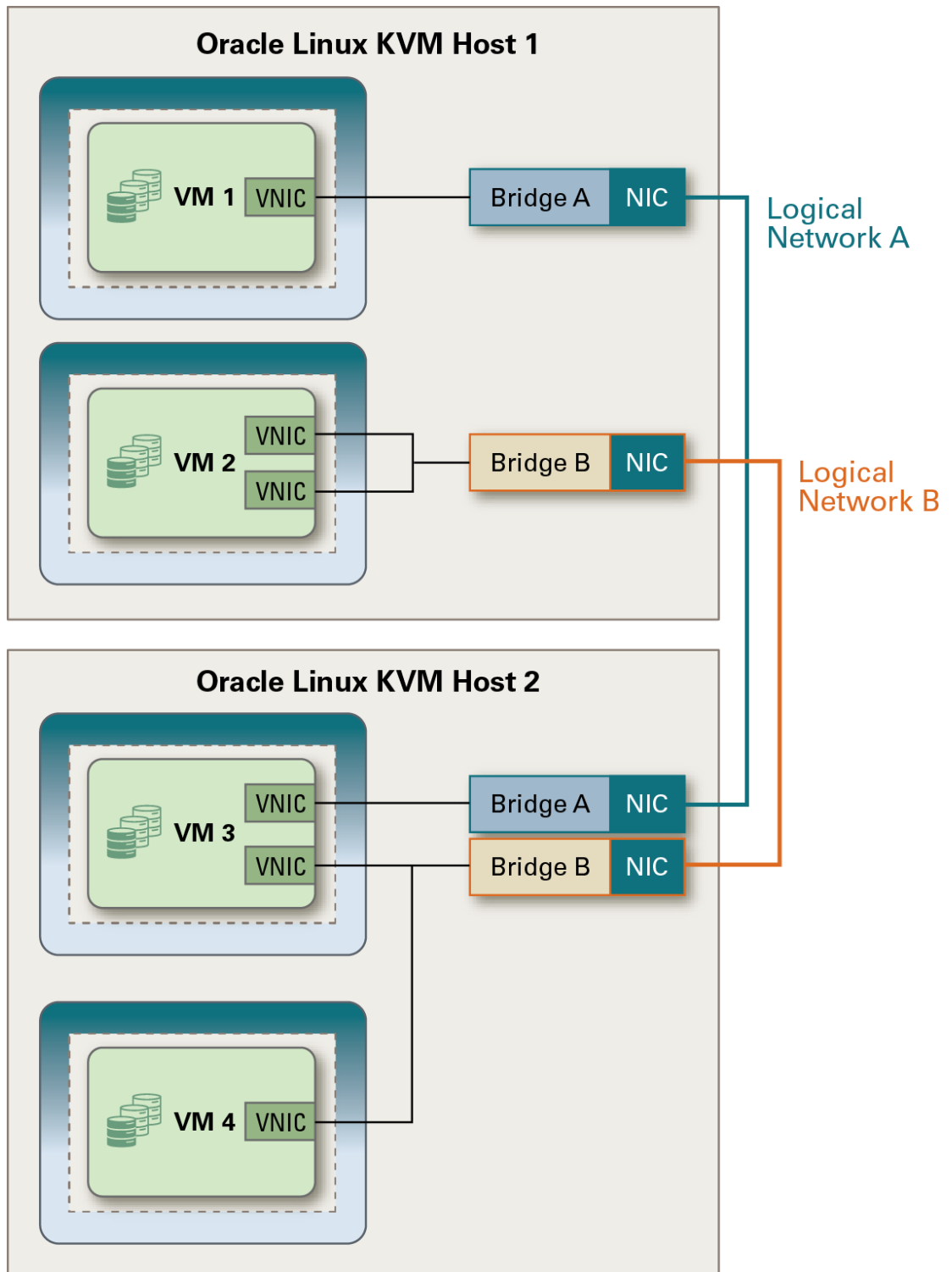
One logical network is configured as the default route for the hosts.

A logical network can be marked as a required network. If a required network ceases to function, any KVM hosts associated with the network become non-operational.

For logical networks that are **not** VM networks, you connect the host directly to the network using either a physical network interface, a VLAN interface, or a bond.

For VM networks, a bridge is created on the host for each logical network. Virtual machine VNICs are connected to the bridges as needed. The bridge is connected to the network using either a physical network interface, a VLAN interface, or a bond.

Figure 4-4 Bridge Networks



You can perform most network configuration operations on hosts from the Administration Portal, including:

- Assign a host NIC to logical networks.

- Configure a NIC's boot protocol, IP settings, and DNS settings.
- Create bonds and VLAN interfaces on KVM hosts.

When there are a large number of KVM hosts and logical networks, using network labels enables you to simplify administration. Labels can be applied to logical networks and host interfaces. When you set a label on a network, you to deploy the network on host NICs that have the same label. This requires that the host NICs are configured for DHCP.

VLANS

A virtual local area network (VLAN) enables hosts and virtual machines to communicate regardless of their actual physical location on a LAN.

VLANS enable you improve security by segregating network traffic. Broadcasts between devices in the same VLAN are not visible to other devices with a different VLAN, even if they exist on the same switch.

VLANS can also help to compensate for the lack of physical NICs on hosts. A host or virtual machine can be connected to different VLANs using a single physical NIC or bond. This is implemented using VLAN interfaces.

A VLAN is identified by an ID. A VLAN interface attached to a host's NIC or bond is assigned a VLAN ID and handles the traffic for the VLAN. When traffic is routed through the VLAN interface, it is automatically tagged with the VLAN ID configured for that interface, and is then routed through the NIC or bond that the VLAN interface is attached to.

The switch uses the VLAN ID to segregate traffic among the different VLANs operating on the same physical link. In this way, a VLAN functions exactly like a separate physical connection.

You need to configure the VLANs needed to support your logical networks before you can use them. This is usually accomplished using switch trunking. Trunking involves configuring ports on the switch to enable multiple VLAN traffic on these ports, to ensure that packets are correctly transmitted to their final destination. The configuration required depends on the switches you use.

When you create a logical network, you can assign a VLAN ID to the network. When you assign a host NIC or bond to the network, the VLAN interface is automatically created on the host and attached to the selected device.

Figure 4-5 VLANs

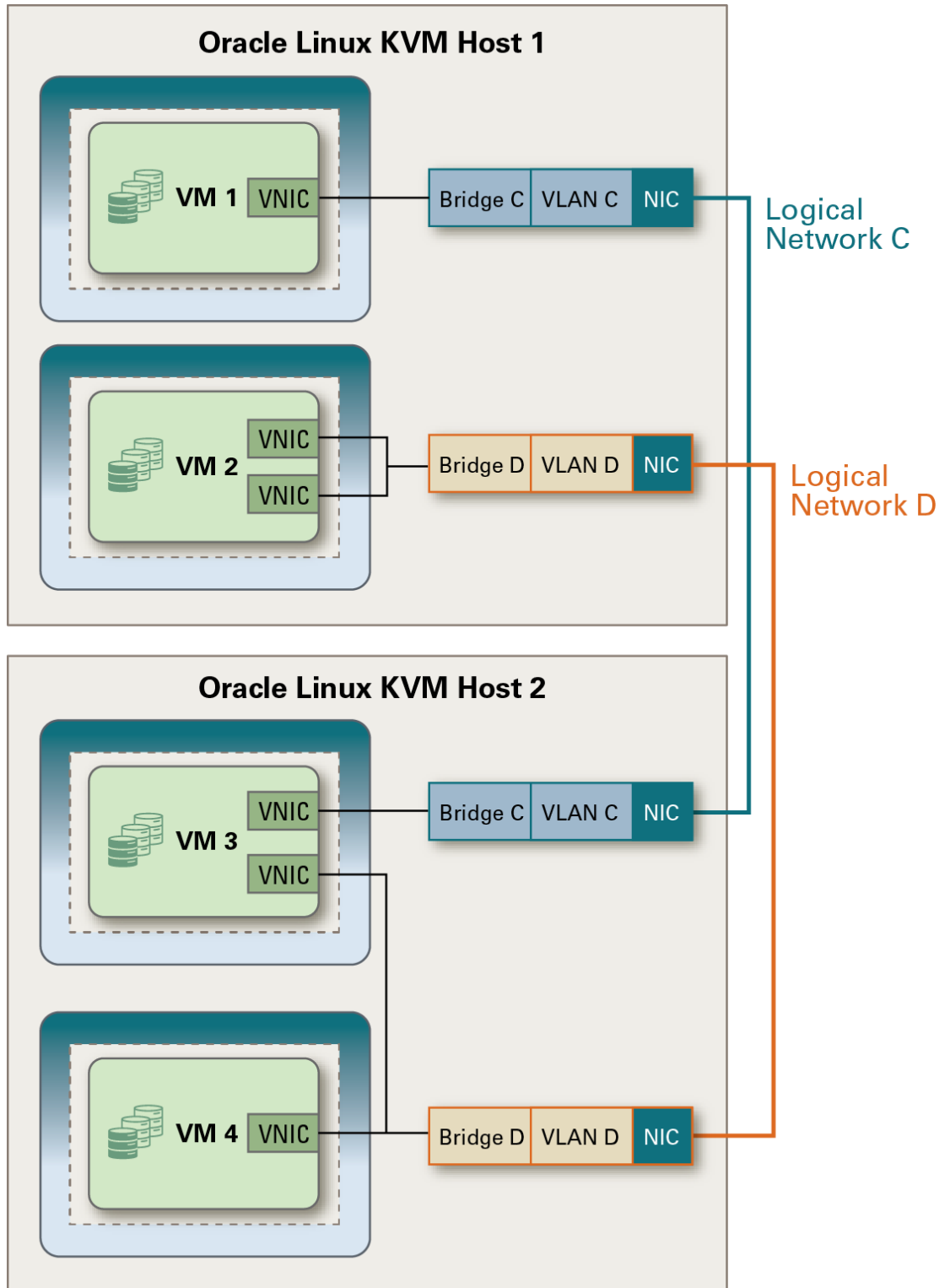
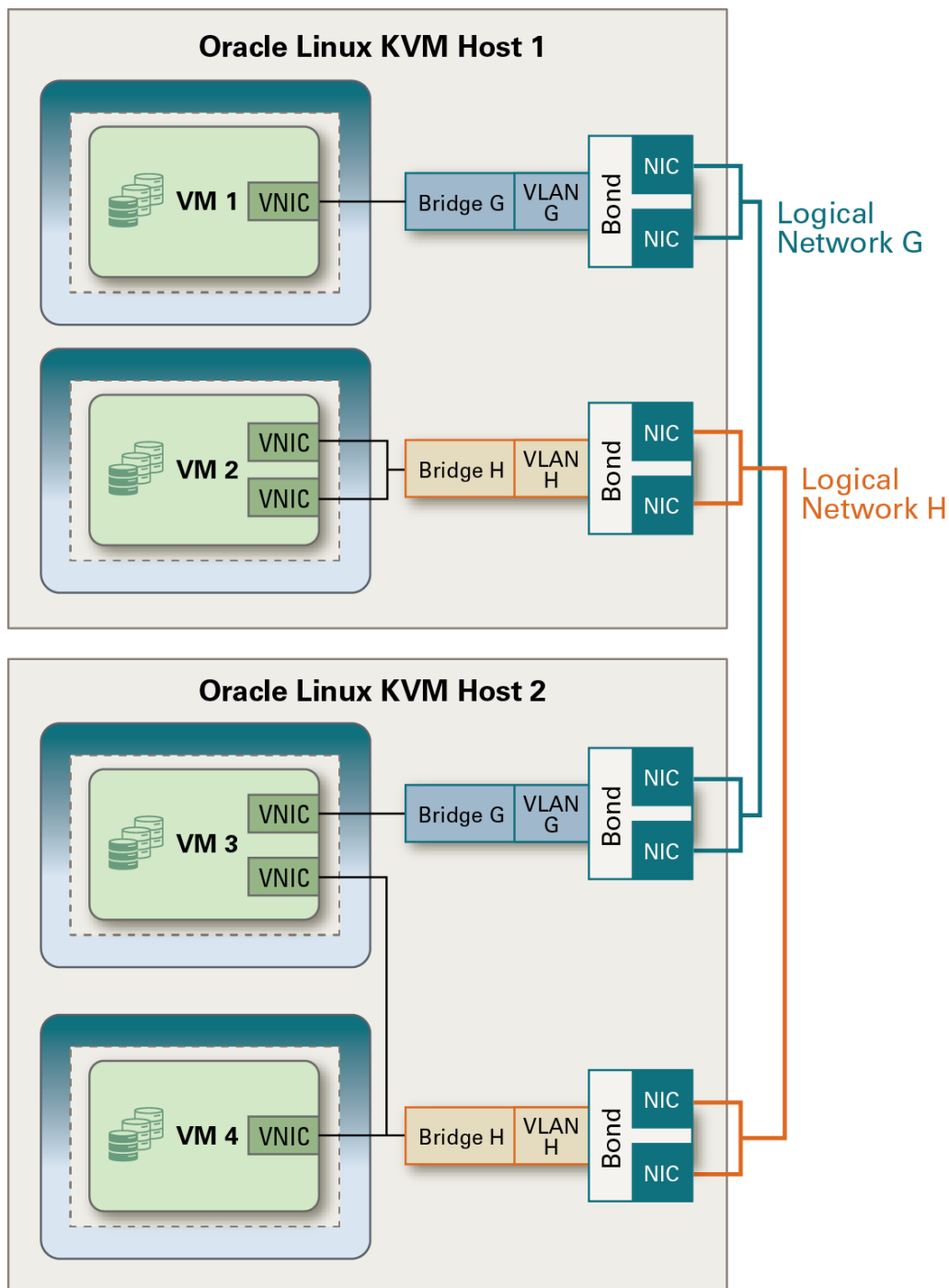


Figure 4-6 VLANs over Network Bonds



Virtual NICs

A virtual machine uses a virtual network interface controller (vNIC) to connect to a logical network.

VNICs are always attached to a bridge on a KVM host. A bridge is a software network device that enables the VNICS to share a physical network connection and to appear as separate physical devices on a logical network.

Oracle Linux Virtualization Manager automatically assigns a MAC address to a VNIC. Each MAC address corresponds to a single VNIC. Because MAC addresses must be unique on a network, the MAC addresses are allocated from a predefined range of addresses, known as a MAC address pool. MAC address pools are defined for a cluster.

Virtual machines are connected to a logical network by their VNICs. The IP address of each VNIC can be set independently, by DHCP or statically, using the tools available in the operating system of the virtual machine. To use DHCP, you need to configure a DHCP server on the logical network.

Virtual machines can communicate with any other machine on the virtual network, and, depending on the configuration of the logical network, with public networks such as the Internet.

For more information, see Customizing vNIC Profiles for Virtual Machines in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Bonds

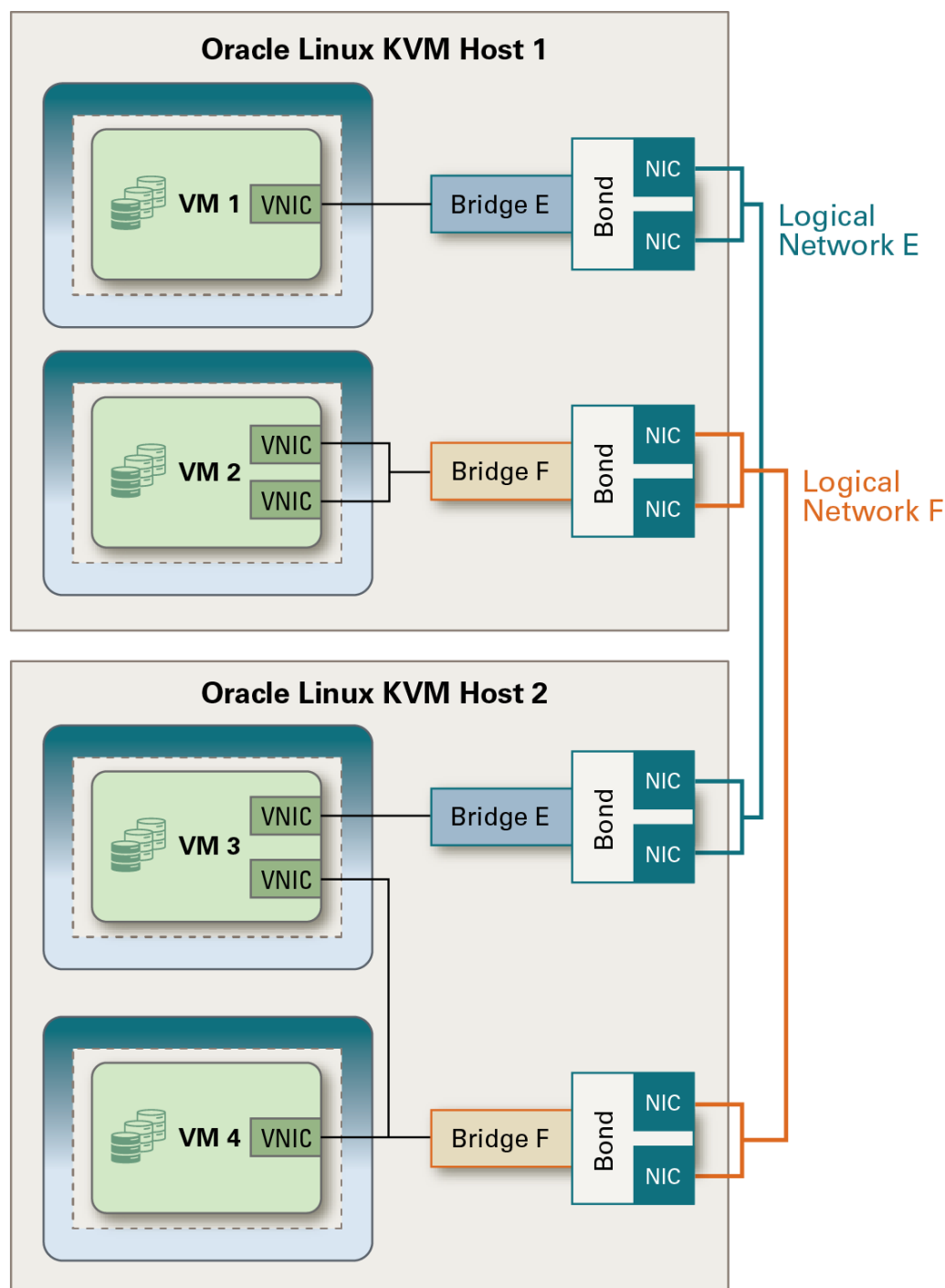
Bonds bind multiple NICs into a single interface. A bonded network interface combines the transmission capability of all the NICs included in the bond and acts as a single network interface, which can provide greater transmission speed. Because all network interface cards in the bond must fail for the bond itself to fail, bonding provides increased fault tolerance.

There are four different bonding modes:

- Mode 1 - Active-Backup
- Mode 2 - Load balance XOR Policy
- Mode 3 - Broadcast
- Mode 4 (default) - Dynamic link aggregation IEEE 802.3ad

Bonding modes 2 and 4 require static etherchannel enabled (not LACP-negotiated) and LACP-negotiated etherchannel enabled on physical switches respectively.

Figure 4-7 Network Bonds



MAC Address Pools

MAC address pools define the range (or ranges) of MAC addresses allocated for each cluster. A MAC address pool is specified for each cluster. By using MAC address pools, the Manager can automatically generate and assign MAC addresses to new virtual network devices, which helps to prevent MAC address duplication. MAC

address pools are more memory efficient when all MAC addresses related to a cluster are within the range for the assigned MAC address pool.

The same MAC address pool can be shared by multiple clusters, but each cluster has a single MAC address pool assigned. A default MAC address pool is created by the Manager and is used if another MAC address pool is not assigned.

 **Note:**

If more than one cluster shares a network, you should not rely solely on the default MAC address pool because the virtual machines in each cluster attempt to use the same range of MAC addresses, which can lead to conflicts. To avoid MAC address conflicts, check the MAC address pool ranges to ensure that each cluster is assigned a unique MAC address range.

The MAC address pool assigns the next available MAC address after the last address that is returned to the pool. If there are no further addresses left in the range, the search starts again from the beginning of the range. If there are multiple MAC address ranges with available MAC addresses defined in a single MAC address pool, the ranges take turns in serving incoming requests in a similar manner as when MAC addresses are selected.

Storage

Oracle Linux Virtualization Manager uses a centralized storage system for virtual machine disk images, ISO files and snapshots. You can use Network File System (NFS), Internet Small Computer System Interface (iSCSI), Fibre Channel Protocol (FCP), or Gluster FS storage. You can also configure local storage attached directly to hosts. For more information, see Storage in the [Oracle Linux Virtualization Manager: Administration Guide](#).

A data center cannot be initialized unless a storage domain is attached to it and activated.

The storage must be located on the same subnet as the Oracle Linux KVM hosts that will use the storage, in order to avoid issues with routing.

Since you need to create, configure, attach and maintain storage, make sure you are familiar with the storage types and their use. Read your storage array manufacturer guides for more information.

Storage Domains

A storage domain is a collection of images that have a common storage interface. A storage domain contains complete images of templates, virtual machines, virtual machine snapshots, or ISO files. Oracle Linux Virtualization Manager supports storage domains that are block devices (SAN - iSCSI or FCP) or a file system (NAS - NFS or Gluster).

On NFS or Gluster, all virtual disks, templates, and snapshots are files. On SAN (iSCSI/FCP), each virtual disk, template or snapshot is a logical volume.

Virtual machines that share the same storage domain can be migrated between hosts that belong to the same cluster.

Storage, also referred to as a **data domain**, is used to store the virtual hard disks, snapshots, ISO files, and Open Virtualization Format (OVF) files for virtual machines and templates.

Every data center must have at least one data domain. Data domains cannot be shared between data centers.

**Note:**

The Administration Portal currently offers options for creating storage domains that are export domains or ISO domains. These options are deprecated.

Detaching a storage domain from a data center stops the association, but does not remove the storage domain from the environment. A detached storage domain can be attached to another data center. And, the data, such as virtual machines and templates, remains attached to the storage domain.

Storage Pool Manager

The Storage Pool Manager (SPM) is a management role assigned to one of the hosts in a data center enabling it to manage the storage domains of the data center. Any host in the data center can run the SPM entity, which is assigned by the engine. SPM controls access to storage by coordinating the metadata across the storage domains. This includes creating, deleting, and manipulating virtual disks (images), snapshots, and templates, and allocating storage for sparse block devices (on SAN).

The host running as SPM can still host virtual resources. The SPM priority setting for hosts enables you to prioritize which host is assigned the SPM role. Since the SPM role uses some of the host's available resources, it is important to prioritize hosts that can afford the resources.

Because the SPM must always be available, the engine assigns the SPM role to another host if the SPM host becomes unavailable. A host with higher SPM priority is assigned the SPM role before a host with lower SPM priority.

Virtual Machine Storage

The Storage Pool Manager (SPM) is responsible for creating and deleting virtual disks, as well as snapshots, and templates. In addition it allocates storage for sparse block devices.

- If you are using NFS or local storage, the SPM creates a thin provisioned virtual disk by default.
- If you are using iSCSI storage or other block-based devices, Logical Unit Numbers (LUNs) are provided to the SPM. Then, a volume group on top of the LUNs and logical volumes for use as virtual machine disks are created and the SPM preallocates the space by default.
- If a virtual disk is thinly-provisioned, a 1 GB logical volume is created with a QCOW2 format. Use thin provisioning for virtual machines with low I/O requirements.
- The virtual machine's host continuously monitors the logical volume used for its virtual disk. You can set a threshold so that when the disk usage nears the threshold the host notifies the SPM and extends the logical volume by 1 GB.

- If the storage in a pool starts to become exhausted, a new LUN can be added to the volume group. The SPM automatically distributes the additional storage to logical volumes that need it.
- If a virtual disk is preallocated, a logical volume of the specified size in GB and a virtual disk of RAW format is created. Use preallocated disks for virtual machines with high levels of I/O. Preallocated disks cannot be enlarged.
- If an application requires storage to be shared between virtual machines, use **Shareable** virtual disks which can be attached to multiple virtual machines concurrently.

QCOW2 format virtual disks cannot be shareable. You cannot take a snapshot of a shared disk and virtual disks that have snapshots that cannot be marked shareable. You cannot live migrate a shared disk.

If the virtual machines are not cluster-aware, mark shareable disks as read-only to avoid data corruption.

- Use direct LUN to enable virtual machines to directly access RAW block-based storage devices on the host bus adapter (HBA). The mapping of the direct LUN to the host causes the storage to be emulated as file-based storage to virtual machines. This removes a layer of abstraction between virtual machines and their data as the virtual machine is being granted direct access to block-based storage LUNs.

Storage Leases

When you add a storage domain to Oracle Linux Virtualization Manager, a special volume is created called **xleases**. Virtual machines are able to acquire a lease on this special volume, which enables the virtual machine to start on another host even if the original host loses power.

A storage lease is configured automatically for the virtual machine when you select a storage domain to hold the VM lease. (See [Configuring a Highly Available Virtual Machine in the Oracle Linux Virtualization Manager: Administration Guide](#).) This triggers a **create a new lease** request to the engine which then send the request to the SPM. The SPM creates a lease and a lease id for the virtual machine on the xreleases volume. VDSM creates the sanlock which is used to acquire an exclusive lock on a virtual disk.

The lease id and other information is then sent from the SPM to the engine. The engine then updates the virtual machine's device list with the lease information.

Local Storage

Local storage is storage that is attached directly to an Oracle Linux KVM host, such as a local physical disk or a locally attached SAN. When a KVM host is configured to use local storage, it is automatically added to a cluster where it is the only host. This is because clusters with multiple hosts must have shared storage domains accessible to all hosts.

When you use local storage, features such as live migration, scheduling, and fencing are not available.

For more information, see [Configuring a KVM Host to Use Local Storage in the Oracle Linux Virtualization Manager: Administration Guide](#).

System Backup and Recovery

You use the `engine-backup` tool to take regular backups of the Oracle Linux Virtualization Manager. The tool backs up the `engine` database and configuration files into a single file and can be run without interrupting the `ovirt-engine` service.

You also use the `engine-backup` tool to restore a backup. However, the steps you need to take can be more involved depending on your restoration destination. For example, the `engine-backup` tool can be used to restore backups to fresh installations of Oracle Linux Virtualization Manager, on top of existing installations of Oracle Linux Virtualization Manager, and using local or remote databases.

If you restore a backup to a fresh installation of Oracle Linux Virtualization Manager, you do not run the `engine-setup` command to configure the Manager.

You can also use data center recovery if the data in your primary data domain gets corrupted. This enables you to replace the primary data domain of a data center with a new primary data domain.

Reinitializing a data center enables you to restore all other resources associated with the data center, including clusters, hosts, and storage domains. You can import any backup or exported virtual machines or templates into the new primary data domain.

For more information, see Backup and Restore in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Users, Roles, and Permissions

In Oracle Linux Virtualization Manager, there are two types of user domains: local domain and external domain. During the installation of the Manager, a default local domain called the `internal` domain is created with a default `admin@internal` user. This account is intended for use when initially configuring the environment and for troubleshooting.

You can create additional users on the `internal` domain using `ovirt-aaa-jdbc-tool` command utility. For more information about creating users, see Administering User and Group Accounts from the Command Line in the [Oracle Linux Virtualization Manager: Administration Guide](#).

User properties consist of the roles and permissions assigned to a user. The security roles for all actions and objects in the platform are granular, inheritable, and provide for multi-level administration.

Roles are sets of permissions defined in the Administration Portal and are used to specify permissions to resources in the environment. There are two types of roles:

- **Administrator Role** - Conveys management permissions of physical and virtual resources through the Administration Portal. Examples of roles within this group are SuperUser, ClusterAdmin and DataCenterAdmin.
- **User Role** - Conveys permissions for managing and accessing virtual machines and templates through the VM Portal by filtering what is visible to a user. Roles can be assigned to the users for individual resources, or levels of objects. Examples of roles within this group are UserRole, PowerUserRole and UserVmManager.

It is possible to create new roles with specific permissions applicable to a user's role within the environment. It is also possible to remove specific permissions to a resource from a role assigned to a specific user.

You can also use an external directory server to provide user account and authentication services. You can use Active Directory, OpenLDAP, and 389ds. Use the `ovirt-engine-extension-aaa-ldap-setup` command to configure the connection to these directories.

 **Note:**

After you have attached an external directory server, added the directory users, and assigned them with appropriate roles and permissions, the `admin@internal` user can be disabled if it is not required. For more information, see [Disabling User Accounts](#) in the [Oracle Linux Virtualization Manager: Administration Guide](#).

For more information on users, roles, and permissions, see [Global Configuration](#) in the [Oracle Linux Virtualization Manager: Administration Guide](#).

System State and History

When you install and configure Oracle Linux Virtualization Manager, you are prompted to install and configure the engine and data warehouse PostgreSQL databases. See [Engine Configuration Options](#) in the [Oracle Linux Virtualization Manager: Getting Started Guide](#).

- The engine database (`engine`) stores information about the state of the Oracle Linux Virtualization Manager environment and its configuration and performance.
- The data warehouse database is a management history database (`ovirt_engine_history`) that can be used by any application to retrieve historical configuration information and statistical metrics for data centers, clusters, and hosts.

The data warehouse service (`ovirt-engine-dwd`) extracts data from the `engine` database and loads it into the `ovirt_engine_history` database. This is commonly known as ETL (extract, transform, load).

Both the history and engine databases can run on a remote host to reduce the load on the Manager host. Running these databases on a remote host is a technology preview feature, see [Technology Preview](#) in the [Oracle Linux Virtualization Manager: Release Notes](#).

For more information, see [Data Warehouse and Databases](#).

Event Logging and Notifications

Oracle Linux Virtualization Manager captures events in the following log files:

- `/var/log/ovirt-engine/engine.log` contains all Oracle Linux Virtualization Manager UI crashes, Active Directory lookups, database issues, and other events.
- `/var/log/vdsm/vdsm.log` is the log file for VDSM, the engine's agent on the virtualization host(s), and contains host-related events.

Within the Administration Portal, you can also view Alerts and Events in the Notification Drawer, which you can access by clicking Bell icon in the upper-right corner.

The `ovirt-log-collector` tool enables you to collect relevant logs from across the environment. To use the tool, you must log into the Oracle Linux Virtualization Manager host as the root user and log into the Administration Portal with administration credentials.

The tool collects all logs from the Manager host, the Oracle Linux KVM hosts it manages, and the database.

Oracle Linux Virtualization Manager provides event notification services that allow you to configure the Engine to notify designated users by email when certain events occur or to send Simple Network Management Protocol (SNMP) traps to one or more external SNMP manager with system event information to monitor your virtualization environment.

For more information about configuring event notifications, see Using Event Notifications in the [Oracle Linux Virtualization Manager: Administration Guide](#).

Data Visualization with Grafana

Grafana is a web-based tool used to display data collected from the data warehouse database (`ovirt_engine_history` database). Data from the Engine is collected every minute and aggregated in hourly and daily aggregations. Data retention is defined during engine setup in the scale setting of the data warehouse configuration. The sample scaling can be set as:

- **Basic** (default) - samples data saved for 24 hours, hourly data saved for 1 (one) month, and daily data; no daily aggregations are saved.
- **Full** (recommended)- samples data saved for 24 hours, hourly data saved for 2 (two) months and daily aggregations saved for 5 (five) years.

Note:

Full sample scaling may require migrating the data warehouse to a separate virtual machine. For more information, see the [oVirt Data Warehouse Guide](#).

For information on configuring Oracle Linux Virtualization Manager for Grafana and the default dashboards, see Using Grafana in the [Oracle Linux Virtualization Manager: Administration Guide](#).

For more information on using Grafana, see the [Grafana website](#).

Default Grafana Dashboards

The following dashboards are available by default in the initial Grafana setup to report data center, cluster, host, and virtual machine data.

Executive

- *System*
Resource usage and up-time for hosts and storage domains in the system, according to the latest configurations.
- *Data Center*

Resource usage, peaks, and up-time for clusters, hosts, and storage domains in a selected data center, according to the latest configurations.

- *Cluster*
Resource usage, peaks, over-commit, and up-time for hosts and virtual machines in a selected cluster, according to the latest configurations.
- *Host*
Latest and historical configuration details and resource usage metrics of a selected host over a selected period.
- *Virtual Machine*
Latest and historical configuration details and resource usage metrics of a selected virtual machine over a selected period.
- *Executive*
User resource usage and number of operating systems for hosts and virtual machines in selected clusters over a selected period.

Inventory

- *Inventory*
Number of hosts, virtual machines, and running virtual machines, resources usage and over-commit rates for selected data centers, according to the latest configurations.
- *Hosts Inventory*
FQDN, VDSM version, operating system, CPU model, CPU cores, memory size, create date, delete date, and hardware details for selected hosts, according to the latest configurations.
- *Storage Domains Inventory*
Domain type, storage type, available disk size, used disk size, total disk size, creation date, and delete date for selected storage domains over a selected period.
- *Virtual Machines Inventory*
Template name, operating system, CPU cores, memory size, create date, and delete date for selected virtual machines, according to the latest configurations.

Service Level

- *Uptime*
Planned downtime, unplanned downtime, and total time for the hosts, high availability virtual machines, and all virtual machines in selected clusters in a selected period.
- *Hosts Uptime*
Uptime, planned downtime, and unplanned downtime for selected hosts in a selected period.
- *Virtual Machines Uptime*
Uptime, planned downtime, and unplanned downtime for selected virtual machines in a selected period.
- *Cluster Quality of Service*
 - *Hosts*
Time selected hosts have performed above and below the CPU and memory threshold in a selected period.
 - *Virtual Machines*
Time selected virtual machines have performed above and below the CPU and memory threshold in a selected period.

Trend

- *Trend*
Usage rates for the 5 most and least utilized virtual machines and hosts by memory and by CPU in selected clusters over a selected period.
- *Hosts Trend*
Resource usage (number of virtual machines, CPU, memory, and network Tx/Rx) for selected hosts over a selected period.
- *Virtual Machines Trend*
Resource usage (CPU, memory, network Tx/Rx, disk I/O) for selected virtual machines over a selected period.
- *Hosts Resource Usage*
Daily and hourly resource usage (number of virtual machines, CPU, memory, network Tx/Rx) for selected hosts in a selected period.
- *Virtual Machines Resource Usage*
Daily and hourly resource usage (CPU, memory, network Tx/Rx, disk I/O) for selected virtual machines in a selected period.