Technical Report

Red Hat Enterprise Linux 6, KVM, and NetApp Storage: Best Practices Guide

Jon Benedict, NetApp
March 2012 | TR-3848

Summary

NetApp® technology enables data centers to extend their virtual infrastructures to include the benefits of advanced storage virtualization. NetApp unified storage platforms offer industry-leading technologies in the areas of storage efficiencies, instantaneous VM and datastore cloning for virtual servers, and virtual data center backup and business continuance solutions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 RHEL KVM on NetApp

1.1 Overview

NetApp® technology enables data centers to extend their virtual infrastructures to include the benefits of advanced storage virtualization. NetApp unified storage platforms offer industry-leading technologies in the areas of storage efficiencies, instantaneous VM and datastore cloning for virtual servers, and virtual data center backup and business continuance solutions.

Although Red Hat is a relative newcomer to the virtualization market, it supports the high performance and open source Kernel-Based Virtual Machine (KVM) hypervisor.

Red Hat® Enterprise Linux® (RHEL) also offers the benefits of flexible deployment:

- It can be deployed as a bare-metal operating system, as a hypervisor, or as a virtual guest operating system.
- From a storage perspective, RHEL KVM supports both SAN (iSCSI, FC, FCoE) and NAS (NFS) for shared virtual machine (VM) storage.

NetApp and Red Hat maintain a long-term strategic alliance that includes end-to-end solution testing between Red Hat products and NetApp storage. As a result of this testing, NetApp has developed operational guidelines and best practices for storage arrays running the NetApp Data ONTAP® operating system in support of Red Hat Enterprise Linux. These guidelines have been extended to include RHEL-based KVM virtualization.

1.2 Intended Audience

This document addresses the needs of system architects, system administrators, and storage administrators who are investigating the use of KVM with RHEL 6 on NetApp storage.

1.3 Scope

This document focuses on KVM as deployed with Red Hat Enterprise Linux 6. Additionally, because KVM is available on many Linux distributions, this document specifically references it as “Red Hat KVM” to differentiate it from KVM as deployed on other Linux distributions.

1.4 Topics Out of Scope

Best practices associated with IP and Fibre Channel networks are not covered in this document. However, a good understanding of these topics is necessary for configuring items such as VLANs, switched fabrics, and other related technologies.

Why Deploy Red Hat KVM with NetApp Storage?

The virtualization of a data center means that physical systems are virtualized as part of a cost-saving effort to reduce both capital expenditures and operating expenses through infrastructure consolidation and increased operational efficiencies. These efforts result in multiple VMs sharing physical resources, including shared storage pools known as *datastores.* Virtualizing demanding business-critical applications such as e-mail and database servers results in increased operational efficiency. This group of systems may share server resources, but it is typically configured with exclusive access to the required storage.

Red Hat and NetApp both offer technologies that natively support multiple storage protocols. These technologies allow customers to deploy best-in-class virtual data centers that leverage the strengths inherent when using these technologies together. This is not a storage area network (SAN) versus network-attached storage (NAS) discussion, but rather a consideration of the operational value based on the type of storage network interconnect available to a virtual data center.

Whether the storage network is FC or Ethernet (NFS, iSCSI, and FCoE), these technologies combine with NetApp storage to scale the largest consolidation efforts and virtualize the most demanding applications without sacrifice or the need to deploy separate hardware to meet the needs of either environment. This virtualization is valuable in a storage array platform.

Essentially, NetApp provides a deep level of storage virtualization that complements the server virtualization offered by Red Hat KVM.

The 80/20 Rule

In designing the storage architecture for a virtual data center, the 80/20 rule applies: 80% of all systems virtualized are for consolidation efforts. The remaining 20% are classified as business-critical applications. Although the 20% can be virtualized successfully, they tend to be deployed on shared storage pools, refers to as *isolated datasets.*

Consolidated datasets have the following characteristics:

- VMs that do not require application-specific backup and restore agents; a “crash-consistent” Snapshot™ copy of the underlying NetApp volume is sufficient
- They typically contain the largest number of virtual machines
- The virtual machines do not typically have large network or storage I/O requirements, but collectively they might represent a resolvable challenge
- Consolidated datasets are ideally served by large, shared, policy-driven storage pools (datastores)

Isolated datasets (for business-critical applications) have the following characteristics:

- VMs that require application-specific backup and restore agents.
- Each individual VM may address a large amount of storage and/or have high I/O requirements.
- Storage design and planning are applied in a similar manner as they are applied to the physical servers.
- Datasets are ideally served by individual, high-performing, and nonshared datastores.

Consolidated datasets work well with NFS datastores because this design is more flexible in terms of capacity than SAN datastores when managing hundreds or thousands of VMs. Isolated datasets run well on all storage protocols; however, some tools or applications may have restricted compatibility with NFS and/or Red Hat supported file systems on SAN datastores.

In most cases, the data center evolution from physical to virtual follows the 80/20 rule, and the native multiprotocol capabilities of NetApp and Red Hat KVM make it faster and easier to virtualize systems than with a traditional storage array platform or multiple disparate storage arrays.

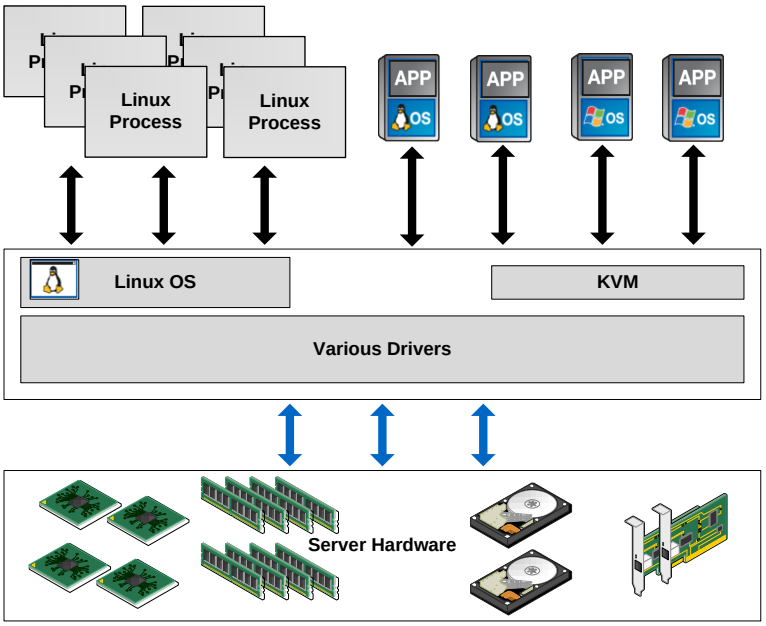
2 History of the KVM Hypervisor

The founders of Qumranet created the Kernel-Based Virtual Machine (KVM) hypervisor in the fall of 2006. At that time, most hypervisors existed as an additional layer on top of the operating system. Traditional hypervisors provided virtualization capability, but they also duplicated the efforts of memory and I/O management.

In contrast, KVM is a loadable kernel module that turns the Linux kernel into a hypervisor. It was accepted by the upstream Linux kernel maintainers in January of 2007 and is now shipped in almost all modern Linux distributions.

In September of 2008, Red Hat purchased Qumranet along with all of the products and protocols that would become Red Hat Enterprise Virtualization (RHEV). Although Red Hat is the largest contributor to KVM, the underlying KVM hypervisor for RHEV is still open source.

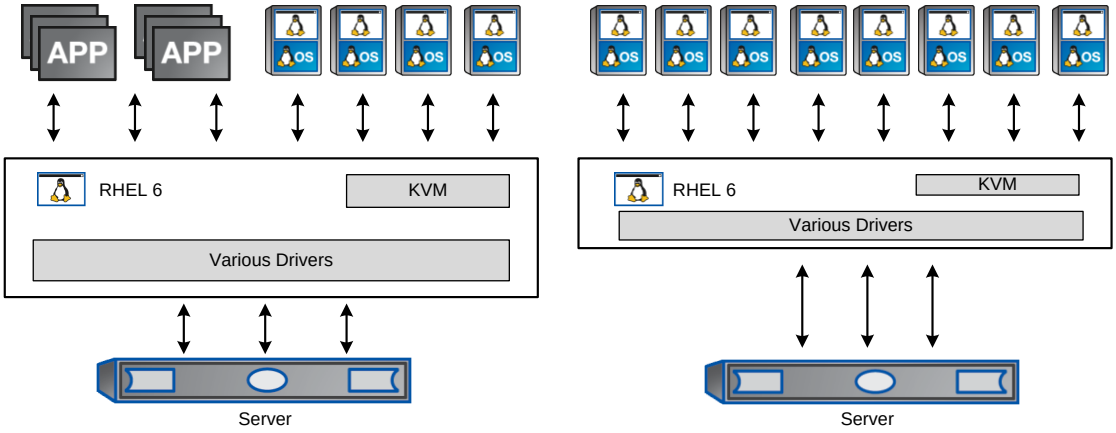
Figure 1) KVM architecture.



Thick and Thin Hypervisors

In the context of Red Hat, KVM can be deployed in one of two ways: a “thick” hypervisor, as deployed on Red Hat Enterprise Linux 6, or a “thin” hypervisor, as deployed in RHEV-H. Both thick and thin deployments are considered “type 1” hypervisors that run on bare metal.

Figure 2) Thick and thin hypervisors.



Thick (Full OS) Hypervisor RHEL 6

Thin Hypervisor RHEV-H

Although both thick and thin hypervisors can be managed by RHEV-M, only RHEV-H depends on RHEV-M. The means of deployment, management, and integration are different when comparing thick and thin hypervisors, and there are differences in support subscriptions. The KVM hypervisor is available in non-Red Hat distributions. Considering these differences, it is incorrect to use the terms “KVM” and “RHEV” interchangeably.

3 Red Hat Enterprise Linux 6 and KVM Installation

3.1 CPU and Memory Considerations

In addition to the CPU and memory requirements published by Red Hat, here are some other points to consider:

- For a production environment, multicore and multisocket CPUs should be used to provide the highest number of available virtual CPUs to virtual machines.
- The more physical RAM available on the Red Hat KVM host, the more virtual memory is available to the virtual machines. Preferably, use a high memory footprint (>24GB RAM).

3.2 Hardware Requirements for Red Hat KVM

Red Hat KVM hosts have the following hardware requirements:

- A 64-bit CPU with the hardware virtualization extensions. This means an AMD system with AMD-V or an Intel® system with Intel VT; maximum of 160 logical CPUs (the theoretical limit is 4,096).
- At least one network controller with a minimum bandwidth of 1Gbps. NetApp and Red Hat best practices specify at least two network controllers; 10 Gigabit Ethernet (10GbE) is preferred for storage networks.
- At least 2GB of RAM for the Red Hat KVM host plus sufficient RAM for VMs, depending on guest OS requirements and workloads. 2GB is the minimum, and NetApp strongly recommends that additional RAM be available for virtualization — a maximum of 2TB host RAM (the theoretical limit is 64TB).
- 6GB of disk space plus the required space for each guest operating system.
- Shared storage to support advanced virtualization capabilities (live migration, copy offload, clone offload, and so on).

Network Cards and HBAs

A Red Hat KVM host should have at least one onboard 1GbE network card for management traffic as well as at least two 10GbE ports for storage traffic. Additionally, some form of out-of-band ports should be available for power management and remote access. Multiple 1GbE network cards can be used if 10GbE is not available.

When attaching to FC SAN, at least one FC host bus adapter (HBA) is required. Hardware-based iSCSI HBAs are recommended, although software-based initiators are supported.

Boot from SAN

Red Hat KVM hosts should be installed to and boot from SAN (FC, FCoE, or iSCSI). This offers multiple benefits in the efficient use of storage, host mobility, disaster recovery, and backup. Boot LUNs on the NetApp controller can easily be recovered, deduplicated, and in some cases migrated.

Disk Layout for Red Hat KVM Hosts

The disk layout for Red Hat KVM hosts should be balanced between Red Hat best practices and the requirements of the data center that is hosting the virtual environment. It is important to follow Red Hat recommendations for swap, which depend on the size of the physical RAM. It is also a best practice to have the `/boot` and `/var` directories on separate partitions. Separate partitions for `/home` and other major directories are not required for the host nodes.

Earlier versions of Red Hat Enterprise Linux required file system alignment to be considered; RHEL 6 aligns properly by default.

3.3 Package Selection

The packages installed on a Red Hat KVM host should be selected carefully. Although graphical interface packages can be installed, the best practice is to avoid graphical environments on the Red Hat KVM host. If a graphical interface is required, NetApp recommends using a separate remote host that has the graphical packages installed. This makes sure that all available resources (CPU, RAM, and I/O) are available to support virtual machines.

Other packages to avoid on a Red Hat KVM host include development libraries, network sniffers, and any services that are not needed. A rule of thumb is that if it does not support, service, or protect virtual machines, then do not install it. This practice leaves more resources available to virtual machines, and there are fewer packages to update and fewer security concerns.

3.4 KVM Host Node Configuration

Network Configuration to Support Virtual Machines

By default, KVM creates a single virtual bridge on the host nodes that allows the virtual guests to communicate with each other and the outside world. The default virtual bridge provides IP addresses (through internal DHCP) to the KVM guests. The default virtual bridge is functional, but very basic; it does not allow hosts outside of the host node to reach back to the KVM guests.

The best practice is to extend the virtual bridge configuration to create at least one additional virtual public bridge. This involves taking over a physical interface and editing the standard NIC configuration file to reflect a bridge instead of an Ethernet device.

This practice also requires configuring the iptables firewall to forward all virtual machine traffic through the virtual bridge.

KVM Host Time Configuration

Network Time Protocol (NTP) must be enabled and configured on every Red Hat KVM host to keep the time synchronized across the KVM environment. If a remote host is used for graphical management, then NTP must be configured on that host as well.

Configure NFS Client to Use Consistent Ports

By default, the portmap service dynamically assigns ports for remote procedure call (RPC) services that can be troublesome when interacting with a firewall. To gain consistent control over which NFS ports are used, it is necessary to configure the two primary NFS client services to use the same ports from connection to connection and host to host.

By default, Red Hat Enterprise Linux 6 mounts NFS storage with best practices. When configuring the NFS datastore to be mounted automatically across reboots, be sure to specify that the mount is a network device.

Register All Red Hat KVM Hosts to Red Hat Network

To make sure that the Red Hat KVM hosts have access to the latest security patches and bug fixes, it is important to register them to the Red Hat Network (RHN). This requires a valid Red Hat support subscription for each Red Hat KVM host.

3.5 KVM Host Security Guidelines

When setting up security, consider the following guidelines:

- Create strong passwords. See the Red Hat guide at https://access.redhat.com/kb/docs/DOC-9128.
- Allow access only to virtualization administrators. There should be nothing on the KVM host that regular users need to see; they should not be provided logins to the KVM host.
- Make a list of all running services and determine which ones do not need to run, then shut them down and disable them.
- Disable RSH, telnet, and FTP in favor of SSH, SCP, and SFTP or another secure FTP server.
- The host `/etc/fstab` file should not use disk labels to boot from.

  Note: When using a disk label as part of a cloning procedure, include a script to switch back to booting from a UUID when the cloned host comes back up.

- Register all Red Hat KVM hosts to RHN to provide access to the latest security patches and bug fixes.

Properly securing the host nodes is of paramount importance. The following subsections discuss multiple security considerations and best practices.

KVM Host Access Security

Good security begins with strong passwords. In addition to strong passwords, create a separate account for each administrator.

Caution

Do not use a single `root` account and password. Rather, grant each virtualization administrator `sudo` (`superuser do`) access.

Firewall and Mandatory Access Control

The firewall provided by iptables should allow only the ports needed to operate the virtual environment and to communicate with the NetApp FAS controller. The best practice is to leave iptables running and open ports as necessary rather than to disable it.

SELinux was developed largely by the National Security Agency (and incorporated into the 2.6 Linux kernel in 2003) to comply with U.S. government computer security policy enforcement. SELinux is built into the kernel and provides a mandatory access control (MAC) mechanism, which allows the administrator to define the permissions for how all processes interact with items such as files, devices, and processes. In RHEL 6, KVM is configured to work with SELinux by default. The best practice is to leave SELinux enabled.

Interhost Security

The primary means of connecting to a virtual server and the virtual shell (virsh) console is by way of SSH and an SSH tunnel, respectively. It is also possible to configure communication to the virsh console with the use of Transport Layer Security (TLS).

For secure communication between Red Hat KVM hosts, it is a best practice to set up and use SSH keys. When using a remote host to manage the KVM environment with graphical tools, the SSH keys can be added to that remote host as well.

Unnecessary and Unsecure Services

It is important to disable or uninstall any services that are not needed. For example, mail servers, Web servers, and databases have no place on a Red Hat host that is meant solely as a hypervisor. If there is a business need to stand up an application or service along with virtual machines, be careful to include only relevant packages and to make the proper configuration to the iptables firewall. The best practice for packages is that if it does not directly service or protect virtual machines, it should not be installed.

Disable RSH, telnet, FTP, and any other unsecure service in favor of SSH, SCP, and SFTP. RSH, telnet, FTP, and other “xinetd” related services are inherently unsecure.

Create a Red Hat KVM Host Template

Create a single Red Hat KVM host template that can be cloned for ease of creation. Although the initial Red Hat KVM build can be done using DVD or network, subsequent builds are much faster when cloned. After the build is complete, it should be made generic by removing configuration artifacts such as hostname, MAC addresses, and so on. These configurations can be configured in the postclone process and still be created much faster than repeated Kickstart deployments or DVD deployments.

Additionally, this makes for very efficient use of storage. Because new KVM hosts are created from the same template, the underlying storage can be deduplicated for significant reduction in space used.

3.6 Red Hat KVM Datastores and File Types

Red Hat KVM File Types

Red Hat KVM employs two primary types of files — a disk image and an Extensible Markup Language (XML) descriptor file. The disk image is the virtual block device that the virtual machine boots from and stores data on. The XML descriptor file provides all of the metadata for a virtual machine, including the virtual machine UUID, network MAC address, disk image location, and other critical information.

The disk image files (discussed in the next section) reside on the shared storage, but by default the XML descriptor files are saved in a directory that is local to the Red Hat KVM host. This needs to be accounted for in backup, disaster recovery, and site failover strategies. There are multiple ways of handling this, including creating a link between the default XML directory and the shared storage.

Raw Disk Image Files and QCOW2 Disk Image Files

Red Hat KVM supports the use of two different disk image formats: raw and qcow2. A raw disk image is a faster format that supports both thick and sparse allocation. Qcow2 has some advanced features such as VM-based Snapshot copies, but at the cost of performance. The best practice when deploying Red Hat KVM on NetApp storage is to use raw disk files and allow the NetApp controller to thin provision, deduplicate data, and provide Snapshot copies on the underlying storage. It can do so much more quickly and efficiently without involving hypervisor CPU cycles.

3.7 LUN-Based Datastores

Overview

Red Hat provides and supports multiple means of using LUN-based storage. The primary choices are:

- Red Hat Enterprise Linux standard file systems such as EXT3 and EXT4
- Red Hat Enterprise Linux clustered file system (GFS2)
- Red Hat Enterprise Linux Logical Volume Manager (LVM2)

The standard Red Hat Enterprise Linux file systems offer a stable and high-performing means of using a LUN-based datastore.

Note: With EXT3 and EXT4, any given virtual machine must be running on only one hypervisor at any given moment. That is, if a virtual machine is running on one Red Hat KVM host, it must not be started, even accidentally, on another Red Hat KVM host. This would cause severe corruption to the virtual machine. Allowing KVM to manage the datastore directly, as opposed to allowing RHEL to manage it, can mitigate this problem. Another option is to add a conditional test to all automation scripts that start or stop virtual machines.

Global File System (GFS2) is a clustered file system that allows multiple servers to read and write simultaneously to the same shared LUN. GFS2 does add a layer of complexity, because it also requires Red Hat Cluster Suite to be installed and configured.

Logical Volume Manager (LVM2) is typically used as a layer of abstraction between a block-based storage device (local disk, direct-attached, SAN, and so on) and the file system. However, LVM can also be used as a means of managing LUNs and individual virtual machines without using a file system between the logical volume and the virtual machine. Essentially, a LUN is configured as an LVM volume; group and virtual machines are created as individual logical volumes within the volume group. This is actually how LUNs are managed under Red Hat Enterprise Linux.

Virtual machines that are stored on a LUN-backed file system (EXT3, EXT4, and GFS2) are created as simple files that act as block devices. When using Logical Volume Manager without a file system, the logical volume is the block device for the virtual machine.

Spanning LUN Datastores

Whether LVM is used with or without a file system, it offers the benefit of growing dynamically. When additional storage is needed for the datastore, a new LUN is created on the NetApp controller, and it is added to the pool of storage in LVM on the Red Hat KVM host. If a file system is used, it is then extended. The combination of LVM on the Red Hat side and the flexible storage on the NetApp side means that traditional scaling obstacles can be quickly overcome.

NFS Datastores

Red Hat Enterprise Linux and Red Hat KVM allow customers to leverage enterprise-class NFS arrays to provide datastores with concurrent access by all of the nodes that use the shared storage. The NetApp NFS offers high performance, the lowest per-port storage cost, and some advanced data management capabilities.

NFS Datastores on NetApp

Deploying Red Hat KVM with the NetApp advanced NFS results in a high-performance; easy-to-manage implementation that provides VM-to-datastore ratios that cannot be accomplished with other storage protocols, such as Fibre Channel. This architecture can result in a tenfold increase in datastore density with a correlated reduction in the number of datastores. With NFS, the virtual infrastructure receives operational savings because there are fewer storage pools to provision, manage, back up, replicate, and so on.

Through NFS, customers receive an integration of Red Hat KVM virtualization technologies with NetApp WAFL® (Write Anywhere File Layout), the NetApp advanced data management and storage virtualization engine. This integration provides transparent access to VM-level storage virtualization offerings such as production-use data deduplication, immediate zero-cost VM and datastore clones, array-based thin provisioning, array-based virtual machine cloning, automated policy-based datastore resizing, and direct access to array-based Snapshot copies.

Virtual machines that are backed by NFS-based datastores are created as simple files that act as block devices.

Datastore Comparison Tables

Many points must be considered in differentiating what is available with each type of datastore and storage protocol. Table 1 compares the features available with each storage option.

Table 1) Datastore supported features.

Capability or Feature	FC or FCoE	iSCSI	NFS
Format	EXT3, EXT4, GFS2, LVM	EXT3, EXT4, GFS2, LVM	NetApp WAFL
Optimal queue depth per LUN or file system	64	64	N/A
Available link speeds	4 and 8GB FC and 10GbE	1 and 10GbE	1 and 10GbE

Table 2 compares storage-related functionality of Red Hat KVM features across different protocols.

Table 2) Red Hat-supported storage-related functionality.

Capacity or Feature	FC or FCoE	iSCSI	NFS
Live migration	Yes	Yes	Yes
NetApp cloned datastores	Yes	Yes	Yes
NetApp cloned virtual machines	No	No	Yes
Data deduplication	Yes	Yes	Yes
Resize datastore	Grow only	Grow only	Grow, autogrow, and shrink
Thin provision datastores	Yes	Yes	Yes
NetApp Snapshot copies	Yes	Yes	Yes
Restore datastores and VMs from SnapMirror® and SnapRestore®	Yes	Yes	Yes
Boot from SAN	Yes	Yes with HBAs	No

3.8 KVM Guest Configuration

Virtual Guest Limits

Red Hat KVM supports the configurations shown in Table 3.

Table 3) Red Hat KVM supported configurations.

Component	Units
Virtual CPUs	Maximum of 16 per guest
Virtual RAM	Maximum of 256GB per 64-bit guest
Virtual RAM	Maximum of 4GB per 32-bit guest

Component	Units
Virtual storage devices	Maximum of 8 per guest
Virtual network devices	Maximum of 8 per guest
Virtual PCI devices	Maximum of 32 per guest

Red Hat KVM Virtualized Guest Support

Red Hat KVM presently supports the following virtualized guest operating systems:

- Red Hat Enterprise Linux 3 (32-bit and 64-bit)
- Red Hat Enterprise Linux 4 (32-bit and 64-bit)
- Red Hat Enterprise Linux 5 (32-bit and 64-bit)
- Red Hat Enterprise Linux 6 (32-bit and 64-bit)
- Windows® XP Service Pack 3 and later (32-bit only)
- Windows 7 (32-bit and 64-bit)
- Windows Server® 2003 Service Pack 2 and later (32-bit and 64-bit)
- Windows Server 2008 (32-bit and 64-bit)
- Windows Server 2008 R2 (64-bit only)

Paravirtualized Drivers Support

The paravirtualized block and network drivers (the virtio drivers) support the following operating systems and versions. The paravirtualized drivers increase the performance of a guest's block and network devices:

- Windows XP
- Windows 7 (32-bit and 64-bit)
- Windows Server 2008 (32-bit and 64-bit)
- Windows Server 2003 R2 (32-bit and 64-bit)
- Red Hat Enterprise Linux 4.8 and newer (32-bit and 64-bit)
- Red Hat Enterprise Linux 5.4 and newer (32-bit and 64-bit)
- Red Hat Enterprise Linux 6.0 and newer (32-bit and 64-bit)

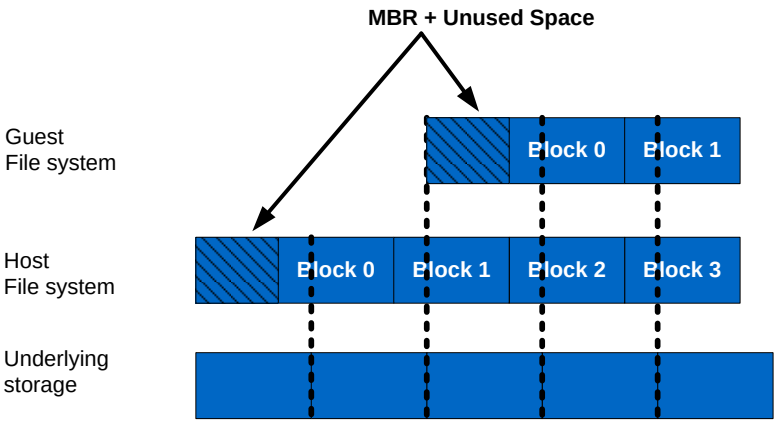
3.9 File System Alignment

In any virtual environment, a number of layers of abstraction exist between physical disks and the VM's virtual disk. Each layer in turn is organized into blocks to make the most efficient use of storage. The focus is not the size of the block, but rather the starting offset.

To avoid latency caused by additional reads and writes, the starting offset of a file system on a virtual machine should line up with the start of the block at the next layer down and continue that alignment all the way down to data blocks at the aggregate layer on the NetApp controller.

This is in no way unique to NetApp; it applies to any storage vendor. It is a simply a by-product of legacy partitioning schemes. For the full explanation of disk alignment in virtual environments, see TR-3747: Best Practices for File System Alignment in Virtual Environments.

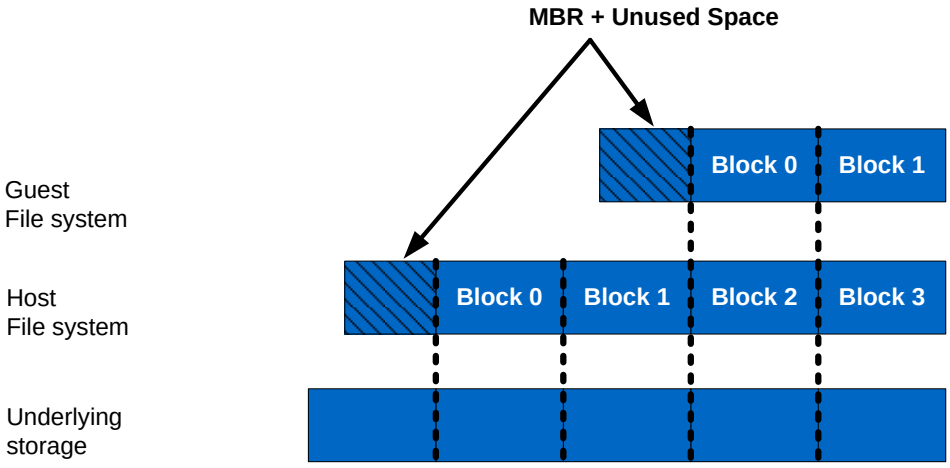
Figure 3) Misaligned file system.



Without correct alignment, significant latency occurs because the storage controller has to perform additional reads and writes for the misaligned blocks. For example, most modern operating systems, such as RHEL and Microsoft® Windows 2000 and 2003, use a starting offset of sector 63. Pushing the offset to sector 64 or sector 128 causes the blocks to align correctly with the layers below.

Microsoft Windows Server 2008, Windows 7, and RHEL 6 all align correctly by default and require no additional consideration. However, earlier versions of Microsoft Windows (Server 2003, XP, and so on) and RHEL (3, 4, and 5) all require additional steps at deployment to facilitate correct file system alignment.

Figure 4) Correctly aligned file system.



3.10 Thick and Thin Provisioning of KVM Guests

Red Hat KVM allows both thick and thin provisioning of guest virtual disks. The general recommendation from Red Hat is to thick provision production guests, and thinly provision desktops and dev/test guests to balance performance and storage capacity.

However, when coupled with NetApp, the underlying storage can be thin provisioned and deduplicated. This allows all Red Hat KVM guests to be thick provisioned but still maintain an efficient storage

environment. The best practice is to thick provision the KVM guests but thin provision and deduplicate the underlying storage.

Create a KVM Guest Template

Instead of creating a new virtual machine guest each time, it is a best practice to create it first with Kickstart and then clone subsequent instantiations.

The concept is almost identical to that of creating a template for Red Hat KVM hosts. The base image is created by using Kickstart, and then the image is made generic. When new guests are needed, NetApp FlexClone® thin-cloning technology is used to clone NFS-based guests. In the case of LUN-based guests, the KVM native `virt-clone` command is used. For Microsoft guests, `sysprep` is also used to make the guest sufficiently generic.

Kickstart

It is a best practice to use Kickstart to build a Red Hat KVM guest for the first time. Kickstart provides a semiautomated way to deploy Red Hat Enterprise Linux in a highly customizable way. After a Red Hat KVM guest is created and made generic, it is a best practice to repeat the deployment by using NetApp FlexClone.

NetApp FlexClone

FlexClone thin-cloning technology is highly efficient in many use cases, and in the case of virtualization it can be used to rapidly clone virtual machines (NFS-based) and hypervisors (iSCSI, FC, and FCoE). Offloading virtual machine cloning from the hypervisor to the NetApp controller is an efficient means of scaling out the environment without taxing CPU cycles on the Red Hat KVM hypervisor. Additionally, FlexClone works with data deduplication and thin provisioning to make a smaller storage footprint.

Cloning LUN-based virtual machines requires the use of the Red Hat KVM native tool.

Guest Timing Issues

All KVM (Windows and RHEL) guests must be configured to use NTP to avoid issues that arise from time skew.

Security Considerations

Similar to the Red Hat host security, the focus is on firewall, mandatory access control, unnecessary services, and unsecure services.

RHEL guests should have iptables running and SELinux enabled. It is a best practice to configure the necessary ports or controls rather than to disable that layer of security. Limit the packages installed to those that are necessary. Do not use RSH, telnet, and FTP; instead, use SSH, SCP, and SFTP.

Most Windows guests have some form of firewall as well. It is a best practice to open a necessary port, rather than to disable that layer of security. Additionally, the running services should be reviewed and any unnecessary services should be shut down. Antivirus software should also be used.

4 NetApp Storage Best Practices for Red Hat KVM

4.1 Storage Array Thin Provisioning

Server administrators often overprovision storage to avoid running out of storage and to prevent the associated application downtime when expanding the provisioned storage. Although no system can be run at 100% storage use, methods of storage virtualization allow administrators to address and oversubscribe storage in the same manner as with server resources (such as CPU, memory, and networking). This form of storage virtualization is referred to as thin provisioning.

Traditional provisioning preallocates storage; thin provisioning provides storage on demand. The value of thin-provisioned storage is that storage is treated as a shared resource pool and is consumed only as each individual VM requires it. This sharing increases the total usage rate of storage by eliminating the provisioned but unused areas of storage that are associated with traditional storage. The drawback to thin provisioning and oversubscribing storage is that (without the addition of physical storage) if every VM requires its maximum possible storage at the same time, not enough storage will be available to satisfy the requests.

4.2 NetApp Thin-Provisioning Options

NetApp thin provisioning can be used along with RHEL KVM thick- or thin-provisioned virtual machines. Red Hat's own best practice is to use thin provisioning for desktops and dev/test environments. Thick provisioning should be used for server workloads. However, this practice does not take into account an enterprise storage array such as NetApp. Enabling thin provisioning (at all layers) on the NetApp controller and thick provisioning the virtual machines in KVM provides proper balance between performance and storage efficiency.

When enabling NetApp thin-provisioned LUNs, NetApp recommends deploying these LUNs in FlexVol® volumes that are also thin provisioned with a capacity that is twice the size of the LUN. By deploying the LUN in this manner, the FlexVol volume acts merely as a quota. The storage consumed by the LUN is reported in the FlexVol volume and its containing aggregate.

Storage Array Deduplication

Similar to the other modern virtualization platforms, Red Hat KVM allows the creation of new virtual machines by cloning a template. The template is essentially a virtual machine that was installed, configured, made generic, and then shut down. The Red Hat cloning process then creates a new virtual machine (and configuration file) based on the template and its configuration file.

NetApp offers a data deduplication technology called *fabric-attached storage (FAS)* data deduplication. Deduplication virtualization technology enables multiple VMs to share the same physical blocks in a NetApp FAS system in the same manner that VMs share system memory, resulting in significant storage savings. Deduplication can be seamlessly introduced into a virtual data center without having to make any changes to the way the Red Hat KVM is maintained or administered. It runs on the NetApp FAS system at scheduled intervals and does not consume any CPU cycles on the hypervisor.

Deduplication is enabled on a volume, and the amount of data deduplication realized is based on the commonality of the data stored in a deduplication-enabled volume. For the largest storage savings, NetApp recommends grouping similar OSs and similar applications into datastores, which ultimately reside on a deduplication-enabled volume.

Deduplication Considerations with LUNs

When enabling deduplication on thin-provisioned LUNs for Red Hat KVM, NetApp recommends deploying these LUNs in FlexVol volumes that are also thin provisioned with a capacity that is twice the size of the LUN. By deploying the LUN in this manner, the FlexVol volume acts merely as a quota. The storage consumed by the LUN is reported in the FlexVol volume and its containing aggregate.

Deduplication Advantages with NFS

Unlike LUNs, when deduplication is enabled with NFS, the storage savings are immediately available and recognized by the virtualization team. The benefit of deduplication is transparent to storage and virtualization admin teams. No special considerations are required for its use with Red Hat KVM.

5 Storage Network Best Practices for Red Hat KVM

5.1 Storage Architecture

Before configuring the storage array to run a virtual infrastructure, take the following steps:

- Separate networks for storage array management and storage I/O. This applies to all storage protocols, but it is especially pertinent to Ethernet-based deployments (NFS, iSCSI, and FCoE). The separation can be physical (subnets) or logical (VLANs), but it must exist.
- When leveraging an IP-based storage protocol I/O (NFS or iSCSI), multiple IP addresses may be required for storage targets. The determination is based on the capabilities of the existing networking hardware.
- With IP-based storage protocols (NFS and iSCSI), multiple Ethernet ports are channeled together. NetApp refers to this function as a virtual interface (VIF). NetApp recommends creating Link Aggregate Control Protocol (LACP) VIFs rather than multimode VIFs whenever possible.

Note: Cisco refers to this as an Etherchannel; NetApp refers to it as a VIF (Data ONTAP 7) or an interface group, or ifgrp (Data ONTAP 8).

Production Ethernet Storage Networks

The goal of any storage network is to provide uninterrupted service to all nodes that connect to it. This section focuses primarily on how to establish a highly available Ethernet storage network. There are two reasons for focusing on Ethernet. First, Fibre Channel storage networks provide a single service, Fibre Channel. These single-purpose networks are simpler to design and deploy in a highly available configuration. Second, the current industry trend focuses solely on multipurpose Ethernet networks (converged networks) that provide storage, voice, and user access.

Regardless of protocol, a storage network must address the following three goals:

- Be redundant across switches in a multiswitch environment
- Use as many available physical paths as possible
- Be scalable across multiple physical interfaces or ports

10GBE or Data Center Ethernet

NetApp Data ONTAP, Red Hat Enterprise Linux, and Red Hat KVM all support 10GbE. An advantage of 10GbE is the ability to reduce the number of network ports in the infrastructure, especially but not limited to blade servers. Additionally, 10GbE can handle several VLANs simultaneously. It is a NetApp best practice to use 10GbE, especially for storage.

VLAN Tagging or 802.1Q

When segmenting network traffic with VLANs, interfaces can either be dedicated to a single VLAN or support multiple VLANs with VLAN tagging.

For systems that have fewer NICs, such as blade servers, VLANs can be very useful. Bonding two NICs together provides a Red Hat KVM server with physical link redundancy. By adding multiple VLANs, it is possible to group common IP traffic onto separate VLANs for optimal performance. NetApp recommends putting virtual machine management traffic, virtual machine network traffic, and IP storage traffic on separate VLANs.

VLANs and VLAN tagging also play a simple but important role in securing an IP storage network. NFS exports can be restricted to a range of IP addresses that are available only on the IP storage VLAN. NetApp storage appliances also allow the iSCSI protocol to be restricted to specific interfaces and/or VLAN tags. These simple configuration settings have an enormous effect on the security and availability of IP-based datastores. If multiple VLANs are being used over the same interface, make sure that sufficient throughput can be provided for all traffic.

Routing and IP Storage Networks

Whenever possible, NetApp recommends configuring storage networks as a single network that is not routed. This method provides good performance and a layer of data security. In the context of Red Hat KVM, this means that logical networks should be created on 10.0.0.0/16 or 192.168.0.0/24 networks.

Separate Ethernet Storage Network

As a best practice, NetApp recommends separating IP-based storage traffic from public IP network traffic by implementing separate physical network segments or VLAN segments. This design follows the architecture of SCSI and FC connectivity. With this design, NetApp does not recommend the routing of data between the storage network and other networks. In other words, do not define a default gateway for the storage network. In the context of Red Hat KVM, this means that a separate logical network should be created for each storage network with VLAN tagging where possible.

NetApp Virtual Interfaces

A virtual interface (VIF, Data ONTAP 7) or interface group (ifgrp, Data ONTAP 8) is a mechanism that supports aggregation of network interfaces into one logical interface unit. This is the same concept as a channel bond in Red Hat Enterprise Linux. Once created, a VIF is indistinguishable from a physical network interface. VIFs are used to provide fault tolerance of the network connection and in some cases higher throughput to the storage device.

Although NetApp supports the use and configuration of several types of interface groups, the best practice is to configure and use the LACP VIF or ifgrp. An LACP ifgrp is a dynamic IEEE 802.3ad-compliant device that uses all physical connections simultaneously for traffic and also provides link status.

The use of ifgrps on NetApp aligns with the use of channel bonds on RHEV, eliminating single points of failure throughout the network.

5.2 Jumbo Frames

It is a NetApp best practice to use jumbo frames for Ethernet storage traffic, especially NFS. Standard frames require NFS datagrams to be broken up and reassembled at the destination. This causes a performance hit on both ends of the wire.

In contrast, a jumbo frame allows NFS datagrams to be sent whole, removing the need to break them up and reassemble them.

Jumbo frames should be configured for the following types of data streams:

- Storage traffic and isolated, nonrouted VLAN for NFS, CIFS, and iSCSI data
- Replication network and isolated, nonrouted VLAN for high-speed storage replication such as SnapMirror data

In the context of RHEV, each virtual bridge that is created on the hypervisor should be created with jumbo frames. Any virtual interface that uses the bridge recognizes that the bridge uses jumbo frames and automatically follows suit.

5.3 Multipathing

The use of multipathing is a best practice regardless of the storage protocol or network topology. Each interface (FC, FCoE, or Ethernet) should have a redundant counterpart that has a separate path to the storage target.

In the Red Hat lexicon, a channel bond is used to create a single logical Ethernet interface from two or more physical interfaces. Cisco refers to this as an Etherchannel; NetApp refers to it as a VIF (Data ONTAP 7) or an interface group, or ifgrp (Data ONTAP 8).

Red Hat also has a native multipathing driver called Device Mapper Multipath I/O (DM-MPIO) that is used to manage the multiple paths to a single target for both performance and stability. It can be used with both FC and Ethernet (1GbE and 10GbE) storage targets.

The best practice is to use multiple Ethernet cards and multiple FC HBAs on both the NetApp storage controller and on the Red Hat KVM host. Each NIC or HBA on the Red Hat KVM host needs to have its own separate path to the storage, thereby eliminating any single point of failure along the path. If the storage is over Ethernet, this requires the use of DM-MPIO as well as channel bonding.

6 Management Best Practices

The management servers described in this section can all be virtualized on a separate group of infrastructure hosts. When the management servers are virtualized, they gain the same benefits as the production virtual machines, such as mobility, availability, and centralized data management on the NetApp controllers.

Libvirt

libvirt is not a server, but a virtualization API and toolkit written in C. Although it is the primary API for KVM, it also supports Xen, OpenVZ, VMware® ESX and GSX, Microsoft Hyper-V™, and several others. Scripts and tools that are used to manage KVM should make API calls to libvirt.

Virtual Machine Manager

The Virtual Machine Manager (VMM) is a graphical interface to the libvirt virtualization API library. As such, it requires a graphical environment to operate. As mentioned earlier in this guide, the best practice is to not install graphical packages on a production Red Hat KVM host. If graphical tools are required, it is better to set up a separate remote administration host to manage the Red Hat KVM hosts and guests.

Use of a Remote Administration Host

If graphical tools are required, NetApp recommends using a remote host to administer the Red Hat KVM environment. The remote host is used to run the entire virtual environment from a central server or workstation instead of on one or more of the host nodes. The only requirement is to install the basic KVM packages needed to run the various administrative commands on the remote host as well as on a GUI desktop such as Gnome or KDE.

Note: All of the security best practices (iptables, SELinux, SSH keys, and so on) for a Red Hat KVM host apply to a remote administration host as well.

A remote administration host has the following uses in a KVM environment:

- Secure host to manage the KVM environment
- Secure host to manage the NetApp FAS controller
- Secure host to manage Red Hat Cluster Suite (if using GFS or GFS2)
- Secure host to run the NetApp Snap Creator™ server (backup framework)

RHN and RHN Satellite

Red Hat Network (RHN) is a secure Web portal as well as the source for all Red Hat related packages, updates, errata, and management tools for RHEL servers. It is a Red Hat and NetApp best practice to subscribe all Red Hat systems to RHN in order to keep up with security patches as well as compliance.

RHN Satellite is essentially an on-site instantiation of RHN. Instead of many systems subscribed to RHN, only the Satellite server is subscribed. All Red Hat systems are then subscribed to the Satellite server. This has many added benefits such as reduced external network traffic and the ability to highly customize software channels and user management. Additionally, RHN Satellite can be used to provision new RHEL servers. RHN Satellite can be deployed on a RHEL KVM guest.

NetApp Operations Manager

NetApp Operations Manager provides a centralized management portal for multiple NetApp storage controllers. It also provides monitoring, alerts, reporting, and configuration tools. NetApp Operations Manager can be deployed as a RHEL or Windows KVM guest.

Kickstart Server

Kickstart is a means of providing semiautomated RHEL installations and can be deployed using CD and answer file, HTTP, NFS, or FTP. It is a best practice to deploy Kickstart as a server on a RHEL KVM guest or as part of a RHN Satellite server.

Scaling Out the Environment

Scalable environments are typically described as *scaling up* (ease of upgrading to more powerful systems) and *scaling out* (ease of growth).

Proper deployment of the NetApp FAS controllers allows nondisruptive upgrades to bigger and faster controllers for scaling up. For instance, deploying a pair of NetApp controllers as an active-active pair allows the workloads of both controllers to be handled by one controller while the other is being replaced or upgraded.

Ease of scaling out is generally more important. Typically, as virtual environments grow, it becomes necessary to balance workloads between controllers and to react quickly to sudden increases in activity. This adds the requirement that both hypervisors and virtual machines can be created on demand. For this reason, NetApp highly recommends using NetApp MultiStore® and FlexClone in a Red Hat KVM environment.

MultiStore provides an additional layer of storage virtualization by way of multiple lightweight instances of Data ONTAP that are created in NetApp vFiler® units. Each vFiler unit maintains its own network routing tables, IP storage, and authentication. In the context of scaling out, vFiler units can be migrated dynamically from one NetApp controller to another.

NetApp FlexClone is used to create writable Snapshot copies of FlexVol volumes, LUNs, and in the case of NFS-based storage, individual virtual machines. This means that as additional KVM hosts are needed, they can be cloned from an existing LUN. And as additional virtual machines are needed, they too can be cloned rapidly.

NFS-based virtual machines should be rapidly deployed by cloning templates. The XML descriptor file can then be cloned quickly by using the native Red Hat KVM cloning tool. LUN-based virtual machines can only be cloned by using the native Red Hat KVM cloning tool.

7 Backup and Disaster Recovery Best Practices for Red Hat KVM

NetApp's approach to backup and disaster recovery is very different from the traditional approach. Where traditional backups make heavy use of tape media, NetApp's approach begins with a Snapshot copy. This Snapshot copy can then be offloaded to a lower tier of disk on the same controller or mirrored to a different controller at a different site. This ultimately means that it takes much less space, does not require additional management, and in the case of disaster (big or small) the data can be quickly restored.

NetApp Snapshot Copies

In the context of a NetApp volume, a Snapshot copy is a point-in-time, read-only copy that is similar to the way the volume appears. Creating a Snapshot copy is nearly instantaneous, with most copies complete in less than a second. After a Snapshot copy is created, it can be used to recover data lost by human error or application error. By default, when a flexible volume is created, a reserve of 20% is maintained.

In general, the best practice for Snapshot reserve in a virtual environment is to set it to 0% and to disable the Snapshot copy schedule. To guarantee file consistency, it is best to quiesce (pause) the virtual guests, perform the Snapshot copy, and then resume the virtual guests. (Information about Snap Creator appears later in this section.)

Here are some key points about Snapshot technology in the context of a KVM environment:

- Snapshot technology provides a natively integrated and easy-to-use data protection utility that helps storage administrators recover data.
- Snapshot copies protect against inadvertent file modification or deletion by making point-in-time backups available.

When creating Snapshot copies of datastores, remember that a Snapshot copy itself is not a backup. The copy can be used as a backup or for disaster recovery, but this means that the copy must be replicated elsewhere. With that in mind, there are also some considerations about the level of consistency of the copy; these considerations are discussed later in this section.

When using Snapshot copies as part of a backup strategy for Red Hat KVM, multiple file and datastores must be accounted for as part of the process. A single VM is composed of two files, a disk image file, and an XML descriptor file. Any Snapshot copy that is created against the datastore captures the disk file. However, the XML descriptor file is not kept on the shared storage by default. The directory that contains the XML descriptor files can be linked to the same shared storage as the virtual machines.

Any datastores that hold application data are an additional consideration.

It is a best practice to use Snap Creator and SnapMirror in conjunction with Snapshot copies to offload the copies for disaster recovery, backup, and/or site failover.

NetApp Snap Creator

Snap Creator is an extensible backup framework that works with many applications and platforms, such as Red Hat KVM. The framework allows triggering NetApp Snapshot copies of volumes (datastores) and also any activities that need to occur before and after the Snapshot copy is taken. It can also trigger SnapMirror activities between NetApp controllers and/or data centers to meet disaster recovery and backup requirements.

There are numerous backup use cases; however, the following three use cases represent the vast majority that need to be addressed in a virtual environment.

Crash-Consistent Backups with Snap Creator

This use case takes a Snapshot copy of a datastore without quiescing any virtual machines or applications; that is, while everything is in flight. The Snapshot copy can then be mirrored to another controller for backup or archiving. This is fine for capturing current state, but a restore would depend on file system logs on the guest operating system to replay properly.

Application-Consistent Backup with Snap Creator

This use case assumes that the application data is on a separate volume from the virtual machine datastore. Snap Creator first triggers the application to quiesce, then triggers the Snapshot copy on the application data volume, then triggers the application to resume. The Snapshot copy can then be mirrored to another controller for backup or archiving.

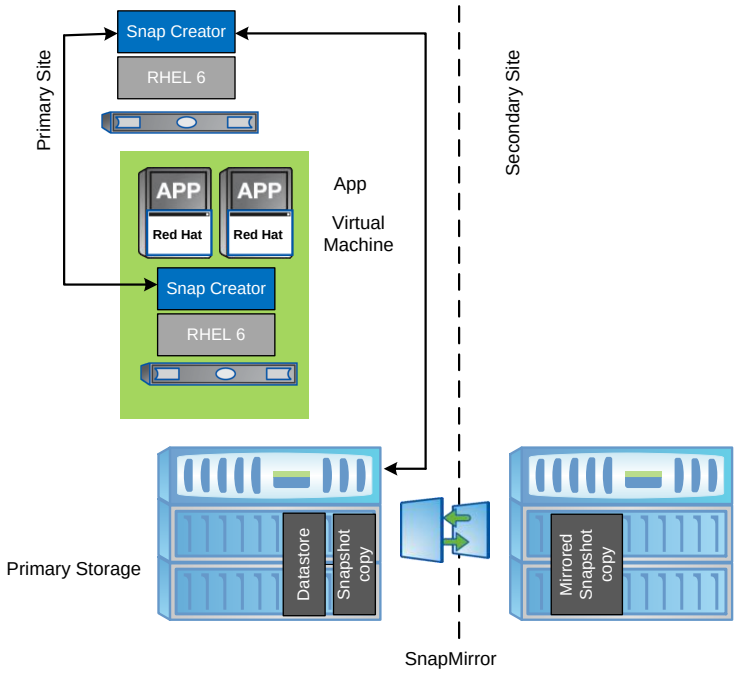
Fully Consistent Snapshot Backup with Snap Creator

This use case is typically an add-on to the application-consistent backup. After the application is quiesced, Snap Creator tells the guest to sync its buffers (RHEL only), then tells Red Hat KVM to pause the virtual machine, triggers the Snapshot copies for the application volume and the virtual machine datastore, tells Red Hat KVM to resume the virtual machine, and finally resumes the application. The speed of the Snapshot copy as well as the speed of the virtual machine pause/resume means that this activity can occur very quickly (less than 5 seconds). The Snapshot copies can then be mirrored to another controller for backup or archiving.

Snap Creator Components

A Snap Creator server should be deployed on a separate server, or virtual machine, outside of the Red Hat KVM environment that it is backing up. RHEL is supported as a Snap Creator server platform. A Snap Creator client is then installed on Red Hat KVM hypervisors and/or guests. Applications that require quiescing must have the Snap Creator agent installed on the virtual machine. The Snap Creator agent is supported on both RHEL and Windows and can be used to back up the data for many different application workloads.

Figure 5) Example Snap Creator server and agent layout.



NetApp SnapMirror

Snap Creator is a critical piece of the backup strategy, but it is not the only piece. A Snapshot copy can be used to restore an entire datastore or an individual virtual machine, but it does not protect the Snapshot copy of the RHEV datastore on its own; this is where SnapMirror is added to the RHEV backup strategy.

NetApp SnapMirror is the data replication software that improves data protection by maintaining duplicate copies of data between NetApp controllers. After the initial baseline data transfer, SnapMirror replicates only changed blocks from the primary storage controller to minimize performance impact on storage and bandwidth impact on the network. Additionally, SnapMirror honors deduplicated blocks.

As a critical piece of disaster recovery planning and implementation, the best practice is to deploy SnapMirror in addition to Snap Creator for RHEV datastore replication. It can also be deployed for site failover. It is important to stagger the transfers for non-peak-load times. Finally, data can also be backed up from a SnapMirror partner to tape, VTL, or other archive device.

NetApp MetroCluster

NetApp MetroCluster™ is the premier site failover solution from NetApp. It synchronously mirrors data at the aggregate layer between sites that are up to 100km apart. As changes to an RHEV virtual machine are written at one site, they are simultaneously written at the other site. Similar to an active-active pair, one controller can handle storage requests for both. However, if an entire site becomes unavailable, the mirrored data can be accessed immediately as one controller takes on the identity of both controllers simultaneously.

Where business needs dictate the highest level of uptime and continuity for RHEV, the NetApp best practice is to use MetroCluster as the storage foundation.

7.1 Site Failover Considerations for Red Hat KVM

When planning site failover strategy for Red Hat KVM, it is important to create the same IP networks and VLANs at both sites. This allows the hypervisors and virtual machines to operate equally well at either site. If the IP space is actually shared between sites, then both sites can be active simultaneously as well.

Traditional Backup Methods

Although the use of Snapshot copies, Snap Creator, and SnapMirror are best practices; they are not requirements for Red Hat Enterprise Virtualization. However, NetApp recommends using some form of data backup as a key foundation piece of enterprise data protection.

NetApp also provides two means of data backup that are included in Data ONTAP and that do not require any additional licenses. The `dump` and `ndmpcopy` tools are available to replicate data to tape drives or to other storage, respectively. This satisfies the requirement for backup utilities in Red Hat virtualization.

8 Conclusion

Red Hat's implementation of KVM offers a highly configurable and high-performance virtual environment that is easy to deploy. This makes it a primary candidate for IT infrastructures that already have their own tools, a foundation of Linux or Linux skills, and the need for a solid virtualization platform that plugs in to an existing environment.

A simple KVM environment can be set up and tested in a matter of minutes. A more complex production KVM infrastructure can be planned and deployed in a few short weeks. The graphical tools enable newcomers to quickly grasp the concepts, and the command line tools are easily integrated into automation, management, and monitoring applications and tools.

From a storage and data efficiency standpoint, NetApp FAS controllers offer a unified, flexible approach to storage. The ability to deliver NFS, iSCSI, and FCP to multiple KVM environments simultaneously means that the storage scales nondisruptively with the KVM environment. Multiple KVM environments with different storage needs can be supported from the same NetApp FAS controller.

Additional NetApp products and technologies such as Snapshot, SnapMirror, and deduplication offer the protection and storage efficiency required in any infrastructure.

The best practices in this guide describe a virtual infrastructure based on KVM and NetApp that serves as a solid foundation for many applications.

References

For additional information, visit:

- KVM Home Page
  www.linux-kvm.org
- Red Hat and Microsoft Virtualization Interoperability
  www.redhat.com/promo/svvp/
- KVM – Kernel-Based Virtual Machine
  www.redhat.com/f/pdf/rhev/DOC-KVM.pdf
- Red Hat Enterprise Linux 6 Virtualization Guide

  docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Virtualization_Administration_Guide/index.html
- Red Hat Enterprise Linux 6 Deployment Guide
  docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html

- Red Hat Enterprise Linux 6 Installation Guide
  docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/index.html
- TR-3747: Best Practices for File System Alignment in Virtual Environments
  media.netapp.com/documents/tr-3747.pdf
- TR-3427: Storage Best Practices and Resiliency Guide
  media.netapp.com/documents/tr-3437.pdf
- TR-3505: NetApp Deduplication for FAS and V-Series Deployment and Implementation Guide
  media.netapp.com/documents/tr-3505.pdf
- TR-3446: SnapMirror Async Overview and Best Practices Guide
  media.netapp.com/documents/tr-3446.pdf

www.netapp.com