



Technical Report

# Red Hat Enterprise Virtualization 3 and NetApp Storage: Best Practices Guide

Jon Benedict, NetApp  
March 2012 | TR-3914

## TABLE OF CONTENTS

<b>1</b>	<b>Red Hat Enterprise Virtualization on NetApp Storage</b>	<b>4</b>
1.1	Overview	4
1.2	The 80/20 Rule	5
1.3	Brief History of the Kernel-Based Virtual Machine	5
1.4	Major Components RHEV 3.0	6
1.5	Thick and Thin Hypervisors	7
<b>2</b>	<b>RHEL 6, KVM, and RHEV-M Installation</b>	<b>7</b>
2.1	RHEV Hardware Requirements	8
2.2	RHEV Software Requirements	9
2.3	Storage Requirements for RHEV and Virtual Machines	9
2.4	Network Layout for Red Hat KVM and NetApp Storage	10
<b>3</b>	<b>Hypervisor Installation</b>	<b>10</b>
3.1	RHEV-H and Use Cases	10
3.2	RHEL 6 and Use Cases	11
3.3	RHEV-H Security Guidelines	11
<b>4</b>	<b>Red Hat Enterprise Virtualization Datastores</b>	<b>11</b>
4.1	Raw Disk Image Files and QCOW2 Disk Image Files	11
4.2	LUN-Based Datastores	11
4.3	NFS Datastores	12
4.4	Datastore Comparison Tables	12
<b>5</b>	<b>RHEV Guest Configuration</b>	<b>13</b>
5.1	File System Alignment Overview	13
5.2	Create a RHEV Guest Template	13
5.3	Kickstart	14
5.4	FlexClone	14
5.5	Guest Timing Issues	14
5.6	Security Considerations	14
5.7	RHEV Virtualized Guest Support	14
5.8	ParaVirtualized Drivers Support	15
<b>6</b>	<b>NetApp Storage Best Practices for RHEV</b>	<b>15</b>
6.1	Storage Array Thin Provisioning	15
6.2	Storage Array Deduplication	16
<b>7</b>	<b>Storage Network Best Practices for RHEV</b>	<b>16</b>

7.1 Storage Architecture Concepts .....	16
7.2 Production Ethernet Storage Networks .....	17
<b>8 Management Best Practices for RHEV and NetApp.....</b>	<b>19</b>
8.1 RHEV-M .....	19
8.2 REST API .....	19
8.3 RHN and RHN Satellite .....	19
8.4 NetApp Operations Manager .....	19
8.5 Kickstart Server .....	19
8.6 Scaling Out the Environment.....	19
<b>9 Backup and Disaster Recovery Best Practices for RHEV .....</b>	<b>20</b>
9.1 NetApp Snapshot.....	20
9.2 NetApp Snap Creator .....	20
9.3 SnapMirror .....	21
9.4 MetroCluster .....	22
9.5 Traditional Backup Methods.....	23
<b>10 Conclusion .....</b>	<b>23</b>
<b>References.....</b>	<b>23</b>

**LIST OF TABLES**

Table 1) Supported disk types. ....	9
Table 2) Datastore-supported features.....	12
Table 3) Red Hat–supported storage-related functionality. ....	12
Table 4) RHEV supported virtual hardware configurations. ....	14

**LIST OF FIGURES**

Figure 1) RHEV architecture.....	6
Figure 2) Thick and thin KVM hypervisors.....	7
Figure 3) Deployment example.....	8
Figure 4) Network layout best practices. ....	10
Figure 5) Site failover.....	22

# 1 Red Hat Enterprise Virtualization on NetApp Storage

## 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 provide industry-leading technologies in the areas of storage efficiency, instantaneous VM and datastore cloning for virtual servers, virtual data center backup, and business continuance solutions.

Red Hat, while a relative newcomer to the virtualization market, provides support for the high-performance and open-source Kernel-Based Virtual Machine (KVM) hypervisor. From a storage perspective, Red Hat Enterprise Virtualization (RHEV) supports both SAN (iSCSI, FC, FCoE) and NAS (NFS) for shared virtual machine storage. Shared storage is required to enable RHEL KVM features such as high availability, live migration, and fault tolerance.

NetApp and Red Hat have maintained a long-term strategic alliance that includes end-to-end solution testing of Red Hat products and NetApp storage. As a result of this testing, NetApp has developed operational guidelines for storage arrays running Data ONTAP® software and Red Hat Enterprise Linux®. These guidelines have been extended to include best practices for RHEV.

The following sections review the best practices to use for NetApp storage and RHEV when deployed together.

### Why Deploy RHEV with NetApp Storage?

The virtualization of a data center results in physical systems being virtualized as part of a cost-saving effort to reduce both capex and opex 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 or database servers results in increased operational efficiency. This latter group of systems might share server resources, but it is typically configured with exclusive access to the required storage.

Red Hat and NetApp offer technologies that natively support multiple storage protocols. These technologies allow customers to deploy best-in-class virtual data centers that leverage the inherent strengths when using these technologies together. This is not a SAN versus 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 simply 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 Enterprise Virtualization.

## 1.2 The 80/20 Rule

In designing the storage architecture for a virtual data center, one can apply what is referred to in this document as the 80/20 rule, which is that 80% of all systems virtualized are for consolidation efforts. The remaining 20% of the systems are classified as business-critical applications, and although they can be virtualized successfully, they tend to be deployed on shared storage pools. We refer to these pools 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 will suffice
- Typically contain the largest number of virtual machines
- The virtual machines within do not typically have large network or storage I/O requirements, but collectively might represent a resolvable challenge
- Consolidated datasets are ideally served by large, shared policy-driven storage pools (or datastores)

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

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

Consolidated datasets work well with NFS datastores because this design provides greater flexibility in terms of capacity and flexibility than SAN datastores when managing hundreds or thousands of VMs. Isolated datasets run well on all storage protocols; however, some tools or applications might have restrictions around 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 enable systems to become virtualized faster and easier than with a traditional storage array platform or multiple disparate storage arrays.

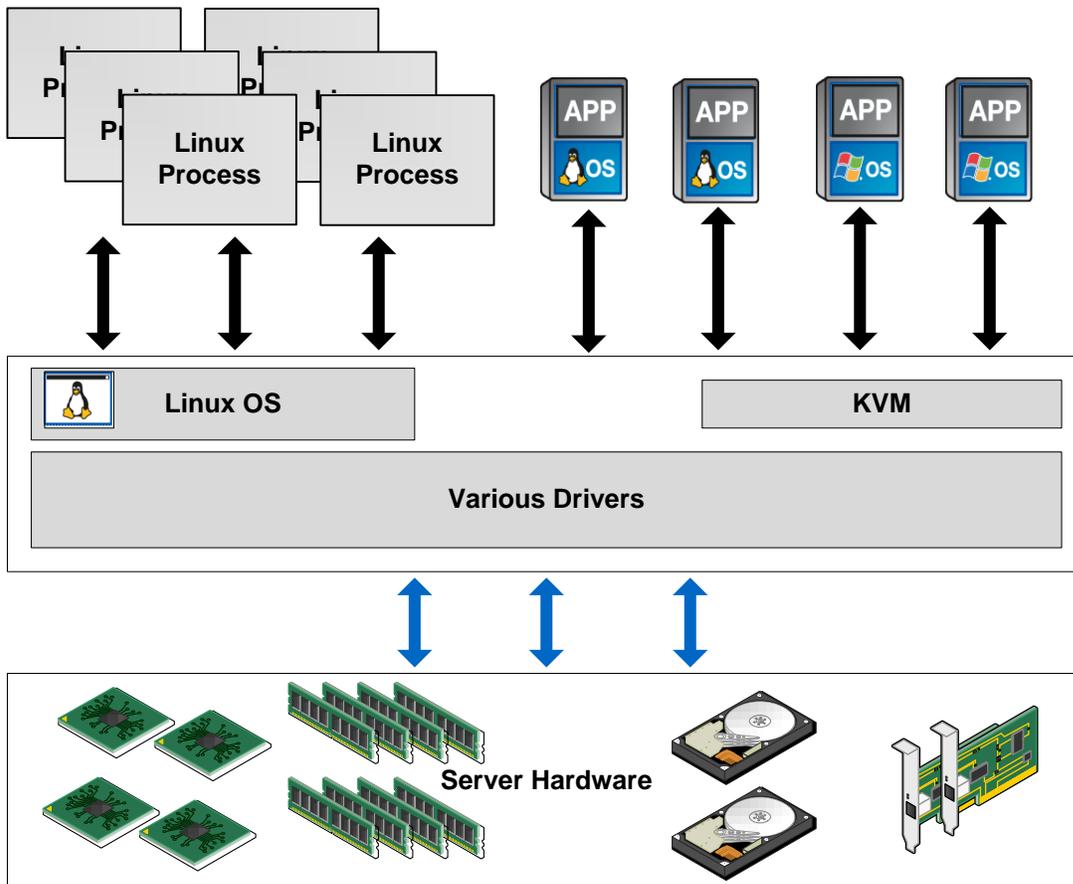
## 1.3 Brief History of the Kernel-Based Virtual Machine

Qumranet, an Israeli startup, launched the KVM hypervisor in the fall of 2006. At that time, most hypervisors existed as an additional layer on top of the operating system. The hypervisor provided virtualization capability, but it 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 RHEV. Although Red Hat is the largest contributor to KVM, the underlying KVM hypervisor for RHEV is still open source.

Figure 1) RHEV architecture.



## 1.4 Major Components RHEV 3.0

Red Hat Enterprise Virtualization (RHEV) includes two major components:

- A management portal, RHEV Manager (RHEV-M)—A Web application built on JBoss and an embedded PostgreSQL database
- A thin hypervisor, RHEV Hypervisor (RHEV-H)—A small-footprint appliance-like hypervisor

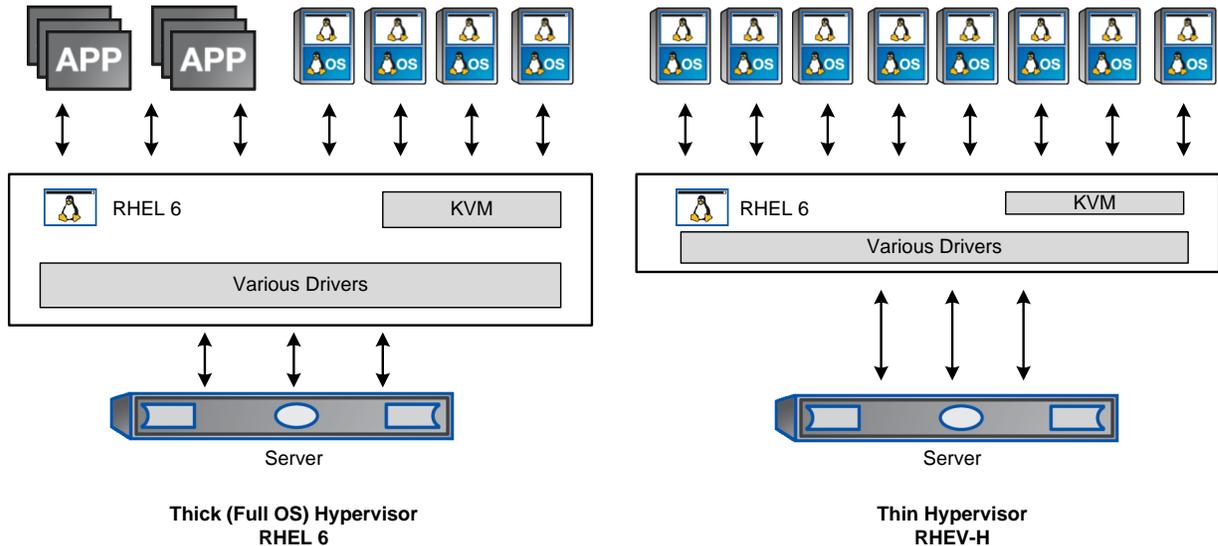
RHEV also supports the use and management of RHEL as a thick hypervisor.

While the ABI remains identical between RHEL 6 and RHEV-H, RHEV-H is devoid of compilers, interpreters, and anything else that does not directly support or protect a VM. RHEV-H carries all the hardware and software certifications that RHEL maintains, with the exception of the CPU requirements listed in section 2.1, “RHEV Hardware Requirements .”

## 1.5 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 KVM hypervisors.



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 also differences in support subscriptions. As mentioned earlier, the KVM hypervisor is available in non-Red Hat distributions. Because of these differences, it is incorrect to use the terms “KVM” and “RHEV” interchangeably.

This document focuses on Red Hat Enterprise Virtualization and how it uses Red Hat KVM as a “control plane.”

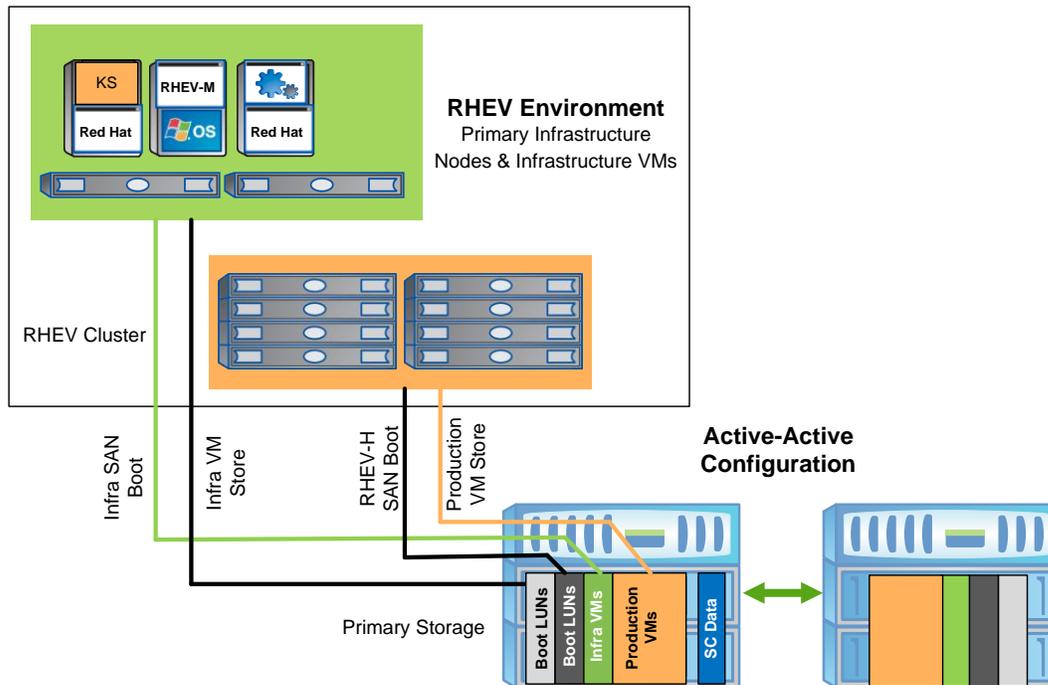
## 2 RHEL 6, KVM, and RHEV-M Installation

Although this document focuses mainly on Red Hat Enterprise Virtualization, the best practice is to deploy RHEV-M as a virtual instance, along with other infrastructure-level services. This ultimately means that Red Hat KVM is used in this infrastructure level or control plane. Follow the guidelines in [TR-3848: "Best Practices for KVM, Red Hat Enterprise Linux 6, and NetApp Storage"](#) to deploy at least two Red Hat KVM hosts that will host virtualized infrastructure services such as RHEV-M, Kickstart, RHN Satellite Server, and NetApp Operations Manager.

In Figure 3, the two servers in the green block represent the Red Hat KVM hosts that host the virtualized infrastructure services. The red block represents the production hosts made up of RHEV-H nodes and/or other Red Hat KVM hosts that are managed by RHEV-M.

All servers are SAN-booted from the NetApp controller. The NetApp controllers are deployed in an active-active configuration.

Figure 3) Deployment example.



## 2.1 RHEV Hardware Requirements

RHEV-H has 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 128 CPUs)
- 1 network interface card (NIC) with bandwidth of at least 1Gbps (dual 10GbE is preferred)
- 10GB RAM plus at least 1GB for each virtual machine (maximum of 1TB RAM)
- 10GB storage plus swap

RHEV-H swap requirements are as follows:

Multiply the amount of system RAM by the expected overcommit ratio and add one of the following:

- 2GB of swap space for systems that have 4GB of RAM or less
- 4GB of swap space for systems that have between 4GB and 16GB of RAM
- 8GB of swap space for systems that have between 16GB and 64GB of RAM
- 16GB of swap space for systems that have between 64GB and 256GB of RAM

RHEV-M has the following hardware requirements:

- A dual-core CPU (quad core is preferred)
- 4GB of RAM (16GB of RAM is preferred)
- 25GB of local space (50GB is preferred)
- 1 NIC with bandwidth of at least 1Gbps

## 2.2 RHEV Software Requirements

- RHEV-M requires RHEL 6
- RHEV-M requires subscription to Red Hat Network (RHN)
- RHEV for desktops also requires access to either Microsoft® Active Directory® or Red Hat IPA
- RHEV-M administrator access has the following requirements:
  - Windows® XP, Windows 7, or Windows 2008 R2
  - Internet Explorer 7 or higher
  - .NET Framework 4
  - RHEV 3.0 includes a tech preview for non-Windows-dependent administrator access
- RHEV-M client portal access requires:
  - RHEL 5.5, 5.6, and 6
  - Windows 7, Windows 2008 R2, or Windows embedded for thin clients
  - Internet Explorer 7 and higher or Mozilla Firefox 3.5 and higher with the SPICE plug-in
- Virtualization hosts require RHEL 6 or RHEV-H 6

## 2.3 Storage Requirements for RHEV and Virtual Machines

RHEV supports the following storage protocols:

- NFS
- iSCSI
- Fibre Channel (FC)
- Fibre Channel over Ethernet (FCoE)

RHEV guests support the following disk types:

Table 1) Supported disk types.

NFS or SAN	Raw or Qcow2	Sparse or Preallocated
NFS	Raw	Preallocated
NFS	Raw	Sparse
NFS	Qcow2	Sparse
SAN	Raw	Preallocated
SAN	Qcow2	Preallocated
SAN	Qcow2	Sparse

A preallocated virtual disk in RHEV is “thick provisioned” in that the space equal to the presented size is allocated at the time of deployment.

A sparse file in RHEV is “thin provisioned” in that more space is presented to the guest than is actually allocated at the time of deployment. As the guest writes to disk, additional space is allocated automatically as necessary.

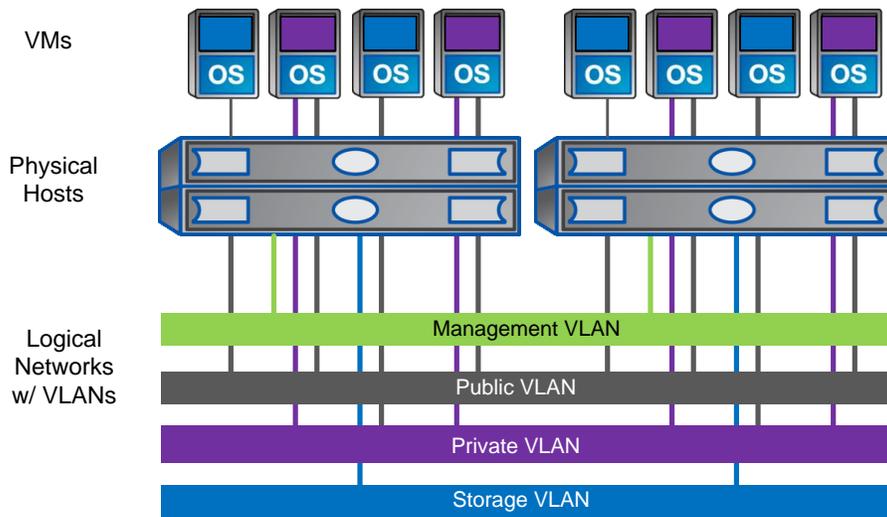
In the case of SAN (iSCSI, FCoE, and FC) storage in RHEV, the entire LUN is initialized with LVM2 as one large Volume Group. Individual disks are handled as Logical Volumes inside the Volume Group. For preallocated disks, the Logical Volume is the same size as the disk presented to the virtual guest. For sparse disks, the Logical Volume is much smaller than what is presented to the guest (starts at 1GB), and it grows automatically as needed.

Virtual disks backed by NFS storage are typically of “raw” type, but Qcow2 is supported. A Qcow2 file is initially created as close to zero in size, but has an initial raw format. As the disk size grows, additional layers are added to the disk, all formatted with Qcow2. In contrast to Qcow2, a raw disk has no additional layer of abstraction.

## 2.4 Network Layout for Red Hat KVM and NetApp Storage

There are multiple “correct” ways to implement a sound storage network. Figure 4 is an illustration of NetApp best practices. Note that the storage network is on a completely separate VLAN from the other data traffic.

Figure 4) Network layout best practices.



## 3 Hypervisor Installation

RHEV-M can manage both thin (RHEV-H) and thick (Red Hat KVM) hypervisors in the same environment. Sections 3.1 and 3.2 outline considerations when determining the best choice to make based on the use case.

In both cases, the hypervisor should be installed to and boot from a SAN LUN. This will allow the NetApp controller to provide the best possible storage efficiency through deduplication and thin provisioning. Ten hypervisors that have been installed on thin-provisioned and deduplicated LUNs will take the amount of space equivalent to a little more than one hypervisor. Additionally, SAN booting of the hypervisors means that the NetApp controller can easily clone a hypervisor template for rapid deployment.

### 3.1 RHEV-H and Use Cases

Red Hat has taken the appliance approach to hypervisors in its creation of RHEV-H. In RHEV 3.0, RHEV-H is an extremely stripped down RHEL 6 distribution. The entire RHEV-H ISO image takes up about 100MB in space because the vast majority of the RHEL packages have been removed.

The only configurable files are related to hostname and networking, since all other files and directories are read-only. Security pieces such as IPtables and SELinux are preconfigured to allow only the proper amount of access for storage, virtual machine, and management traffic.

Because the majority of the file system on RHEV-H is read-only and not many libraries are installed, no other packages may be updated or installed. From a security perspective, this is preferable in that there is

not much to exploit. However, if there is a requirement for a backup or additional monitoring agent, it is not possible to deploy it on RHEV-H.

### 3.2 RHEL 6 and Use Cases

Red Hat KVM can also be managed by RHEV-M and provides its own benefits and use cases. While RHEL 6 can't be stripped down to the extent of RHEV-H, it can be stripped down nonetheless. The single biggest use case for using Red Hat KVM within RHEV is the ability to install a backup agent such as the NetApp Snap Creator™ agent or even a third-party agent.

For any backup requirement that includes quiescing the virtual machine prior to triggering a Snapshot copy, this is a benefit. It should be noted that virtual machines hosted on RHEV-M can be quiesced via the RHEV API. However, if an additional monitoring agent is needed on the hypervisor, this is only possible on the thick hypervisor.

Refer to [TR-3848: "Best Practices for KVM, Red Hat Enterprise Linux 6, and NetApp Storage"](#) for the remaining guidelines for Red Hat KVM deployment and security.

### 3.3 RHEV-H Security Guidelines

Some of these guidelines are elementary, but are still worth mentioning.

- Create good passwords. For guidelines, refer to <https://access.redhat.com/kb/docs/DOC-9128>.
- Allow access to only virtualization administrators. Normal users should not be able to access anything on the RHEV-H host or be provided logins to RHEV-H.
- RHEV-H is already stripped of unnecessary services, packages, and inherently insecure protocols.

## 4 Red Hat Enterprise Virtualization Datastores

### 4.1 Raw Disk Image Files and QCOW2 Disk Image Files

RHEV supports the use of two different disk image formats: raw and qcow2. A raw disk image is faster and is the default for preallocated virtual disks. Qcow2 has some advanced features such as virtual machine-based snapshots, but they come at the cost of performance. The best practice when deploying virtual machines on RHEV is to preallocate disk files and allow the NetApp controller to thin provision, deduplicate data, and create Snapshot copies. It can do so much more quickly and efficiently without involving the hypervisor CPU cycles.

### 4.2 LUN-Based Datastores

Red Hat Enterprise Virtualization uses Logical Volume Manager (LVM2) for managing LUN-based datastores. LVM2 is typically used as a layer of abstraction between a block-based storage device (local disk, direct attach, SAN, and so on) and the file system. However, it is used in RHEV as a means of managing LUNs and individual virtual machines without using a file system. Essentially, a LUN is configured as an LVM Volume Group and virtual machines are created as individual Logical Volumes within the Volume Group. This precludes the need for a clustered file system or adding extra steps to any automation scripts for starting and stopping virtual machines.

### 4.3 NFS Datastores

Red Hat Enterprise Virtualization allows customers to leverage enterprise-class NFS arrays to provide datastores with concurrent access by all of the nodes utilizing the shared storage. The NetApp NFS provides high performance, the lowest per-port storage costs, and advanced data management capabilities.

#### NFS Datastores on NetApp

Deploying RHEV 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 FC. This architecture can result in a 10x 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 forth.

Through NFS, customers receive an integration of Red Hat KVM virtualization technologies with the WAFL<sup>®</sup> file system, 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, array-based thin provisioning, 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.

### 4.4 Datastore Comparison Tables

Differentiating what is available with each type of datastore and storage protocol can require considering many points. Table 2 compares the features available with each storage option.

Table 2) Datastore-supported features.

Capability/Feature	FC/FCoE	iSCSI	NFS
Format	LVM	LVM	NetApp WAFL
Optimal Queue Depth per LUN/File System	(Preconfigured on RHEV-H)	(Preconfigured on RHEV-H)	N/A
Available Link Speeds	4 and 8GB FC and 10GbE	1 and 10GbE	1 and 10GbE

Table 3 compares storage-related functionality of Red Hat KVM features across different protocols.

Table 3) Red Hat-supported storage-related functionality.

Capacity/Feature	FC/FCoE	iSCSI	NFS
Live Migration	Yes	Yes	Yes
Data Deduplication	Yes	Yes	Yes
Resize Datastore	Grow only	Grow only	Grow, autogrow, shrink
Thin Provision Datastores	Yes	Yes	Yes
NetApp Snapshot	Yes	Yes	yes
Restore Datastores and VMs from SnapMirror <sup>®</sup> and SnapRestore <sup>®</sup> Technologies	Yes	Yes	Yes

Capacity/Feature	FC/FCoE	iSCSI	NFS
Boot from SAN	Yes	Yes with HBAs	No

This is the default setting for all VMs on NetApp NFS.

## 5 RHEV Guest Configuration

### 5.1 File System Alignment Overview

In any virtual environment, there are a number of layers of abstraction between the 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 extra 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 is applicable for any storage vendor. It is a simple 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."](#)

Without proper 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 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 properly with the layers below.

Microsoft Windows Server 2008, Windows 7, and RHEL 6 all align properly 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 time for proper file system alignment.

### Thick or 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 thin 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 maintains an efficient storage environment. The best practice is to thick provision the KVM guests and thin provision and deduplicate the underlying storage.

### 5.2 Create a RHEV Guest Template

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

The concept is almost identical to creating a template for Red Hat KVM hosts. The base image is created via Kickstart, and the image then is "made generic." When new guests are needed, the RHEV native cloning tools can be used to provision new virtual machines. For Microsoft guests, "sysprep" should be used to make the guest sufficiently generic.

### 5.3 Kickstart

It is a best practice to use Kickstart to build a RHEV guest for the first time. Kickstart provides a semi-automated way to deploy Red Hat Enterprise Linux in a highly customizable way. Once a RHEV guest is created and made generic, it is a best practice to repeat the deployment by using NetApp FlexClone® technology.

### 5.4 FlexClone

FlexClone is highly efficient in many use cases, but for now can only be used for cloning RHEL-based “thick” hypervisors. The use of FlexClone to clone virtual machines within RHEV is not yet supported.

### 5.5 Guest Timing Issues

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

### 5.6 Security Considerations

Much like the Red Hat host security, the focus is on firewall, mandatory access control, unnecessary services, and insecure services.

RHEL guests should have IPtables running and SELinux enabled. As a best practice, configure the necessary ports or controls rather than disable that layer of security. Limit the packages installed to those that are necessary. And absolutely avoid RSH, telnet, and FTP in favor of SSH, SCP, and SFTP.

Most Windows guests have some form of firewall as well. Again, it is a best practice to open necessary ports, rather than disable that layer of security. Additionally, the running services should be reviewed and, if necessary, you should shut down unnecessary services.

### 5.7 RHEV Virtualized Guest Support

RHEV 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 newer (32-bit only)
- Windows 7 (32-bit and 64-bit)
- Windows Server 2003 Service Pack 2 and newer (32-bit and 64-bit)
- Windows Server 2008 (32-bit and 64-bit)
- Windows Server 2008 R2 (64-bit only)

RHEV supports the following virtual hardware configurations (Table 4).

Table 4) RHEV supported virtual hardware configurations.

Hardware	Configuration
Virtual CPUs	Up to 16 per guest
Virtual RAM	Up to 256GB per 64-bit guest
Virtual RAM	Up to 4GB per 32-bit guest

Hardware	Configuration
Virtual storage devices	Up to 8 per guest
Virtual network devices	Up to 8 per guest
Virtual PCI devices	Up to 32 per guest

## 5.8 ParaVirtualized Drivers Support

The paravirtualized block and network drivers (the virtio drivers) support the following operating systems and versions. The paravirtualized drivers increase performance for 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)

## 6 NetApp Storage Best Practices for RHEV

### 6.1 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 unused but provisioned 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, there will not be enough storage to satisfy the requests.

#### NetApp Thin Provisioning Options

NetApp thin provisioning can be used with RHEV 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's. Enable thin provisioning (at all layers) on the NetApp controller and then thick provision the virtual machines in RHEV. This will provide the proper balance between performance and storage efficiency.

When enabling NetApp thin-provisioned LUNs, NetApp recommends deploying these LUNs in NetApp FlexVol<sup>®</sup> volumes that are also thin provisioned with a capacity that is two times 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.

## 6.2 Storage Array Deduplication

Like other modern virtualization platforms, Red Hat Enterprise Virtualization 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 creates a new virtual machine (and configuration file) based on the template and its configuration file.

NetApp offers a data deduplication technology called 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. It can be seamlessly introduced into a virtual data center without having to make any changes to the way the RHEV is maintained or administered. Deduplication 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 greatest 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 dedupe on thin-provisioned LUNs for Red Hat Enterprise Virtualization, NetApp recommends deploying these LUNs in FlexVol volumes that are also thin provisioned with a capacity that is two times 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 with LUNs, when deduplication is enabled with NFS, the storage savings are immediately available and recognizable. The benefit of dedupe is transparent to storage and virtualization admin teams. No special considerations are required for its usage with Red Hat Enterprise Virtualization.

## 7 Storage Network Best Practices for RHEV

### 7.1 Storage Architecture Concepts

Prepare the following before configuring the storage array to run a virtual infrastructure.

- Separate networks for storage array management and storage I/O. This concept applies to all storage protocols but is very pertinent to Ethernet-based deployments (NFS, iSCSI, FCoE). The separation can be physical (subnets) or logical (VLANs), but it must exist.
- If 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 LACP VIFs over multimode VIFs whenever possible.

## 7.2 Production Ethernet Storage Networks

The goal of any storage network is to provide uninterrupted service to all nodes connecting to it. This section is focused primarily on how to establish a highly available Ethernet storage network. The reasons for focusing on Ethernet are twofold. First, Fibre Channel storage networks provide a single service—Fibre Channel. As such, these single-purpose networks are simpler to design and deploy in a highly available configuration. Second, the current industry trend is solely focused 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 Enterprise Virtualization all provide support for 10GbE. An advantage of 10GbE is its 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 either can be dedicated to a single VLAN or can 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 hypervisor with physical link redundancy. By adding multiple VLANs, you can 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 restriction of the iSCSI protocol 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 used over the same interface, make sure that sufficient throughput can be provided for all traffic.

### Routing and IP Storage Networks

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

### Separate Ethernet Storage Networks

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 routing 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 RHEV, this means that a separate logical network should be created for each storage network with VLAN tagging where possible.

## NetApp Virtual Interfaces

A VIF (Data ONTAP 7) or ifgrp (Data ONTAP 8) is a mechanism that supports the 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 “ifgrp,” 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 utilizes 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.

## Jumbo Frames

It is a NetApp best practice to use jumbo frames for Ethernet storage traffic, especially NFS. Standard frames require that NFS datagrams 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, thus removing the need to break up and reassemble. Jumbo frames should be configured for the following types of data streams:

- Storage traffic and isolated, nonrouted VLANs for NFS, CIFS, or iSCSI data
- Replication network and isolated, nonrouted VLANs 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 created virtual interface that utilizes the bridge will recognize that the bridge uses jumbo frames and will automatically follow suit.

## 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 ifgrp (Data ONTAP 8).

Red Hat also has a native multipathing driver called Device Mapper Multipath I/O (DM-MPIO). It 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 as well as any Red Hat hypervisors (thick and thin). Each NIC or HBA on the Red Hat hypervisor will need to have its own separate path to the storage, thereby eliminating any single point of failure along the path. In turn, this will require the use of DM-MPIO as well as channel bonding if the storage is over Ethernet.

## 8 Management Best Practices for RHEV and NetApp

The management servers below can all be virtualized on a separate group of “infrastructure hosts.” By virtualizing the management servers, they gain the same benefits as the production virtual machines such as mobility, availability, and centralized data management on the NetApp controller(s).

### 8.1 RHEV-M

Red Hat Enterprise Virtualization Manager (RHEV-M) is the primary means of interacting with RHEV-H, virtual machines, logical networks, RHEV datastores, and all other aspects of the RHEV environment. Administrative access, power user access, and VDI access should all go through RHEV-M.

### 8.2 REST API

RHEV-M uses a RESTful API that allows interacting with all aspects of RHEV without having to go through the Web portal. All automation tools and scripts should write to this API.

### 8.3 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 to keep up with security patches as well as compliance.

RHN Satellite is essentially an on-site instantiation of RHN. Instead of having 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 benefits such as reduced external network traffic as well as 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 in the infrastructure RHEL 6 KVM hosts.

### 8.4 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 an RHEL or a Windows guest in the infrastructure RHEL 6 KVM hosts.

### 8.5 Kickstart Server

Kickstart is a means of providing semi-automated RHEL installations and can be deployed via CD and answer file, HTTP, NFS, or FTP. It is a best practice to deploy Kickstart as a server on an RHEL KVM guest or as part of an RHN Satellite server within the infrastructure hosts.

### 8.6 Scaling Out the Environment

When a discussion of a “scalable” environment arises, it typically is described as “scaling up” (ease of upgrading to more powerful systems) and “scaling out” (ease of growth).

Proper deployment of 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.

Arguably, the ease of scaling out is more important. Typically, as virtual environments grow, it becomes necessary to balance workloads between controllers and react quickly to sudden increases in activity. For this reason, NetApp highly recommends using NetApp MultiStore<sup>®</sup> software in an RHEV environment.

MultiStore provides an additional layer of storage virtualization by way of multiple lightweight instances of Data ONTAP that are created in what are referred to as “vFiler<sup>®</sup> units.” Each vFiler unit maintains its own network routing tables, IP storage, and authentication. Although there are many benefits to using MultiStore in the context of scaling out, vFiler units can be migrated dynamically from one NetApp controller to another.

Additionally, MultiStore can be used to support multiple RHEV environments simultaneously while keeping them completely separated from each other at the storage layer.

FlexClone can be used to create writable Snapshot copies of FlexVol volumes, LUNs, and, in the case of RHEV-H, to rapidly deploy hypervisors. The use of FlexClone to clone individual virtual machines within RHEV is not yet supported.

## 9 Backup and Disaster Recovery Best Practices for RHEV

### 9.1 NetApp Snapshot

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

In general, the best practice for a Snapshot reserve in a virtual environment is to set it to 0% and to disable the Snapshot copy schedule. To maintain file consistency, it is best to quiesce (pause) the virtual guests, create the Snapshot copy, and then resume the virtual guests. (See section 9.2, “NetApp Snap Creator.”) Following are some key points about Snapshot copies in the context of an RHEL 6 KVM environment.

- Snapshot copies provide 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.

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

### 9.2 NetApp Snap Creator

NetApp Snap Creator is an extensible backup framework that works with many applications and platforms, such as Red Hat KVM or an RHEV guest. The framework enables not only triggering NetApp Snapshot copies of volumes (datastores), but also any activities that need to occur before and after the Snapshot copy is taken. Additionally, it can trigger NetApp 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 majority that need to be addressed in a virtual environment.

## Crash-Consistent Backups with Snap Creator

This use case creates 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 archive. Although this might be suitable for capturing the current state, the restore would depend on the file system logs on the guest operating system for proper replay.

Snap Creator does not need to talk to RHEV, the hypervisor, or the application in this scenario; it only needs to know when to trigger Snapshot on the NetApp controller and mirror the Snapshot copy to another controller. The Snap Creator agent is not used in this case.

## Application-Consistent Backup with Snap Creator

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

## 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 the hypervisor to pause the virtual machine, creates 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 (<5 seconds). The Snapshot copies can then be mirrored to another controller for backup or archive.

The Snap Creator agent needs to be installed on the virtual machine and the hypervisor in this use case. In the case of RHEV, this means that the hypervisor node cannot run RHEV-H; it must be Red Hat KVM as managed by RHEV-M.

## Deploying Snap Creator Server

The Snap Creator server should be deployed on a separate server or virtual machine outside of the Red Hat KVM and RHEV environment that it is backing up. This is because if the Snap Creator server is running on an infrastructure virtual machine, it could possibly quiesce the virtual machine that it is running on. It would then not be able to resume itself.

Deploying it within the RHEV production virtual machines might make it more difficult to trigger actions on the infrastructure cluster. Therefore, the best practice for deploying Snap Creator server is to place it on a server outside the control of the servers and virtual machines that it is meant to back up.

## 9.3 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 doesn't protect the Snapshot copy of the RHEV datastore on its own; this is where NetApp SnapMirror is added to the RHEV backup strategy.

SnapMirror is a data replication product 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 the performance impact on storage and the 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 nonpeak load times. Finally, data can also be backed up from a SnapMirror partner to tape, VTL, or another archive device.

## 9.4 MetroCluster

MetroCluster™ software 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 to the other site. Like an active-active pair, one controller can handle storage requests for both. However, if the entire site becomes unavailable, the mirrored data can be accessed immediately because one controller takes on the identity of both controllers simultaneously.

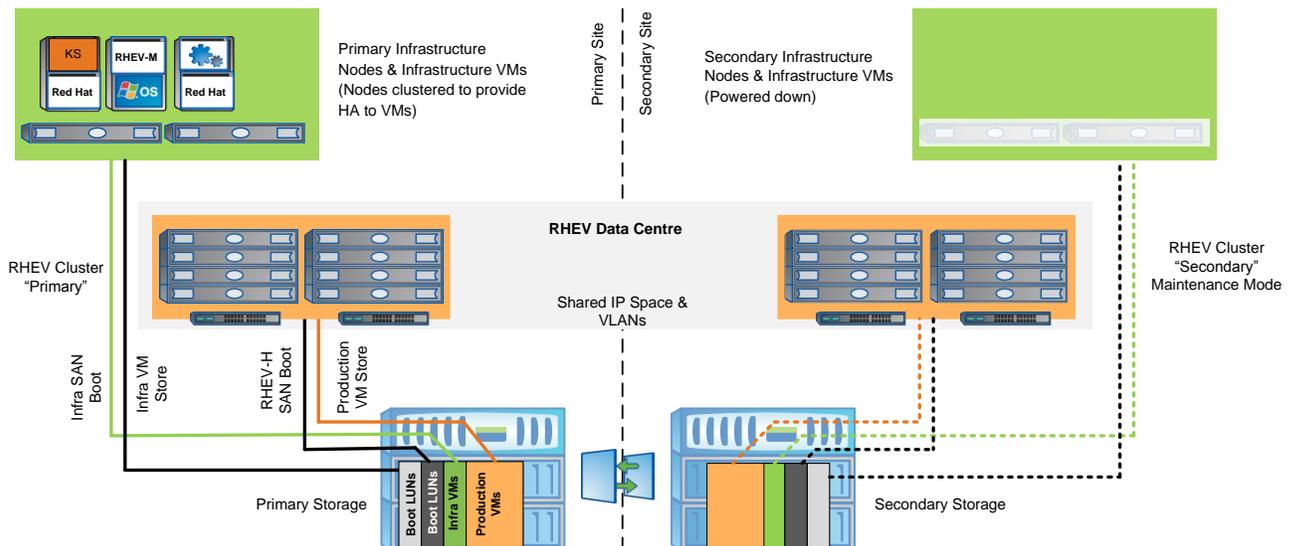
If business needs dictate the highest level of uptime and continuity for RHEV, the NetApp best practice is to use MetroCluster as the storage foundation.

### Site Failover Considerations for RHEV

When planning a site failover strategy for RHEV, it is important to note that RHEV-M must “think” it is being brought up in its original site. For this reason, it is a best practice to deploy RHEV-M on a virtualized instance of RHEL 6 attached to NetApp shared storage. Then, if a site failover is initiated, the virtual machine itself will run on any RHEL 6 hypervisor.

From a hypervisor perspective, it is best to create “primary” and “secondary” RHEV clusters for each group of servers that need the level of service provided by site failover. Both RHEV clusters are part of the same RHEV data center, but the hypervisors in the secondary cluster are placed in “maintenance mode” until they are needed. Additionally, it is important to create the same IP networks and VLANs at both sites.

Figure 5) Site failover.



The net result in a site failover is that all critical virtual machines are back up in minutes. Not having RHEV hypervisors (thick or thin) that are already registered in RHEV-M at the secondary site adds to the time and steps required to get back to a running state.

## 9.5 Traditional Backup Methods

Although using NetApp Snapshot, Snap Creator, and SnapMirror technologies is a best practice, they are not required 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 includes two means of data backup in Data ONTAP 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.

## 10 Conclusion

RHEV offers customers a consistent, robust, and viable platform for hosting VMs. NetApp provides native support for storage protocols as well as the means to scale, protect, and enable RHEV. Together, RHEV and NetApp provide customers with a solid and scalable virtualization platform that still allows flexibility.

Although this technical report provides best practices as they apply to RHEV and NetApp, it is not meant to be the final word on implementation. You may need additional expertise from Red Hat, NetApp, and/or a network vendor for a definitive solution that fits a specific business need.

## References

- DataFabric Manager Server: Operations Manager Administration Guide  
[http://now.netapp.com/NOW/knowledge/docs/DFM\\_win/rel40/html/software/opsmgr/frameset.html](http://now.netapp.com/NOW/knowledge/docs/DFM_win/rel40/html/software/opsmgr/frameset.html)
- Data ONTAP 7.3 Data Protection Online Backup and Recovery Guide  
<https://now.netapp.com/NOW/knowledge/docs/ontap/rel734/pdfs/ontap/onlinebk.pdf>
- Data ONTAP 8.0 7-Mode High-Availability Configuration Guide  
<http://now.netapp.com/NOW/knowledge/docs/ontap/rel80/pdfs/ontap/haconfig.pdf>
- Red Hat Enterprise Linux 6 Deployment Guide  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Deployment\\_Guide/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html)
- Red Hat Enterprise Linux 6 Installation Guide  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Installation\\_Guide/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/index.html)
- Red Hat Enterprise Linux 6 Storage Administration Guide  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Storage\\_Administration\\_Guide/](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Storage_Administration_Guide/)
- Red Hat Enterprise Virtualization 3 Administration Guide  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Virtualization/3.0/html/Administration\\_Guide/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Virtualization/3.0/html/Administration_Guide/index.html)
- Red Hat Enterprise Virtualization 3 Installation Guide  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Virtualization/3.0/html/Installation\\_Guide/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Virtualization/3.0/html/Installation_Guide/index.html)
- TR-3446: SnapMirror Async Overview and Best Practices [Guide](#)  
<http://media.netapp.com/documents/tr-3446.pdf>
- TR-3548: Best Practices for MetroCluster Design and Implementation  
<http://media.netapp.com/documents/tr-3548.pdf>
- TR-3649: Best Practices for Secure Configuration of Data ONTAP 7G  
<http://media.netapp.com/documents/tr-3649.pdf>

- TR-3703: Using SnapMirror for Disaster Protection with Block Access Protocols  
<http://media.netapp.com/documents/TR-3703.pdf>
- TR-3747: Best Practices for File System Alignment in Virtual Environments  
<http://media.netapp.com/documents/tr-3747.pdf>
- TR-3800: Fibre Channel over Ethernet (FCoE) End-to-End Deployment Guide  
<http://media.netapp.com/documents/TR-3800.pdf>
- TR-3848: Best Practices for KVM and Red Hat Enterprise Linux on NetApp Storage  
<http://media.netapp.com/documents/tr-3848.pdf>

NetApp provides no representations or warranties regarding the accuracy, reliability, or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.

Go further, faster®

© 2012 NetApp, Inc. All rights reserved. No portions of this document may be reproduced without prior written consent of NetApp, Inc. Specifications are subject to change without notice. NetApp, the NetApp logo, Go further, faster, DataFabric, Data ONTAP, FlexClone, FlexVol, MetroCluster, MultiStore, Snap Creator, SnapMirror, SnapRestore, Snapshot, vFiler, and WAFL are trademarks or registered trademarks of NetApp, Inc. in the United States and/or other countries. Linux is a registered trademark of Linus Torvalds. Intel is a registered trademark of Intel Corporation. Windows, Microsoft, and Active Directory are registered trademarks of Microsoft Corporation. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. TR-3914-0312