



# Red Hat Enterprise Linux Built on FlexPod Deployment Guide

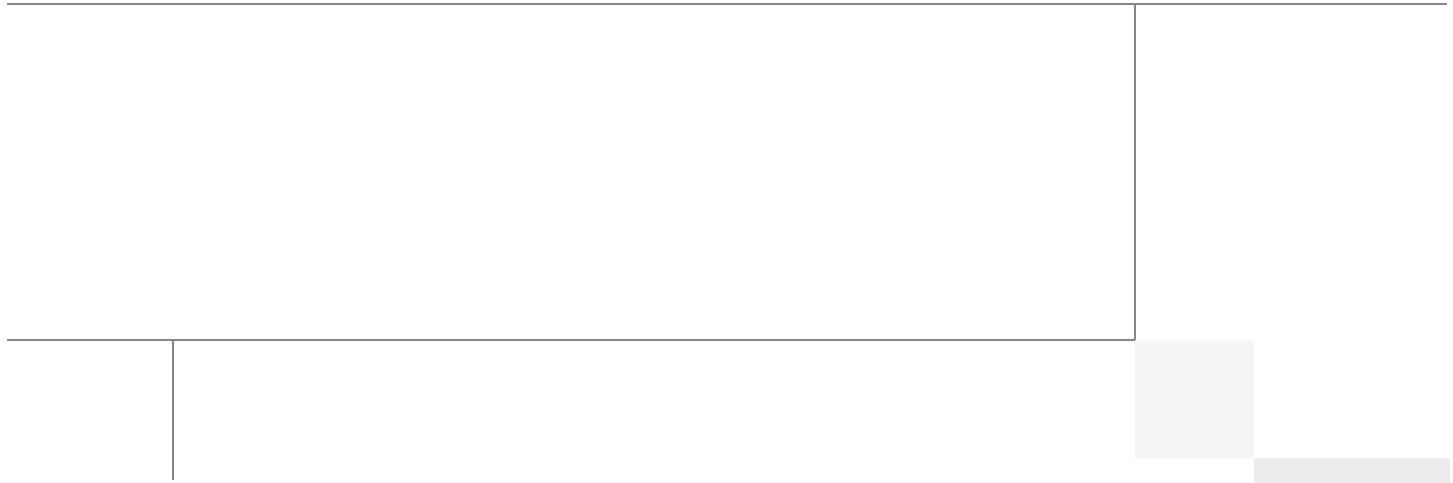
Last Updated: August 31, 2011



Cisco  
Validated  
Design



Building Architectures to Solve Business Problems



## About the Authors



Ramesh Isaac

Ramesh Isaac, Technical Marketing Engineer, Systems Development Unit, Cisco

Ramesh Isaac has worked in data center and mixed-use lab settings since 1995. He started in information technology supporting UNIX environments and focused on designing and implementing multi-tenant virtualization solutions in Cisco labs over the last couple of years. Ramesh holds certifications from Cisco, VMware, and Red Hat.



Jon Benedict

Jon Benedict, Reference Architect, Infrastructure and Cloud Engineering, NetApp

Jon Benedict is a reference architect in the Infrastructure & Cloud Engineering team at NetApp. Jon is largely focused on designing, building, and evangelizing cloud and shared storage solutions based around NetApp for enterprise customers. Prior to NetApp, he spent many years as a consultant, integrator, and engineer with expertise in Unix and Linux. Jon holds many industry certifications including several from Red Hat.



Chris Reno

Chris Reno, Reference Architect, Infrastructure and Cloud Engineering, NetApp

Chris Reno is a reference architect in the NetApp Infrastructure and Cloud Enablement group and is focused on creating, validating, supporting, and evangelizing solutions based on NetApp products. Before being employed in his current role, he worked with NetApp product engineers designing and developing innovative ways to perform Q&A for NetApp products, including enablement of a large grid infrastructure using physical and virtualized compute resources. In these roles, Chris gained expertise in stateless computing, netboot architectures, and virtualization.



Aleksandr Brezhnev

Aleksandr Brezhnev, Managing Principle Architect, Red Hat, Inc.

Aleksandr Brezhnev is a platform solutions architect at Red Hat. He is currently focused on partner enablement and solution development based on Red Hat virtualization and cloud products. Aleksandr is an expert in system tuning and database and application optimization on Red Hat platforms. He has been with Red Hat for more than 10 years and in prior roles he was a consulting engineer and technology development manager for strategic accounts in financial and healthcare verticals.

# About Cisco Validated Design (CVD) Program

---

The CVD program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information visit <http://www.cisco.com/go/designzone>.

ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, "DESIGNS") IN THIS MANUAL ARE PRESENTED "AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE. IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at <http://www.cisco.com/go/trademarks>. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

NetApp, the NetApp logo, Go further, faster, AutoSupport, DataFabric, Data ONTAP, FlexPod, MultiStore, Near-Store, NOW, and vFiler are trademarks or registered trademarks of NetApp, Inc. in the United States and/or other countries.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Red Hat Enterprise Linux Built on FlexPod Deployment Guide

© 2011 Cisco Systems, Inc. All rights reserved.



# Red Hat Enterprise Linux Built on FlexPod Deployment Guide

---

## Red Hat Enterprise Linux Built on FlexPod Overview

Industry trends indicate a vast data center transformation toward shared infrastructures. Enterprise customers embrace agile platforms supporting flexible allocation of computational resources between dedicated physical servers, heterogeneous virtualized environments, and clouds.

FlexPod™ is a predesigned, base configuration that is built on the Cisco® Unified Computing System™ (UCS™), Cisco Nexus® data center switches, and NetApp® FAS storage components. Combined with the Red Hat® Enterprise Linux® operating system and its add-on options such as Scalable File System, High-Availability, and Resilient Storage, FlexPod can be used as a trusted platform for mission-critical applications. FlexPod is a baseline configuration, but also has the flexibility to be sized and optimized to accommodate many different use cases. Red Hat Enterprise Linux on FlexPod can be used in standalone or cluster configurations on physical UCS servers or as a guest operating system in virtualized environments.

The FlexPod Deployment Guide is available at:

[http://www.cisco.com/en/US/docs/solutions/Enterprise/Data\\_Center/Virtualization/flexpod\\_deploy.html](http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/Virtualization/flexpod_deploy.html).

NetApp partners may find additional FlexPod information at: <https://fieldportal.netapp.com/>.

## Audience

This document provides recommendations for deploying a standalone Red Hat Enterprise Linux 6 server and a high availability cluster on FlexPod. The intended audience for this document includes, but is not limited to, sales engineers, field consultants, professional services personnel, IT managers, partner engineering personnel, and customers who want to deploy a Red Hat Enterprise Linux on FlexPod hardware.



### Note

For more detailed deployment information, Cisco and NetApp partners should contact their local account teams or visit <http://www.netapp.com/us/technology/flexpod/>.



### Corporate Headquarters:

Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Copyright © 2011 Cisco Systems, Inc. All rights reserved.

## FlexPod Architecture

Cisco and NetApp have provided documentation relating to best practices for building the FlexPod shared infrastructure stack. As part of the FlexPod offering, Cisco and NetApp designed a reference architecture with a technical specifications sheet and bill of materials that is highly modular or “pod-like”.

The Red Hat Enterprise Linux 6 deployment model leverages the basic FlexPod configuration. All recommendations in the following sections need to be considered in the context of the *FlexPod Deployment Guide* at:

[http://www.cisco.com/en/US/docs/solutions/Enterprise/Data\\_Center/Virtualization/flexpod\\_deploy.html](http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/Virtualization/flexpod_deploy.html).

For the Red Hat Enterprise Linux 6 deployment on FlexPod, the hardware consists of the following components:

- Boot LUNs configured on NetApp storage and SAN attached through Cisco Fabric Interconnects
- VLANs implemented with Cisco Nexus switches
- Physical compute systems implemented as service profiles on Cisco UCS blades equipped with Cisco UCS Virtual Interface Cards
- Cisco UCS Manager, which can be used to define service profiles on Cisco UCS blades, get console access, select boot device order, attach virtual DVD for installation of the operating system, and as a fencing device in cluster configurations.
- Data volumes exported from NetApp storage through NFS and iSCSI

The default hardware is detailed in the FlexPod technical specifications. All systems and fabric links feature redundancy, providing end-to-end high availability. In addition, Red Hat Enterprise Linux utilizes multipath access to boot LUN and network channel bonding, and can be configured with the High Availability Add-On, providing failover cluster resources and services. Each component in the default base design can be scaled flexibly to support a specific customer’s business requirements.

## FlexPod Deployment Procedures

The base FlexPod deployment procedures are described in the *FlexPod Deployment Guide* at:

[http://www.cisco.com/en/US/docs/solutions/Enterprise/Data\\_Center/Virtualization/flexpod\\_deploy.html](http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/Virtualization/flexpod_deploy.html). The remainder of this document guides the reader through the specific steps to deploy Red Hat Enterprise Linux 6 and the High Availability Add-On on the base FlexPod architecture.

### Create Boot LUNs on NetApp Storage

This section describes additional procedures necessary on the NetApp controllers to provision boot storage devices for Red Hat Enterprise Linux 6 on Cisco UCS service profiles. At the end of this workflow the following objectives should be met:

- Fibre Channel target ports defined
- Fibre Channel interface groups (igroups) defined for each service profile
- Boot LUNs allocated for each Cisco UCS service profile
- Boot LUN mapped to associated Cisco UCS service profile

The following process outlines the steps necessary:

- Create igroups.
- Create LUNs for Cisco UCS service profiles.
- Map LUNs to igroups.

## Data Volumes on NetApp Storage

This section describes the procedures for configuring data volumes on NetApp storage that can be used by Red Hat Enterprise Linux 6 servers through NFS and iSCSI. At the end of this workflow the following objectives should be met:

- iSCSI target ports defined
- iSCSI interface groups (igroups) defined for each server
- Data LUNs allocated for each server
- Data LUNs mapped to associated server
- NFS exports defined for each server

The following process outlines the steps necessary:

- Create igroups.
- Create FlexVol<sup>®</sup> volume to contain LUNs.
- Create LUNs for the Cisco UCS service profiles.
- Map LUNs to igroups.
- Create FlexVol volumes for NFS exports.
- Define IP address access for NFS exports.

## Cisco UCS Polices and Profiles

This section provides the procedure for configuring the Cisco UCS in a FlexPod environment for Red Hat Enterprise Linux 6 and High Availability Add-On deployment. This workflow should utilize or recreate the configuration resources listed in the *FlexPod Deployment Guide* at:

[http://www.cisco.com/en/US/docs/solutions/Enterprise/Data\\_Center/Virtualization/flexpod\\_deploy.html](http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/Virtualization/flexpod_deploy.html) in the section “Cisco Unified Computing System Deployment Procedure.”

- Create an Ethernet Adapter Policy suitable for Red Hat Enterprise Linux 6.
- Configure Templates from defined Pools, Policies, Interface Templates, and Orgs.
- Create Service Profiles from Templates.



### Note

An Ethernet Adapter Policy for Red Hat Enterprise Linux 6 is made to disable the Large TCP Receive Offload (LRO) function on the adapter. The onboard LRO can cause problems with forwarding traffic to virtual machines hosted on Red Hat Enterprise Linux 6. Also the network stack in the operating system contains hardware independent LRO optimization, which makes onboard LRO redundant.

The Ethernet Adapter Policy for Red Hat Enterprise Linux 6 is created by:

- Creating an Ethernet Adapter Policy in UCSM called “RedHat” with settings identical to the default Linux Ethernet Adapter Policy and disabling Large TCP Receive Offload.

UCS Service Profile Templates for Red Hat Enterprise Linux 6 hosts require creating new or using existing Pools, Policies, Interface Templates, and Orgs described in the *FlexPod Deployment Guide*:

- Sub-Org for hosts
- UUID Pool
- WWNN Pool
- WWPN Pools
- vNIC Templates
- vHBA Templates
- Boot Policies
- Server Pools
- MAC Pools

Service Profiles are provisioned from the Service Profile Template to blades in the configured server pool. Once a Boot LUN is associated with a specific Service Profile, the hardware is abstracted to provide a stateless computing resource that can be re-associated with any comparable UCS blade.

## Cisco SAN Connectivity

This section describes additional configuration procedures on the Cisco Nexus 5548 platforms in the FlexPod environment required to provide connectivity between boot LUNs configured on NetApp storage and Cisco UCS service profiles for Red Hat Enterprise Linux 6 servers. This section has the following objectives:

- Define Fibre Channel aliases for Service Profiles vHBAs and NetApp controller ports
- Establish Fibre Channel Zoning and working sets

The following steps should be performed on each Nexus platform:

- Identify controller interfaces and vHBAs from powered on hosts in flogi database or collect independently from Filer and configured Service Profiles.
- Create meaningful device aliases for the pwwn of the Service Profile vHBAs and NetApp controller interfaces in the device-alias database.
- Define zones containing the device aliases for the respective Service Profile vHBAs and the NetApp controller interfaces they connect to for their boot LUN.
- Add the zones as members to the zoneset and activate the zoneset.

## Cisco Networking

This section describes the procedures that must be implemented on Cisco Nexus switches to support Layer 2 multicast communications required by the Red Hat Enterprise Linux 6 High Availability Add-On. To enable this Layer 2 multicast communication, there must be an IGMP snooping querier enabled.

When PIM is not enabled on an interface because the multicast traffic does not need to be routed, you must configure an IGMP snooping querier to send membership queries. The configured querier IP should be a non-routed address that could be assigned from a VLAN requiring the multicast communication. To enable an IGMP snooping querier you must:

- Configure **ip igmp snooping querier** <address> for any VLAN requiring multicast that does not have a Layer 3 interface.

# Software Deployment Procedures

## Red Hat Enterprise Linux 6

This section describes the installation of Red Hat Enterprise Linux 6 Server on Cisco UCS in a FlexPod environment and should result in the following:

- A functional standalone Red Hat Enterprise Linux 6 Server
- Multipath access to the Fibre Channel attached boot LUN
- Network connectivity according to the Cisco UCS profile
- Network channel bonding setup compatible with Cisco UCS
- NFS connectivity to data volume on NetApp storage
- iSCSI connectivity to data volume on NetApp storage

The following outlines the process to install Red Hat Enterprise Linux 6 in a FlexPod environment:

- Red Hat Enterprise Linux 6 deployment through the Cisco UCS Manager KVM Console.

The operating system distribution media in the form of an ISO image is mounted to the Cisco UCS service profile through the KVM console. Alternatively, the operating system can be deployed from the PXE-boot server.

A Fibre Channel attached LUN preconfigured on the NetApp storage is used as a system boot device. Multipath and single path configurations are supported.

The operating system installation process is described in the *Red Hat Enterprise Linux 6 Installation Guide* at: [http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/index.html).

- Set up Ethernet interface options for the Cisco M81KR virtual interface card.
- Configure network channel bonding.
- Set up VLAN interfaces.
- Set up routing.
- Set up DNS.
- Review and configure the system firewall.
- Configure NTP service.
- Review and configure Red Hat Subscription Manager.
- Register the system with the Red Hat Network using Red Hat Subscription Manager.
- Activate subscriptions assigned to the system.
- Review and configure access to YUM repositories.
- Review and configure YUM plug-ins.
- Perform system update.
- Add/remove software packages and package groups.
- Review and configure the default runlevel.

- Enable and disable services.
- Review and configure authentication.
- Review and configure the server's power management profile.
- Review confined services and make adjustments to SELinux configuration.
- Review application resource requirements and make adjustments to cgroups configuration.
- Review and configure files in the /etc/sysconfig directory.
- On a system requiring access to data volumes through NFS, make configurations in the /etc/fstab file.
- Configure and enable Automounter service, if necessary.
- Set up Jumbo Frames MTU on network interfaces dedicated for NFS or iSCSI traffic.
- On a system requiring access to data volumes through iSCSI, install the iSCSI Storage Client group of software packages.
- Perform discovery of iSCSI targets configured on NetApp storage.
- Review and configure iSCSI session authentication parameters.
- Login to the discovered iSCSI targets.
- Review iSCSI session negotiated parameters and make configuration adjustments, if necessary.
- Enable iSCSI service.
- Configure Volume Group(s) and Logical Volume(s) on iSCSI attached storage.

Detailed descriptions of the Red Hat Enterprise Linux 6 configuration procedures are available in the following guides:

- *Red Hat Enterprise Linux 6 Deployment Guide*
- *DM Multipath*
- *Logical Volume Manager Administration*
- *Managing Confined Services*
- *Power Management Guide*
- *Resource Management Guide*
- *Security Guide*
- *Storage Administration Guide*

## High-Availability Add-On

This section describes the installation of Red Hat Enterprise Linux 6 High-Availability Add-On software on Cisco UCS in a FlexPod environment and configuration of a sample MySQL database failover service. It should result in the following:

- Two-node failover cluster setup
- Node fencing through Cisco UCS Manager
- IP address failover resource for MySQL database server connections
- HA-LVM resource for database data
- File system failover resource for MySQL database files

- MySQL failover service

**Note**

The Red Hat Enterprise Linux High-Availability Add-On supports cluster node fencing through Cisco UCS Manager beginning with version 6.1. However, support for fencing Cisco UCS service profiles created within a suborganization hierarchy requires an update to 6.1.z or 6.2.

The following outlines the process of installing the Red Hat Enterprise Linux 6 High-Availability Add-On in a FlexPod environment and sample configuration of a two-node failover cluster with MySQL database service:

- Verify two Cisco UCS service profiles are assigned to be cluster nodes.  
They should be installed with Red Hat Enterprise Linux 6.1 or later and subscribed to the Red Hat Network or have access to a YUM repository with High-Availability Add-On software.  
The network connectivity should be enabled through the network service; the NetworkManager service must be disabled or uninstalled.
- Verify that cluster nodes have open iSCSI sessions with the same iSCSI target from NetApp storage. Perform iSCSI Storage Client installation and configuration, if necessary.
- Verify DNS configuration for all cluster nodes, their cluster interconnect interfaces, and virtual IP failover resources. Reconfigure DNS server or add static records into the /etc/hosts files, if necessary.
- Verify that cluster nodes have IP network connectivity through the VLAN dedicated for cluster interconnect.
- Verify network connectivity between cluster nodes and the Cisco UCS Manager that will be used as a fencing device.
- Review cluster quorum requirements and make a decision about the usage of the Quorum Disk (Qdisk). In general, the Qdisk is not required, but it may be necessary if the cluster has to sustain the loss of all but one of its nodes. The two-node cluster is a special case.
- Gather data required for configuration of cluster interconnect, cluster fencing, failover resources, and services.
- Configure Cisco Nexus switches for multicast traffic propagation through a VLAN assigned for cluster interconnect.
- Install the High Availability group of packages on cluster nodes.
- Adjust the system firewall configuration to enable cluster communications.
- Disable ACPI service on cluster nodes.
- Enable ricci service on cluster nodes.
- Create the cluster configuration file.
- Add records about all cluster nodes.
- Add records about fencing methods to the cluster nodes.
- Define fencing device with the Cisco UCS fencing agent.
- Add fencing instances to the cluster nodes.
- Specify two-node cluster configuration with required options.
- Synchronize configuration files on cluster nodes and activate the cluster.
- Verify cluster quorum status.

- Verify node fencing configuration.
- Enable the cman and rgmanager services after successful verification of the basic cluster setup.

At this point the cluster should be ready for deployment of failover resource groups. The Red Hat Enterprise Linux High-Availability Add-On contains templates for various infrastructure services, databases, and application servers that can be deployed as failover resource groups. The details are available in the *Red Hat Enterprise Linux 6 Cluster Configuration Guide*.

The following steps outline a sample MySQL database failover configuration:

- Configure HA-LVM mode for a shared volume group on the cluster nodes.
- Add failover domain definition to the cluster configuration.
- Add nodes to the failover domain.
- Add the failover resource definitions required to support high availability MySQL database service:
  - IP address
  - LVM
  - ext4 file system
  - MySQL service
- Define MySQL service in the failover domain.
- Add subservice for IP address failover to the MySQL service.
- Add subservice for LVM failover to the MySQL service.
- Add subservice for ext4 file system failover to the LVM failover subservice.
- Add subservice for MySQL database to the ext4 file system subservice.
- Restart the configured high-availability MySQL database service.
- Verify the proper IP address failover, activation of logical volume, file system mount, and the MySQL process restart.

## Appendix—Sample Configurations and Procedures

### Hardware, Software, and Firmware Used

The hardware, software, and firmware listed in [Table 1](#) were used in the configuration examples provided in the following sections of this guide.

**Table 1** Configuration Verification Components Used

| Component                 | Version                    | Model           |
|---------------------------|----------------------------|-----------------|
| Cisco ACE                 | A4(2.1a)                   | 4170            |
| Cisco ASA                 | 8.4(2)                     | 5580/5585       |
| Cisco Nexus               | 5.0(3) N2(1)               | 5548UP          |
| Cisco VIC (Palo) Firmware | 1.4(31)                    | M81KR           |
| Cisco UCS Blade           | S5500.1.4.3.0.061020111201 | B250 M2/B200 M2 |
| Cisco UCS Firmware        | 1.4(31)                    | 6120            |

**Table 1 Configuration Verification Components Used**

|                                                        |             |         |
|--------------------------------------------------------|-------------|---------|
| NetApp Data ONTAP®                                     | 8.0.1       |         |
| NetApp Storage Controller                              |             | FAS3270 |
| Red Hat Cluster Suite                                  | 6.1         |         |
| Red Hat Enterprise Linux                               | 6.1         |         |
| Updated Red Hat Enterprise Linux Packages <sup>1</sup> | (ccs, cman) |         |

1. Red Hat Enterprise Linux High-Availability Add-On supports cluster node fencing through Cisco UCS Manager starting with version 6.1. However, the support for fencing Cisco UCS service profiles created within suborganization hierarchy requires 6.1.z.

## Sample Boot LUN Configuration

Before Red Hat Enterprise Linux 6 can be installed, boot LUNs must be created for each blade. Each LUN should be properly zoned in the SAN fabric as well as mapped with an interface group (igroup) on the NetApp controller. Following is the igroup named iBack\_01 that contains the WWIDs of the Palo card.

```
dc22-netapp1*> igroup show iBack_01
iBack_01 (FCP) (ostype: linux):
    20:00:00:25:b5:26:a3:bf (logged in on: 1a)
    20:00:00:25:b5:26:b0:df (logged in on: 1b)
```

Following is the boot LUN itself

```
dc22-netapp1*> lun show /vol/fp_bootvol/back_01
/vol/fp_bootvol/back_01    40.0g (42953867264)    (r/w, online, mapped)
```

This example shows the specific mapping for the boot LUN as well as the igroup type of FCP:

```
dc22-netapp1*> lun show -m /vol/fp_bootvol/back_01
/vol/fp_bootvol/back_01    iBack_01    0    FCP
```

## Sample iSCSI Data LUN Configuration

To centralize the application data for easier backup, storage efficiency, and high-availability, create a separate LUN on the NetApp storage controller to be accessed with iSCSI.

Like the boot LUNs, the data LUNs must be mapped properly not only from host to LUN, but from igroup to LUN. In this example, the igroup “fp\_backend\_iscsi” is listed. Instead of FCP style WWIDs, iSCSI uses the initiator name generated by the software initiator on the host.

```
server_vfiler_1@dc22-netapp1*> igroup show fp_backend_iscsi
fp_backend_iscsi (iSCSI) (ostype: linux):
    iqn.1994-05.com.redhat:e486f696cfff (logged in on: CNA-VIF0-489)
    iqn.1994-05.com.redhat:2aed97dcb846 (logged in on: CNA-VIF0-489)
```

Here is the data LUN itself:

```
server_vfiler_1@dc22-netapp1*> lun show /vol/fp_lunvol/fp_lun
/vol/fp_lunvol/fp_lun    120.0g (128861601792)    (r/w, online, mapped)
```

Here is the igroup to data LUN mapping, where the igroup type is “iSCSI”:

```
server_vfiler_1@dc22-netapp1*> lun show -m /vol/fp_lunvol/fp_lun
/vol/fp_lunvol/fp_lun    fp_backend_iscsi    0    iSCSI
```

## Creating a UCSM Ethernet Adapter Policy for Red Hat Enterprise Linux 6

Create an Ethernet Adapter Policy for Red Hat Enterprise Linux 6 through the UCS Fabric Interconnect CLI:

```
dc26-6120-1-B# scope org /
dc26-6120-1-B /org # create eth-policy RedHat
dc26-6120-1-B /org/eth-policy* # set trans-queue count 1
dc26-6120-1-B /org/eth-policy* # set trans-queue ring-size 256
dc26-6120-1-B /org/eth-policy* # set recv-queue count 1
dc26-6120-1-B /org/eth-policy* # set recv-queue ring-size 512
dc26-6120-1-B /org/eth-policy* # set comp-queue count 2
dc26-6120-1-B /org/eth-policy* # set interrupt count 4
dc26-6120-1-B /org/eth-policy* # set offload tcp-tx-checksum enabled
dc26-6120-1-B /org/eth-policy* # set offload tcp-rx-checksum enabled
dc26-6120-1-B /org/eth-policy* # set offload tcp-segment enabled
dc26-6120-1-B /org/eth-policy* # set offload large-receive disabled
dc26-6120-1-B /org/eth-policy* # set rss receivesidescaling disabled
dc26-6120-1-B /org/eth-policy* # set failback timeout 5
dc26-6120-1-B /org/eth-policy* # set interrupt mode msi-x
dc26-6120-1-B /org/eth-policy* # commit-buffer
dc26-6120-1-B /org/eth-policy #
```

## Sample Nexus 5548 Zoning

The following is a zoning example for a host to connect through a vHBA to the NetApp appliance controller connection carrying a boot LUN.

### Step 1 Create a device-alias:

```
DC24-N5K-1# conf t
```

Enter configuration commands, one per line, and end with CNTL/Z.

```
DC24-N5K-1(config)# device-alias database
DC24-N5K-1(config-device-alias-db)# device-alias name fp-rhel-fronted-1a pwwn
20:00:00:25:b5:26:a2:9f
DC24-N5K-1(config-device-alias-db)# end
DC24-N5K-1# sh device-alias database | in fp-rhel-fronted-1a
device-alias name fp-rhel-fronted-1a pwwn 20:00:00:25:b5:26:a2:9f
DC24-N5K-1#
```

### Step 2 Create zone for host vHBA and controller pwwn:

```
DC24-N5K-1(config)# zone name fp-rhel-fronted-1 vsan 18
DC24-N5K-1(config-zone)# member device-alias fp-rhel-fronted-1a
DC24-N5K-1(config-zone)# member device-alias dc22-netapp1-1a
DC24-N5K-1(config-zone)# end
DC24-N5K-1# sh zone name fp-rhel-fronted-1
zone name fp-rhel-fronted-1 vsan 18
  pwwn 20:00:00:25:b5:26:a2:9f [fp-rhel-fronted-1a]
  pwwn 50:0a:09:81:9d:5f:27:e0 [dc22-netapp1-1a]
DC24-N5K-1#
```

### Step 3 Add zone to zoneset and activate:

```
DC24-N5K-1(config)# zoneset name dc24-n5k-1-vsan18 vsan 18
DC24-N5K-1(config-zoneset)# member fp-rhel-fronted-1
DC24-N5K-1(config-zoneset)# exit
DC24-N5K-1(config)# zoneset activate name dc24-n5k-1-vsan18 vsan 18
Zoneset activation initiated. check zone status
```

```
DC24-N5K-1(config)#
```

## Adding an IGMP Snooping Querier

Configure an IGMP Snooping Querier on a Cisco Nexus 5548 to allow Layer 2 multicast on a VLAN that does not have a configured Layer 3 VLAN interface:

```
DC25-N5K-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
DC25-N5K-1(config)# vlan 483
DC25-N5K-1(config-vlan)# ip igmp snooping querier 192.168.83.99
DC25-N5K-1(config-vlan)# end
DC25-N5K-1#
```

## Sample iSCSI Volume Setup

**Step 1** Install iSCSI client software on cluster nodes:

```
# yum -y groupinstall 'iSCSI Storage Client'
Loaded plugins: product-id, rhnplugin, subscription-manager
Updating Red Hat repositories.
Setting up Group Process
Resolving Dependencies
--> Running transaction check
---> Package iscsi-initiator-utils.x86_64 0:6.2.0.872-21.el6 will be installed
--> Finished Dependency Resolution
...
Complete!
```

**Step 2** Discover iSCSI targets from cluster nodes:

```
# iscsiadm --mode discoverydb --type sendtargets --portal 192.168.89.100 --discover
192.168.89.100:3260,2006
iqn.1992-08.com.netapp:sn.1573857230:vf.998c5ffc-a33e-11e0-a09a-00a09814476a
```

**Step 3** Login to the discovered iSCSI target:

```
# iscsiadm --mode node --targetname
iqn.1992-08.com.netapp:sn.1573857230:vf.998c5ffc-a33e-11e0-a09a-00a09814476a --portal
192.168.89.100 --login
Logging in to ...
... successful.
```

**Step 4** Verify the status of iSCSI sessions:

```
# service iscsi status
iSCSI Transport Class version 2.0-870
version 2.0-872
Target: iqn.1992-08.com.netapp:sn.1573857230:vf.998c5ffc-a33e-11e0-a09a-00a09814476a
Current Portal: 192.168.89.100:3260,2006
Persistent Portal: 192.168.89.100:3260,2006
*****
Interface:
*****
Iface Name: default
Iface Transport: tcp
Iface Initiatorname: iqn.1994-05.com.redhat:2aed97dcb846
```

```

Iface IPaddress: 192.168.89.151
Iface HWaddress: <empty>
Iface Netdev: <empty>
SID: 1
iSCSI Connection State: LOGGED IN
iSCSI Session State: LOGGED_IN
Internal iscsid Session State: NO CHANGE
*****
Negotiated iSCSI params:
*****
HeaderDigest: None
DataDigest: None
MaxRecvDataSegmentLength: 262144
MaxXmitDataSegmentLength: 65536
FirstBurstLength: 65536
MaxBurstLength: 65536
ImmediateData: Yes
InitialR2T: No
MaxOutstandingR2T: 1
*****
Attached SCSI devices:
*****
Host Number: 4State: running
scsi4 Channel 00 Id 0 Lun: 0
    Attached scsi disk sdeState: running

```

## Sample High-Availability Add-On Setup

**Step 1** Install the Red Hat Enterprise Linux 6 High-Availability Add-On software:

```

# yum -y groupinstall 'High Availability'
Loaded plugins: product-id, rhnplugin, subscription-manager
Updating Red Hat repositories.
Setting up Group Process
Resolving Dependencies
--> Running transaction check
---> Package ccs.x86_64 0:0.16.2-35.el6 will be installed
---> Package cman.x86_64 0:3.0.12-41.el6 will be installed
...
Complete!

```

**Step 2** Configure the firewall.

Start the firewall configuration utility `system-config-firewall-tui` or `system-config-firewall` and enable the ports required for Red Hat Cluster. The result should look like that the following example:

```

# cat /etc/sysconfig/iptables
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 5404 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 5405 -j ACCEPT

```

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 11111 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21064 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 16851 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

### Step 3 Disable ACPI on cluster nodes.

On all cluster nodes perform the following commands:

```
# chkconfig acpid off
# service acpid stop
Stopping acpi daemon: [ OK ]
```

### Step 4 Enable ricci on all cluster nodes.

Set the password for the ricci account and enable the service:

```
# passwd ricci
Changing password for user ricci.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
# chkconfig ricci on
# service ricci start
Starting oddjobd: [ OK ]
generating SSL certificates... done
Generating NSS database... done
Starting ricci: [ OK ]
```

### Step 5 Create the cluster configuration file for the two-node cluster with fencing through the Cisco UCS agent:

```
# ccs -f cluster.conf --createcluster fp-cluster
# ccs -f cluster.conf --addnode fp-backend-1-ic
Node fp-backend-1-ic added.
# ccs -f cluster.conf --addnode fp-backend-2-ic
Node fp-backend-2-ic added.
# ccs -f cluster.conf --addmethod UCS fp-backend-1-ic
Method UCS added to fp-backend-1-ic.
# ccs -f cluster.conf --addmethod UCS fp-backend-2-ic
Method UCS added to fp-backend-2-ic.
# ccs -f cluster.conf --addfencedev ucs agent=fence_cisco_ucs ipaddr=172.30.161.33
login=rhel-mgmt-be passwd=roZes123 suborg=/org-Dedicated/org-FlexPod-RHEL/org-Backend/
# ccs -f cluster.conf --addfenceinst ucs fp-backend-1-ic UCS port=FlexPod_RHEL_Backend-1
# ccs -f cluster.conf --addfenceinst ucs fp-backend-2-ic UCS port=FlexPod_RHEL_Backend-2
# ccs -f cluster.conf --setfencedaemon post_fail_delay=0 post_join_delay=30
# ccs -f cluster.conf --setcman two_node=1 expected_votes=1

# cat cluster.conf
<cluster config_version="10" name="fp-cluster">
  <fence_daemon post_fail_delay="0" post_join_delay="30"/>
  <clusternodes>
    <clusternode name="fp-backend-1-ic" nodeid="1">
      <fence>
        <method name="UCS">
          <device name="ucs" port="FlexPod_RHEL_Backend-1"/>
        </method>
      </fence>
    </clusternode>
    <clusternode name="fp-backend-2-ic" nodeid="2">
      <fence>
        <method name="UCS">
          <device name="ucs" port="FlexPod_RHEL_Backend-2"/>
        </method>
      </fence>
    </clusternode>
  </clusternodes>
</cluster>
```

```

        </method>
    </fence>
</clusternode>
</clusternodes>
<cman expected_votes="1" two_node="1"/>
<fencedevices>
    <fencedevice agent="fence_cisco_ucs" ipaddr="172.30.161.33" login="rhel-mgmt-be"
name="ucs" passwd="roZes123" suborg="/org-Dedicated/org-FlexPod-RHEL/org-Backend/" />
</fencedevices>
<rm>
    <failoverdomains/>
    <resources/>
</rm>
</cluster>

```

**Step 6** Synchronize configuration files and activate the cluster:

```

# cp cluster.conf /etc/cluster
# ccs -h fp-backend-1-ic --sync --activate
# ccs -h fp-backend-1-ic --startall

```

**Step 7** Verify that the cluster is in quorum:

```

# clustat
Cluster Status for fp-cluster @ Tue Aug  2 11:26:17 2011
Member Status: Quorate

Member Name                                ID    Status
-----
fp-backend-1-ic                            1 Online, Local
fp-backend-2-ic                            2 Online

# chkconfig cman on
# chkconfig rgmanager on

```

**Step 8** Verify node fencing:

```

fp-backend-1 # fence_node fp-backend-2-ic

```

**Step 9** Configure HA-LVM mode:

- Ensure that the parameter `locking_type` in the global sections of the `/etc/lvm/lvm.conf` file is set to the value `"1"`.
- Create the logical volume and file system using standard LVM2 and file system commands; for example:

```

# pvcreate /dev/sde
# vgcreate fp_vg /dev/sde
# lvcreate -L 120G -n fp_lv fp_vg
# mkfs -t ext4 /dev/fp_vg/fp_lv

```

- Edit the `volume_list` field in the `/etc/lvm/lvm.conf` file. Include the name of the root volume group and the cluster node's hostname as listed in the `/etc/cluster/cluster.conf` file and preceded by `"@"`. Note that this string must match the node name given in the `cluster.conf` file. Below is a sample entry from `/etc/lvm/lvm.conf` on `fp-backend-1-ic`:

```

volume_list = [ "VolGroup", @fp-backend-1-ic ]

```

This tag is used to activate shared volume groups or logical volumes. Do not include in the list the names of any volume groups that are to be shared using HA-LVM.

- Update the `initramfs` image on all cluster nodes. To do this on Red Hat Enterprise Linux 6:

```

# dracut -H -f /boot/initramfs-$(uname -r).img $(uname -r)

```

- e. Reboot all nodes to ensure the correct initramfs image is in use.

**Step 10** Add a failover domain:

```
# ccs -h fp-backend-1-ic --addfailoverdomain FD
```

**Step 11** Add nodes to the failover domain:

```
# ccs -h fp-backend-1-ic --addfailoverdomainnode FD fp-backend-1-ic
# ccs -h fp-backend-1-ic --addfailoverdomainnode FD fp-backend-2-ic
```

**Step 12** Add failover resources:

```
# ccs -h fp-backend-1-ic --addresource ip address=10.39.102.149/25 sleeptime=10
# ccs -h fp-backend-1-ic --addresource lvm name=mysql-lv vg_name=fp_vg lv_name=fp_lv
```

The shared logical volume is created on a NetApp LUN with thin provisioning. We are mounting ext4 file system on this volume with the discard option allowing NetApp to reclaim storage. This option may be not optimal for some workloads.

```
# ccs -h fp-backend-1-ic --addresource fs name=mysql-fs device=/dev/fp_vg/fp_lv
fstype=ext4 mountpoint=/var/lib/mysql options=discard
# ccs -h fp-backend-1-ic --addresource mysql name=mysql-db listen_address=10.39.102.149
startup_wait=10
```

**Step 13** Define MySQL resource group with subservices:

```
# ccs -h fp-backend-1-ic --addservice db-ha-service domain=FD autostart=1
recovery=relocate
# ccs -h fp-backend-1-ic --addsubservice db-ha-service ip ref=10.39.102.149/25
# ccs -h fp-backend-1-ic --addsubservice db-ha-service ip:lvm ref=mysql-lv
# ccs -h fp-backend-1-ic --addsubservice db-ha-service ip:lvm:fs ref=mysql-fs
# ccs -h fp-backend-1-ic --addsubservice db-ha-service ip:lvm:fs:mysql ref=mysql-db
# cat /etc/cluster/cluster.conf
<cluster config_version="21" name="fp-cluster">
  <fence_daemon post_fail_delay="0" post_join_delay="30"/>
  <clusternodes>
    <clusternode name="fp-backend-1-ic" nodeid="1">
      <fence>
        <method name="UCS">
          <device name="ucs" port="FlexPod_RHEL_Backend-1"/>
        </method>
      </fence>
    </clusternode>
    <clusternode name="fp-backend-2-ic" nodeid="2">
      <fence>
        <method name="UCS">
          <device name="ucs" port="FlexPod_RHEL_Backend-2"/>
        </method>
      </fence>
    </clusternode>
  </clusternodes>
  <cman expected_votes="1" two_node="1"/>
  <fencedevices>
    <fencedevice agent="fence_cisco_ucs" ipaddr="172.30.161.33" login="rhel-mgmt-be"
name="ucs" passwd="roZes123" suborg="/org-Dedicated/org-FlexPod-RHEL/org-Backend"/>
  </fencedevices>
  <rm>
    <failoverdomains>
      <failoverdomain name="FD" nofailback="0" ordered="0" restricted="0">
        <failoverdomainnode name="fp-backend-1-ic"/>
      </failoverdomain>
    </failoverdomains>
  </resources>
```

```

<ip address="10.32.102.149/25"/>
<lvm lv_name="fp_lv" name="mysql-lv" vg_name="fp_vg"/>
<fs device="/dev/fp_vg/fp_lv" fstype="ext4" mountpoint="/var/lib/mysql"
name="mysql-fs" options="discard"/>
<mysql listen_address="10.39.102.149" name="mysql-db" startup_wait="10"/>
</resources>
<service autostart="1" domain="FD" name="db-ha-service" recovery="relocate">
  <ip ref="10.39.102.149/25">
    <lvm ref="mysql-lv">
      <fs ref="mysql-fs">
        <mysql ref="mysql-db"/>
      </fs>
    </lvm>
  </ip>
</service>
</rm>
</cluster>

```

**Step 14** Start the high-availability MySQL database service:

```

# clusvcadm -e db-ha-service
# clustat
Cluster Status for fp-cluster @ Fri Aug 12 13:44:18 2011
Member Status: Quorate

Member Name                                ID    Status
-----
fp-backend-1-ic                            1 Online, Local, rgmanager
fp-backend-2-ic                            2 Online, rgmanager

Service Name                                Owner (Last)                                State
-----
service:db-ha-service                      fp-backend-2-ic                             started

```

**Step 15** Verify service relocation:

```

# clusvcadm -r db-ha-service
Trying to relocate service:db-ha-service...Success
service:db-ha-service is now running on fp-backend-1-ic

```

## References

- Red Hat Online Documentation: <http://docs.redhat.com>
- Red Hat Customer Portal: <https://access.redhat.com>
- Cisco Nexus 5548 Switch: <http://www.cisco.com/en/US/products/ps11215/index.html>
- Cisco Unified Computing System: <http://www.cisco.com/en/US/netsol/ns944/index.html>
- NetApp FAS3210 Storage Controller:  
[http://now.netapp.com/NOW/knowledge/docs/hardware/hardware\\_index.shtml#Storage%20appliances%20and%20V-series%20systems/gFilers](http://now.netapp.com/NOW/knowledge/docs/hardware/hardware_index.shtml#Storage%20appliances%20and%20V-series%20systems/gFilers)
- NetApp Support (formerly NetApp on the Web (NOW®)) site: <http://now.netapp.com>