Best Practices for Sharing an iSCSI SAN Infrastructure with Dell PS Series and Dell SC Series Storage using Linux Hosts

Dell Storage Engineering
July 2016

A Dell Reference Architecture

Revisions

Date	Description
July 2015	Initial release
December 2015	Updated for RHEL 7.2 and added dedicated host information
July 2016	Updated to include host configuration without PS Series HIT/Linux kit

Acknowledgements

Author: Steven Lemons

Table of contents

1 Introduction

Dell™ PS Series and SC Series storage systems both support storage area networks (SANs) over the iSCSI protocol. This document provides best practices for deploying:

- Linux host servers, specifically those running Red Hat® Enterprise Linux® (RHEL) 6 or 7, connected to an existing PS Series storage target to simultaneously connect to an SC Series storage target over a shared iSCSI SAN infrastructure (shared)
- Linux host servers, specifically those running RHEL 6 or 7, with both PS Series and SC Series storage targets, where only the iSCSI SAN infrastructure is shared; each host has a dedicated connection to either PS Series or SC Series storage targets

This paper also provides analysis of performance and high availability of the shared iSCSI SAN infrastructure consisting of PS Series and SC Series storage arrays.

The instructions included in this document cover both RHEL 6 and RHEL 7 environments. When necessary, specific Command Line Interface (CLI) differences between RHEL versions are addressed.

1.1 Scope

The scope of this paper focuses on the following:

- Dedicated switches for iSCSI storage traffic
- Non-DCB (Data Center Bridging) enabled iSCSI SAN
- Standard TCP/IP implementations utilizing standard network interface cards (NICs)
- Linux operating-system-provided software iSCSI initiator
- Virtual LAN (VLAN) untagged solution
- IPv4 only for PS Series and SC Series

The scope of this paper does **not** include:

- 1GbE or mixed-speed iSCSI SAN (combination of 1GbE and 10GbE)
- DCB or sharing the same SAN infrastructure for multiple traffic types
- iSCSI offload engine (iSOE)
- NIC partitioning (NPAR)
- VLAN tagging at the switch, initiator, or target
- SC Series storage systems using Fibre Channel over Ethernet (FCoE) SAN connectivity
- Non-MPIO (Multipath Input/Output) implementation

1.2 Audience

This paper is intended for storage administrators, network administrators, SAN system designers, storage consultants, or anyone tasked with configuring a SAN infrastructure for PS Series and SC Series storage. It is assumed that readers have experience in designing or administering a shared storage solution. There are assumptions made in terms of familiarity with all current Ethernet standards as defined by the Institute of Electrical and Electronic Engineers (IEEE), as well as TCP/IP standards defined by the Internet Engineering Task Force (IETF), FC standards defined by the T11 committee, and the International Committee for Information Technology Standards (INCITS).

1.3 Terminology

The following terms are used throughout this document:

Converged network adapter (CNA): A network adapter that supports convergence of simultaneous communication of both traditional Ethernet and TCP/IP protocols as well as storage networking protocols such as internet SCSI (iSCSI) or Fibre Channel over Ethernet (FCoE) using the same physical network interface port.

Data Center Bridging (DCB): A set of enhancements made to the IEEE 802.1 bridge specifications for supporting multiple protocols and applications in the same data center switching fabric. It is made up of several IEEE standards including Enhanced Transmission Selection (ETS), Priority-based Flow Control (PFC), Data Center Bridging Exchange (DCBX), and application Type-Length-Value (TLV). For more information, see the document, Data Center Bridging: Standards, Behavioral Requirements, and Configuration Guidelines with Dell EqualLogic iSCSI SANs.

Fault domain (FD): A set of hardware components that share a single point of failure. For controller-level redundancy, fault domains are created for SC Series storage to maintain connectivity in the event of a controller failure. In a dual-switch topology, each switch acts as a fault domain with a separate subnet and VLAN. Failure of any component in an FD will not impact the other FD.

Host Integration Tools (HIT): Dell PS Series Host Integration Tools include several easy-to-use host-based software modules that tightly integrate PS Series storage area networks (SANs) with hosts and applications, delivering advanced data protection, high availability and performance, and simplified management of application data and virtual machines for Microsoft and Linux environments.

iSCSI offload engine (iSOE): Technology that can free processor cores and memory resources to increase I/Os per second (IOPS) and reduce processor utilization.

NIC partitioning (NPAR): A technology used by Broadcom and QLogic which enables traffic on a network interface card (NIC) to be split into multiple partitions. NPAR is similar to QoS on the network layer and is usually implemented with 10GbE.

Link aggregation group (LAG): A group of Ethernet switch ports configured to act as a single high-bandwidth connection to another switch. Unlike a stack, each individual switch must still be administered separately and function independently.

Local area network (LAN): A network carrying traditional IP-based client communications.

Logical unit (LUN): A number identifying a logical device, usually a volume that is presented by an iSCSI or Fibre Channel storage controller.

Multipath I/O (MPIO): A host-based software layer that manages multiple paths for load balancing and redundancy in a storage environment.

Native VLAN and default VLAN: The default VLAN for a packet that is not tagged with a specific VLAN or has a VLAN ID of 0 or 1. When a VLAN is not specifically configured, the switch default VLAN will be utilized as the native VLAN.

Network interface card (NIC): A network interface card or network interface controller is an expansion board inserted into the computer/server so that the computer/server can connect to a network. Most NICs are designed for a particular type of network (typically Ethernet) protocol (typically TCP/IP) and media.

Storage area network (SAN): A Fibre Channel, Ethernet, or other specialized network infrastructure specifically designed to carry block-based traffic between one or more servers to one or more storage and storage inter-process communications systems.

Virtual LAN (VLAN): A method of virtualizing a LAN to make it appear as an isolated physical network. VLANs can reduce the size of and isolate broadcast domains. VLANs still share resources from the same physical switch and do not provide any additional Quality of Service (QoS) services such as minimum bandwidth, quality of a transmission, or guaranteed delivery.

VLAN tag: IEEE 802.1Q: The networking standard that supports VLANs on an Ethernet network. This standard defines a system of tagging for Ethernet frames and the accompanying procedures to be used by bridges and switches in handling such frames. Portions of the network which are VLAN-aware (IEEE 802.1Q conformant) can include VLAN tags. When a frame enters the VLAN-aware portion of the network, a tag is added to represent the VLAN membership of the frame's port or the port/protocol combination. Each frame must be distinguishable as being within exactly one VLAN. A frame in the VLAN-aware portion of the network that does not contain a VLAN tag is assumed to be flowing on the native (or default) VLAN.

Yellowdog Updater, Modified (YUM): Linux operating systems using the RPM Package Manager leverage the Yellowdog Updater, Modified (YUM) open-source command-line package management utility for automatic updates, package and dependency management, as well as RPM-based software installations.

2 Storage product overview

This section provides an overview of the Dell Storage products and technologies presented in this paper.

2.1 PS Series storage

PS Series arrays deliver the benefits of consolidated networked storage in a self-managing iSCSI SAN that is affordable and easy to use, regardless of scale. Built on an advanced, peer storage architecture, PS Series storage simplifies the deployment and administration of consolidated storage environments, enabling perpetual self-optimization with automated load balancing across PS Series members in a pool. This provides efficient scalability for both performance and capacity without forklift upgrades. PS Series storage provides a powerful, intelligent and simplified management interface.

2.2 SC Series storage

SC Series storage features multi-protocol support and self-optimizing, tiering capabilities. It can be configured with all flash, as a hybrid system, or with only traditional spinning disks and features automatic migration of data to the most cost-effective storage tier. Efficient thin provisioning and storage virtualization enable disk capacity usage only when data is written, enabling a pay-as-you-grow architecture. This self-optimizing system can reduce overhead cost and free up the administrator for other important tasks.

3 PS Series and SC Series iSCSI SAN coexistence

PS Series and SC Series storage arrays can coexist in a shared iSCSI SAN, either with shared hosts or dedicated hosts. Shared hosts not only share the iSCSI SAN infrastructure, but also connect to storage targets on both the PS Series and SC Series arrays. Shared-host coexistence (see Figure 1) shares the iSCSI SAN infrastructure and has all hosts connected to both array platforms. When hosts are dedicated (see Figure 2), each host in the iSCSI infrastructure connects to targets from either the PS Series array or SC Series array, but not both. Dedicated host coexistence utilizes a shared iSCSI SAN infrastructure only.

3.1 Topology of a shared iSCSI SAN infrastructure with shared hosts

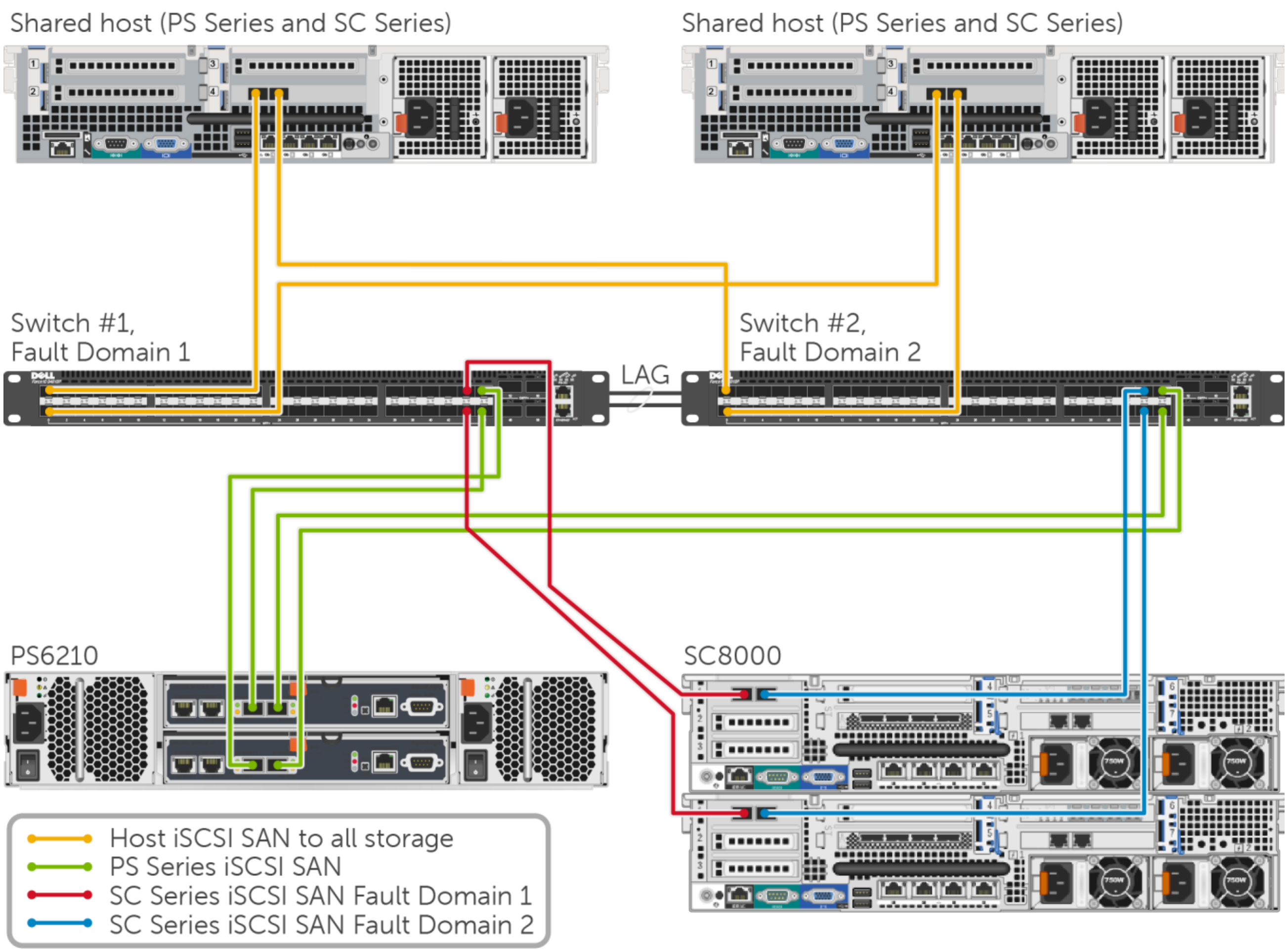

Figure 1 Shared iSCSI SAN with shared hosts reference topology

3.2 Topology of a shared iSCSI SAN infrastructure with dedicated hosts

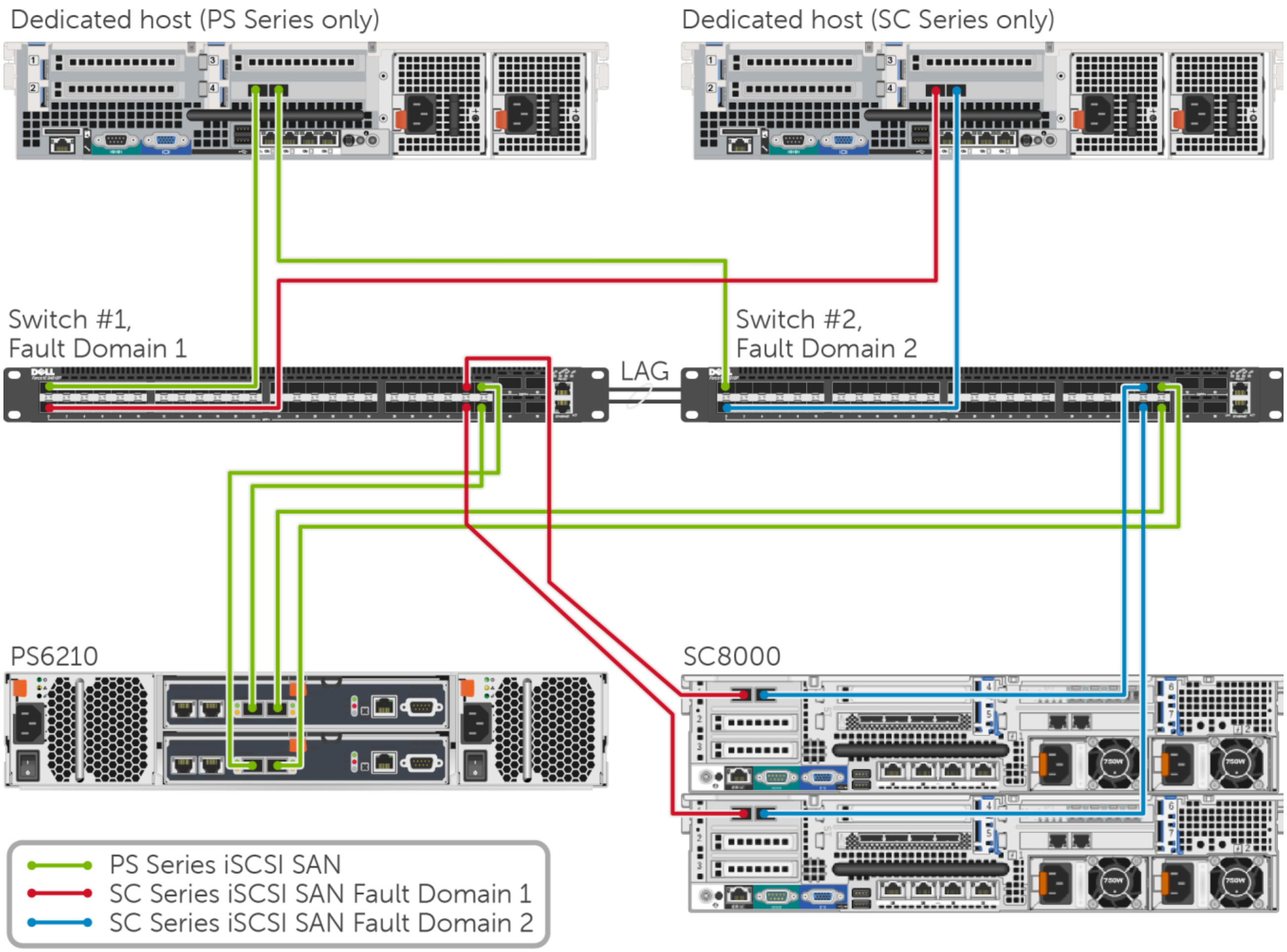

Figure 2 Shared iSCSI SAN with dedicated hosts reference topology

3.3 PS Series specific settings

The use case defined in this paper consists of SC Series storage arrays sharing the Linux host servers and Ethernet iSCSI SAN with deployed and operational PS Series storage array(s). It is recommended that the Ethernet network using iSCSI and Linux host servers accessing PS Series storage are configured using best practice recommendations as defined in the documents, *Red Hat Enterprise Linux 6 or 7 Host Configuration* and *Dell PS Series Configuration Guide*.

3.4 PS Series specific host settings without HIT/Linux

This paper recommends following the *Red Hat Enterprise Linux 6 or 7 Host Configuration* best practices for Linux host configuration, which uses the HIT/Linux kit for performance tuning and optimization when accessing PS Series storage. However, it is possible to apply the same performance tuning and optimization settings on the Linux host without the HIT/Linux kit. The following subsections highlight those settings within the Linux host that must be manually configured when not utilizing the HIT/Linux kit for PS Series access.

> **Note:** For more configuration information pertaining to Dell PS Series and Linux without the HIT/Linux kit, refer to Red Hat Enterprise Linux Configuration Guide for Dell Storage PS Series Arrays

3.4.1 Software requirements

Three software packages are needed for both RHEL 6 and RHEL 7 iSCSI connectivity to both PS Series and SC Series storage arrays. The following example shows the installation instructions for these three software packages on both RHEL 6 and RHEL 7.

```
[root@ ~]# yum install -y iscsi-initiator-utils device-mapper-multipath device-
mapper-multipath-libs
```

> **Note:** The `-y` flag is only needed if you want the yum program to answer all prompts with `yes` during the installation process.

3.4.2 MTU and Jumbo frames

The industry standard Ethernet frame size is 1500 bytes, however, to move more data more efficiently through your SAN, a larger frame size of 9000 bytes is recommended. This 9000-byte Ethernet frame size is referred to as Jumbo frames. This MTU sizing must be configured on and must match between the PS Series array's eth ports, each port in the Ethernet switch configuration, and any NIC/CNA port at the host. If the MTU size in all three components (storage array, Ethernet switch, and host) do not match, anomalous behavior can occur within the SAN.

This larger MTU size is configured for both RHEL 6 and RHEL 7 within each networking interface configuration file located at: **/etc/sysconfig/network-scripts/ifcfg-X** (**X** is the interface name). The variable within this configuration file that must be changed is **MTU**. To set this configuration value within the appropriate networking interface file, add the following using your preferred text editor:

```
MTU=9000
```

For example, the following command shows the Jumbo frames MTU size configured for both ports (p1 and p2) of the iSCSI NIC on a RHEL 6 or RHEL 7 host:

```
[root@ ~]# egrep MTU /etc/sysconfig/network-scripts/ifcfg-p1*

/etc/sysconfig/network-scripts/ifcfg-p1p1:MTU=9000

/etc/sysconfig/network-scripts/ifcfg-p1p2:MTU=9000
```

Once this configuration value has been added, a restart of the network service for both RHEL 6 and RHEL 7 must happen for the changes to take effect. This is accomplished through the following commands for each RHEL version:

RHEL 6:

```
[root@RHEL6 ~]# service network restart
```

RHEL 7:

```
[root@RHEL7 ~]# systemctl restart network
```

The following examples for each RHEL version verify the new MTU size configured for both ports of the iSCSI NIC:

RHEL 6:

```
[root@RHEL6 ~]# ifconfig | egrep -A 3 "p1p1|p1p2"

p1p1      Link encap:Ethernet  HWaddr 00:10:18:ED:54:00

          inet addr:10.10.10.108  Bcast:10.10.10.255  Mask:255.255.255.0

          inet6 addr: fe80::210:18ff:feed:5400/64 Scope:Link

          UP BROADCAST RUNNING MULTICAST  MTU:9000  Metric:1

--

p1p2      Link encap:Ethernet  HWaddr 00:10:18:ED:54:02

          inet addr:10.10.10.109  Bcast:10.10.10.255  Mask:255.255.255.0

          inet6 addr: fe80::210:18ff:feed:5402/64 Scope:Link

          UP BROADCAST RUNNING MULTICAST  MTU:9000  Metric:1
```

RHEL 7:

```
[root@RHEL7 ~]# ip addr show | egrep "p1p1|p1p2" | egrep -i mtu

6: p1p1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP qlen 1000

7: p1p2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP qlen 1000
```

3.4.3 Kernel configuration parameters

The following Linux kernel parameters need to be configured for both RHEL 6 and RHEL 7 to provide a starting point from which to achieve optimal performance for the SAN environment deployed within. While these settings are automatically added when using the **eqltune** utility included with the HIT/Linux kit, the following values can be manually added to either a RHEL 6 or RHEL 7 host through editing the **/etc/sysctl.conf** file and then reading those values into the running kernel through the **sysctl** command.

1. If using multiple iSCSI connections from the host to the PS Series array, the default Linux ARP behavior for these iSCSI connectors must be changed to prevent ARP resets (RST) from the iSCSI initiators to the target. Changing this ARP reset behavior allows more than a single iSCSI interface to receive/serve traffic. Add the following configuration variables to the **/etc/sysctl.conf** file for each iSCSI interface accessing the PS Series array (for this example, interface p1p1 and p1p2 are used).

   ```
   net/ipv4/conf/p1p1/arp_ignore = 1

   net/ipv4/conf/p1p1/arp_announce = 2

   net/piv4/conf/p1p1/rp_filter = 2

   net/ipv4/conf/p1p2/arp_ignore = 1

   net/ipv4/conf/p1p2/arp_announce = 2

   net/ipv4/conf/p1p2/rp_filter = 2
   ```

2. The network kernel settings need to be configured to allow for a larger receive/send socket buffer as well as increase the Linux autotuning TCP buffer limit to 16MB. For a RHEL 6 system, the following variables are added to the existing **/etc/sysctl.conf** file, while for a RHEL 7 system, these would be added to a new file titled **/etc/sysctl.d/dell_ps.conf**.

   ```
   net.core.rmem_max = 16777216

   net.core.wmem_max = 16777216

   net.ipv4.tcp_rmem = 8192 87380 16777216

   net.ipv4.tcp_wmem = 4096 65536 16777216

   net.core.wmem_default = 262144

   net.core.rmem_default = 262144
   ```

3. Once steps 1 and 2 have been completed, read these new values into the running kernel using the **sysctl -p** command.

   RHEL 6:

   ```
   [root@RHEL6 ~]# sysctl –p /etc/sysctl.conf
   ```

   RHEL 7:

   ```
   [root@RHEL7 ~]# sysctl –p {/etc/sysctl.conf,/etc/sysctl.d/dell_ps.conf}
   ```

3.4.4 iSCSI daemon parameters

The following iSCSI daemon parameters need to be configured for both RHEL 6 and RHEL 7 to provide a starting point from which to achieve optimal performance for the SAN environment deployed within. While these settings are automatically changed from their default values when using the HIT/Linux kit, the following values can be manually added to either a RHEL 6 or RHEL 7 host through editing the **/etc/iscsi/iscsid.conf** file and then restarting the iSCSI daemon.

```
node.session.cmds_max = 1024

node.session.queue_depth = 128

node.session.iscsi.FastAbort = No
```

Next, restart the iSCSI daemon with the following commands for each RHEL version.

RHEL 6:

```
[root@RHEL6 ~]# /etc/init.d/iscsid restart
```

RHEL 7:

```
[root@RHEL7 ~]# systemctl restart iscsid
```

> **Note:** If there are active iSCSI connections on the host, prior to restarting the iSCSI daemon for the new configuration values to take effect, those iSCSI targets must be logged out of with the **iscsiadm -n node -u** command.

3.4.5 Adding multiple iSCSI interfaces

Each iSCSI interface that accesses the PS Series array must be added to the iscsiadm interface database and it must have a defined alias name for use when binding with a software iSCSI session so the iSCSI daemon is aware of each interface and can be available for multipath use. In the following example, two iSCSI interfaces (p1p1 and p1p2) are added to the iscsiadm interface database and their network interface names are updated. This command is the same for both RHEL 6 and RHEL 7.

```
[root@RHEL7 ~]# iscsiadm –m iface –I p1p1 --op=new

[root@RHEL7 ~]# iscsiadm –m iface –I p1p2 --op=new

[root@RHEL7 ~]# iscsiadm –m iface –I p1p1 --op=update –n iface.net_ifacename –v
p1p1

[root@RHEL7 ~]# iscsiadm –m iface –I p1p2 --op=update –n iface.net_ifacename –v
p1p2
```

Now, these two iSCSI interfaces are validated as present within the iscsiadm interface database (highlighted in yellow), and their network interface alias names (highlighted in green) are verified as updated through the following command, which displays the list of configured iSCSI interfaces from the **/var/lib/iscsi/ifaces** directory.

```
[root@RHEL7 ~]# iscsiadm -m iface -o show | egrep "p1p1|p1p2"

p1p1 tcp,<empty>,<empty>,p1p1,<empty>

p1p2 tcp,<empty>,<empty>,p1p2,<empty>
```

3.4.6 Discovering iSCSI PS targets

The iSCSI daemon can now discover PS Series volumes, referred to as targets when using the iscsiadm command, using the two configured iSCSI interfaces. In the following example, the iscsiadm command queries the PS Series group located at 10.10.10.10 using both iSCSI interfaces (p1p1 and p1p2), reporting back the available volumes for the host. This command is the same for both RHEL 6 and RHEL 7.

```
[root@RHEL7 ~]#  iscsiadm -m discoverydb -t sendtargets -p 10.10.10.10 -o new -D

10.10.10.10:3260,1 iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-
938d6f1b16457683-rhel7-bcm-01

10.10.10.10:3260,1 iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-
f8bd6f1b16757683-rhel7-bcm-02

10.10.10.10:3260,1 iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-
938d6f1b16457683-rhel7-bcm-01

10.10.10.10:3260,1 iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-
f8bd6f1b16757683-rhel7-bcm-02
```

Two volumes were discovered in this example (RHEL7-BCM-01 and RHEL7-BCM-02) and each iSCSI interface reported back seeing both volumes for a total of four targets displayed.

3.4.7 Accessing iSCSI PS volumes

To access the PS Series volumes, the host must log into previously discovered targets with the iscsiadm command. All of the discovered targets within the iscsiadm discovery database can be logged into at the same time with the **iscsiadm -m node -l** command. In the following example, the host logs into all targets within the iscsiadm discovery database from each configured iSCSI interface. This command is the same for both RHEL 6 and RHEL 7.

```
[root@RHEL7 ~]# iscsiadm -m node -l

Login to [iface: p1p1, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-
938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260] successful.

Login to [iface: p1p2, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-
938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260] successful.
```

```
Login to [iface: p1p1, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260] successful.

Login to [iface: p1p2, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260] successful.
```

When necessary, the host must log out of each discovered target from the PS Series array. The logout process is executed with the **iscsiadm -m node -u** command which logs out of every target defined within its discovered target database. This command is the same for both RHEL 6 and RHEL 7.

```
[root@RHEL7 ~]# iscsiadm –m node –u

Logging out of session [sid: 182, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260]

Logging out of session [sid: 183, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260]

Logging out of session [sid: 184, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260]

Logging out of session [sid: 185, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260]

Logout of [sid: 182, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260] successful.

Logout of [sid: 183, target: iqn.2001-05.com.equallogic:0-af1ff6-122e5d4db-938d6f1b16457683-rhel7-bcm-01, portal: 10.10.10.10,3260] successful.

Logout of [sid: 184, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260] successful.

Logout of [sid: 185, target: iqn.2001-05.com.equallogic:0-af1ff6-13de5d4db-f8bd6f1b16757683-rhel7-bcm-02, portal: 10.10.10.10,3260] successful.
```

3.4.8 Multipath configuration

When using the HIT/Linux kit, the installed EqualLogic Host Connection Manager (ehcmd) service uses the host software iSCSI initiator to connect to the discovered volumes. Without the HIT/Linux kit, the multipathd service is needed to allow for multiple I/O paths (referred to as MPIO) between the host and storage array.

The instructions within section 4.4, step 8, demonstrate the process of obtaining the WWID of the newly discovered volumes through the iscsiadm command. These instructions also include the recommended host configuration defaults to access both PS and SC storage using the multipathd service for optimal MPIO within the SAN.

3.5 SC Series specific settings

A typical SC Series iSCSI implementation involves two separate, dedicated Ethernet fabrics as two fault domains with an independent IP subnet and unique, non-default VLANs in each switch fabric. However, to enable SC Series to coexist with PS Series and share the Ethernet SAN infrastructure using the iSCSI storage protocol, use a single subnet for all host and storage ports.

To implement this correctly, a basic understanding of PS Series and SC Series storage is needed. This paper provides an overview of both storage types.

Each PS Series volume is presented as a unique target with LUN 0. The PS Series volumes that are accessible to the host are listed in the iSCSI initiator properties. When a volume is connected, the iSCSI initiator establishes the initial iSCSI session and then the PS Series MPIO plug-in determines if additional sessions are necessary for redundancy.

Each SC Series array has both front-end and back-end ports. The front-end ports are presented with unique target LUN IDs. Every initiator IP has a connection to each port that it can access. Redundant connections are made by creating multiple sessions with each of the virtual iSCSI ports of the SC Series storage system. For example, one initiator port and two target ports in each fault domain means there will be four connections (two for each fault domain).

Note that the host port on one fault domain can access the target port on the other fault domain through the switch interconnection. Ensure that the host ports are connected to the appropriate fault domain and target port, physically and in iSCSI sessions. This minimizes inter-switch link (ISL) traffic and ensures that at least some iSCSI sessions persist in the event that a component fails in a fault domain. The following sections discuss ways to ensure that the physical connectivity and iSCSI sessions are established correctly.

3.5.1 SC Series host physical connectivity and IP assignment

Depending on the OS-specific implementation, different methods are used to connect the arrays and assign IP addresses. Since SC Series fault domains are connected by an ISL and are in a single IP subnet, it is important to ensure that iSCSI sessions are properly established within their fault domains. Host ports connected to Fault Domain 1 should connect to switch fabric and storage ports on Fault Domain 1 physically. The same rule applies for Fault Domain 2. This step is important because, with a single subnet, it is possible for the hosts to access SC Series storage ports on both fault domains. The correct connectivity minimizes ISL traffic and ensures that at least some iSCSI sessions will persist in the event of a component failure.

Figure 3 depicts proper connection from each host port to the SC Series storage ports within the same fault domain without traversing the switch interconnection.

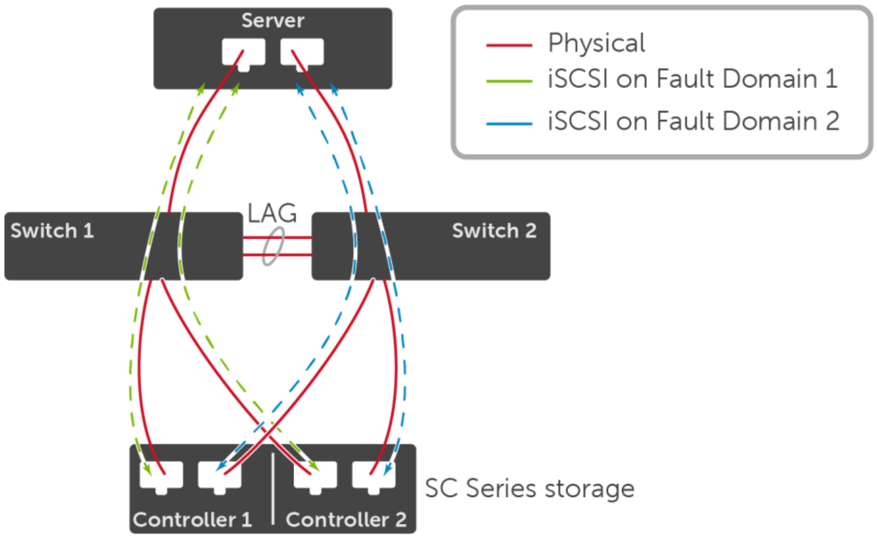

Figure 3 Connecting the host to SC Series ports

Note: With the approach discussed in this paper, misconfiguration of the SC Series connectivity (for example, host ports not connected to the correct fault domain) can lead to loss of volume access in the event of a switch failure.

4 Enabling RHEL 7.0 host access to PS Series and SC Series iSCSI storage

When enabling the Linux host for shared access to both PS Series and SC Series iSCSI storage arrays, the following Linux tools as well as configuration instructions are recommended.

4.1 Useful Linux tools

Identifying which SC Series volume correlates to a specific Linux device can be challenging, but the following tools can help simplify this process. Many of these tools are included in the base Linux install, while others only require installation through the YUM command.

4.1.1 lsscsi

lsscsi is a tool that parses information from the /proc and /sys pseudo-file systems into a simple human-readable output. Although it is not currently included in the base installs for RHEL 6 or RHEL 7, it is in the base repository and can be easily installed using YUM.

1. Use the following command to check for the lsscsi package.

   ```
   [root@RHEL7 ~]# rpm -qa | egrep -i lsscsi
   ```

2. If no results are returned, the lsscsi package is not installed. Use the following command to install the lsscsi package:

   ```
   [root@RHEL7 ~]# yum install lsscsi -y
   ```

The following output is a sample return from the lsscsi command on RHEL 7 showing both SC and PS Series volumes.

The first column shows the designation for the volume with the format: `[host:channel:target:lun]`. Table 1 describes each element within the brackets from the lssscsi command.

```
[root@RHEL7 ~]# lsscsi
[11:0:0:1]    disk    COMPELNT Compellent Vol    0605   /dev/sdb
[11:0:0:2]    disk    COMPELNT Compellent Vol    0605   /dev/sdc
[11:0:0:3]    disk    COMPELNT Compellent Vol    0605   /dev/sdd
[11:0:0:4]    disk    COMPELNT Compellent Vol    0605   /dev/sde
[136:0:0:0]   disk    EQLOGIC  100E-00           8.0    /dev/sdcd
[138:0:0:0]   disk    EQLOGIC  100E-00           8.0    /dev/sdah
[139:0:0:0]   disk    EQLOGIC  100E-00           8.0    /dev/sdai
[140:0:0:0]   disk    EQLOGIC  100E-00           8.0    /dev/sdaj
```

Table 1 iSCSI volume designation

Designation	Description
scsi_host	The local HBA host that the volume is mapped to
channel	The iSCSI bus address, which is always zero (0)
target	The SC Series front-end ports (targets)
lun	The LUN that the volume is mapped on

4.1.2 scsi_id

scsi_id is a utility that retrieves and generates a unique SCSI identifier, commonly referred to as the WWID of the volume. It is available in all base Linux installations. This WWID can be used to correctly match the volume serial number reported in the SC Series GUI for accurate correlation when mapping volumes within the Linux operating system.

```
[root@RHEL7 ~]# /lib/udev/scsi_id -g -u /dev/sde

36000d31000edef000000000000000080
```

Note: In RHEL 7, the absolute path location of the scsi_id utility is /lib/udev/scsi_id. In RHEL 6, it can be located at /sbin/scsi_id.

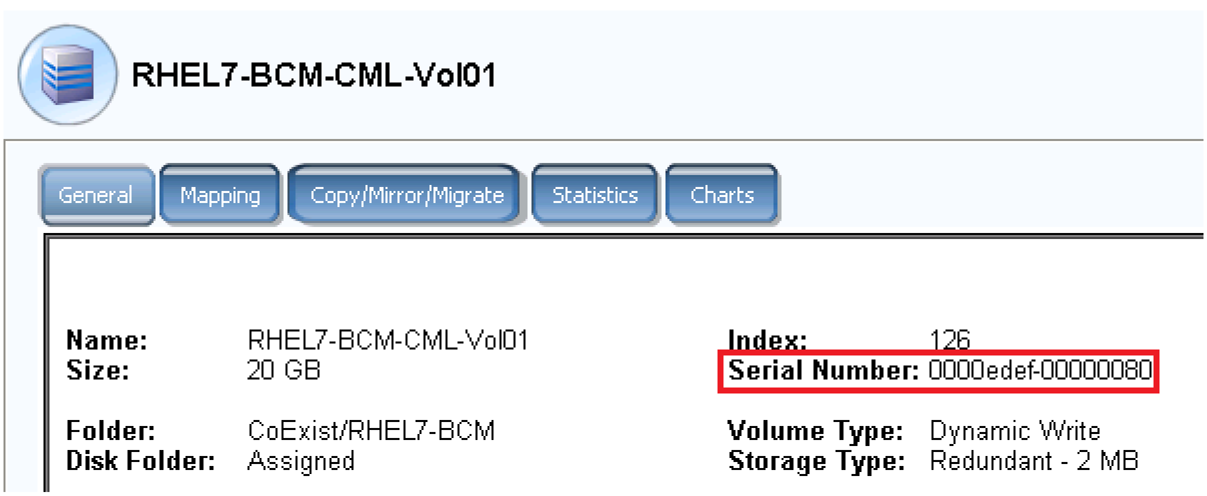

Figure 4 Example SC Series LUN serial number correlation

4.2 Host Integration Tools (HIT kit) for RHEL

The Linux servers accessing PS Series storage are configured using best practice recommendations as defined in the documents, *Red Hat Enterprise Linux 6 or 7 Host Configuration* and *PS Series Configuration Guide*.

4.3 RHEL native multipathing

RHEL provides native multipathing through the device-mapper-multipath package that can be used with any storage platform if a vendor-provided multipath solution is not available.

4.4 Configuring the RHEL iSCSI software initiator to access volumes

Use the following steps to query and establish iSCSI sessions from a Linux host to access SC Series volumes. These steps are also used when accessing PS Series volumes without the HIT/Linux kit.

After the Linux host has established iSCSI connections to the PS Series array(s) (section 3.4.7), the next steps are to confirm connectivity to either the SC Series or PS Series array(s) as well as obtain a list of available targets being presented from the queried storage array(s).

With HIT/Linux kit:

Obtain the iSCSI NIC name(s) through the HIT/Linux kit `ehcmcli` utility (located at **/usr/sbin/ehcmcli**) previously installed during HIT kit setup.

```
[root@RHEL7 ~]# ehcmcli status

Generating diagnostic data, please wait...


=========================================================

Adapter List

=========================================================

  Name: p1p1

    IP Address: 10.10.10.110

    HW addr: 00:10:18:EF:81:B0


  Name: p1p2

    IP Address: 10.10.10.111

    HW addr: 00:10:18:EF:81:B2
```

Without HIT/Linux kit:

Display the iSCSI NIC name(s) that were added to the iscsiadm interface database from section 3.4.5 through the following command, which displays the list of configured iSCSI interfaces from the **/var/lib/iscsi/ifaces** directory.

```
[root@RHEL7 ~]# iscsiadm –m iface –o show | egrep “p1p1|p1p2”
p1p1 tcp,<empty>,<empty>,p1p1,<empty>
p1p2 tcp,<empty>,<empty>,p1p2,<empty>
```

1. Confirm connectivity to the SC Series group IPs by using the ping utility, specifying each port of the iSCSI NICs individually.

```
[root@RHEL7 ~]# ping -I p1p1 10.10.10.35
64 bytes from 10.10.10.35: icmp_seq=1 ttl=64 time=0.139 ms
64 bytes from 10.10.10.35: icmp_seq=2 ttl=64 time=0.073 ms

[root@RHEL7 ~]# ping -I p1p1 10.10.10.36
64 bytes from 10.10.10.36: icmp_seq=1 ttl=64 time=0.094 ms
64 bytes from 10.10.10.36: icmp_seq=2 ttl=64 time=0.095 ms

[root@RHEL7 ~]# ping -I p1p2 10.10.10.35
64 bytes from 10.10.10.35: icmp_seq=1 ttl=64 time=0.094 ms
64 bytes from 10.10.10.35: icmp_seq=2 ttl=64 time=0.095 ms

[root@RHEL7 ~]# ping -I p1p2 10.10.10.36
64 bytes from 10.10.10.36: icmp_seq=1 ttl=64 time=0.095 ms
64 bytes from 10.10.10.36: icmp_seq=2 ttl=64 time=0.094 ms
```

2. Show the contents of the `iscsiadm` database confirming only the PS Series array(s) exist.

```
[root@RHEL7 ~]# iscsiadm -m discoverydb -o show
10.10.10.10:3260 via sendtargets
```

3. Query each SC Series group IP to populate the `iscsiadm` utilities database with available targets from the presented SC Series controllers.

```
[root@RHEL7 ~]# iscsiadm -m discoverydb -p 10.10.10.35 -t sendtargets -o
new -D

10.10.10.35:3260,0 iqn.2002-03.com.compellent:5000d31000edef2f

10.10.10.35:3260,0 iqn.2002-03.com.compellent:5000d31000edef31

10.10.10.35:3260,0 iqn.2002-03.com.compellent:5000d31000edef2f

10.10.10.35:3260,0 iqn.2002-03.com.compellent:5000d31000edef31


[root@RHEL7 ~]# iscsiadm -m discoverydb -p 10.10.10.36 -t sendtargets -o
new -D

10.10.10.36:3260,0 iqn.2002-03.com.compellent:5000d31000edef2e

10.10.10.36:3260,0 iqn.2002-03.com.compellent:5000d31000edef30

10.10.10.36:3260,0 iqn.2002-03.com.compellent:5000d31000edef2e

10.10.10.36:3260,0 iqn.2002-03.com.compellent:5000d31000edef30
```

4. Show the contents of the updated iscsiadm database to confirm addition of the queried SC Series group IPs.

```
[root@RHEL7 ~]# iscsiadm -m discoverydb -o show

10.10.10.10:3260 via sendtargets

10.10.10.35:3260 via sendtargets

10.10.10.36:3260 via sendtargets
```

5. Using the `iscsiadm` utility, log out of all existing iSCSI connections.

```
[root@RHEL7 ~]# iscsiadm -m node –u
```

Note: Logging out of all existing iSCSI connections is a considered a **production disruption** (regardless of volume status being production or non-production) and should be scheduled during non-critical business hours because this command will close any/all connections to the PS Series array(s).

6. Using the `iscsiadm` utility, confirm there are no active sessions.

```
[root@RHEL7 ~]# iscsiadm -m session

iscsiadm: No active sessions.
```

7. Using the `iscsiadm` utility, log into all targets within the discovery database.

```
[root@RHEL7 ~]# iscsiadm -m node –l
```

8. Obtain the Linux device name of the newly presented SC Series volume.

```
[root@RHEL7 ~]# lsscsi
[231:0:0:4]  disk    COMPELNT Compellent Vol   0605  /dev/sdb
[232:0:0:4]  disk    COMPELNT Compellent Vol   0605  /dev/sdc
[235:0:0:4]  disk    COMPELNT Compellent Vol   0605  /dev/sdd
[236:0:0:4]  disk    COMPELNT Compellent Vol   0605  /dev/sde
```

9. Obtain the WWID of each device name (these should all be the same because there are two connections per controller to the SC Series).

```
[root@RHEL7 ~]# /lib/udev/scsi_id -g -u /dev/sdb
36000d31000edef000000000000000080
[root@RHEL7 ~]# /lib/udev/scsi_id -g -u /dev/sdc
36000d31000edef000000000000000080
[root@RHEL7 ~]# /lib/udev/scsi_id -g -u /dev/sdd
36000d31000edef000000000000000080
[root@RHEL7 ~]# /lib/udev/scsi_id -g -u /dev/sde
36000d31000edef000000000000000080
```

The following one-line command leverages the `lsscsi` and `scsi_id` commands and can be used to get both the Linux device name and associated WWID for easier reference.

```
[root@RHEL7 ~]# for x in `lsscsi|egrep -i compelnt|awk -F' ' '{print
$7}'`;do echo -n "$x:";/lib/udev/scsi_id -g -u $x;done|sort -t":" -k2
/dev/sdb:36000d31000edef000000000000000080
/dev/sdc:36000d31000edef000000000000000080
/dev/sdd:36000d31000edef000000000000000080
/dev/sde:36000d31000edef000000000000000080
```

10. Now that the WWIDs of the SC Series volume are identified, the next step is to install and configure `multipathd` on the Linux host.

    a. Use the following command to check for installation of the device-mapper-multipath package.

    ```
    [root@RHEL7 ~]# rpm –qa | egrep –i device-mapper-multipath
    ```

    b. If no results are returned, the device-mapper-multipath package is not installed. Use the following command to install it.

    ```
    [root@RHEL7 ~]# yum install device-mapper-multipath –y
    ```

    c. Create the **/etc/multipath.conf** file, start the `multipathd` service, and set the service to start with reboot.

    ```
    [root@RHEL7 ~]# mpathconf --enable --with_multipathd y
    ```

    d. Edit **/etc/multipath.conf** to handle the previously found WWIDs from the SC Series. The resulting configuration should be identical to the following sample.

    ```
    blacklist_exceptions {

            # PS Array

            wwid "360fff1ba4d5d2e12837645161b6f8d93"       #PS_Vol01

            # SC Array

            wwid "36000d31000edef000000000000000080"       #SC_Vol01

    }


    defaults {

            find_multipaths yes

            user_friendly_names yes

            polling_interval 5

            path_selector   "round-robin 0"

            path_grouping_policy    multibus

            path_checker    tur

            no_path_retry    queue

    }
    ```

```
multipaths {
        multipath {
                wwid    360fff1ba4d5d2e12837645161b6f8d93
                alias   PS_Vol01
        }
        multipath {
                wwid    36000d31000edef0000000000000000080
                alias   SC_Vol01
        }
}
```

e. Restart the `multipathd` service for configuration changes to take effect.
RHEL 7:

```
[root@RHEL7 ~]# systemctl restart multipathd
```

RHEL 6:

```
[root@RHEL6 ~]# service multipathd restart
```

f. Check the status of the `multipathd` service to confirm it is running and enabled to start with reboot. Look for `enabled` or `disabled` (highlighted in the following samples) on the line beginning with `Loaded` to confirm service start on reboot.
RHEL 7:

```
[root@RHEL7 ~]# systemctl status multipathd
multipathd.service - Device-Mapper Multipath Device Controller
   Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled)
   Active: active (running) since Tue 2015-12-01 16:05:04 CST; 56s ago
```

RHEL 6:

```
[root@RHEL6 ~]# service multipathd status;/sbin/chkconfig --list
multipathd
multipathd (pid  2628) is running...
multipathd      0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

11. With the `multipathd` service configured and running, confirm local mapping of the SC Series volume for further local file system creation.

```
[root@RHEL7 ~]# ls -al /dev/mapper|egrep CMLVol01

lrwxrwxrwx  1 root root       7 Dec  1 16:05 CMLVol01 -> ../dm-5
```

4.5 Shared host connections to both storage platforms (PS Series and SC Series)

In a shared environment where Linux hosts connect to both PS Series and SC Series storage targets, the multipath and iSCSI settings optimized by the HIT kit (described previously in section 4.4) either aligned or did not interfere with SC Series storage best practices.

It is recommended to run the `eqltune` utility (which is installed with the HIT/Linux kit, located at **/usr/sbin/eqltune** on both RHEL 6 and RHEL 7) after completing shared configurations to see any suggestions and warnings of the configurations for sysctl tunables, Ethernet devices, and iSCSI settings. By default, the `eqltune` utility, when executed, runs in check mode and displays a summary of all settings along with any suggestions or warnings in current configuration. After reviewing these suggestions and warnings, if proposed changes are acceptable, run `eqltune fix` to have the utility make those changes for you.

Note: Some of the suggestions concerning iSCSI settings must be manually edited in /etc/iscsi/iscsid.conf with a restart of the `iscsid` service

```
[root@RHEL7 ~]# eqltune fix

Dell EqualLogic 'eqltune' version 1.4.0 Build 416402

Copyright (c) 2010-2014 Dell Inc.


Repairing all critical system issues...


Block Devices

-------------

No issues to fix


Sysctl Tunables

---------------

No issues to fix
```

```
Ethernet Devices
----------------
No issues to fix

iSCSI Settings
--------------
No issues to fix

External Utility Settings
-------------------------
No issues to fix

EqualLogic Host Tools
---------------------
No issues to fix

Notice
------
eqltune previously backed up the following 3 system configuration files:
  * /var/lib/equallogic/eqltune.backup/sysctl.conf
  * /var/lib/equallogic/eqltune.backup/iscsid.conf
  * /var/lib/equallogic/eqltune.backup/lvm.conf
```

4.6 Dedicated host connections to each storage platform (PS Series or SC Series)

When Linux hosts are dedicated to either the PS Series or SC Series storage platform, the native multipathing must be used when connecting to SC Series storage.

To configure Linux hosts for dedicated connections to PS Series, use the best practices defined in the documents, *Red Hat Enterprise Linux 6 or 7 Host Configuration* and *PS Series Configuration Guide* referenced in section 4.2.

To configure Linux hosts for dedicated connections to SC Series, the `iscsi-initiator-utils` package will need to be installed since the HIT kit was not installed.

```
[root@RHEL7 ~]# yum install iscsi-initiator-utils -y
```

Once the `iscsi-initiator-utils` package is installed, complete remaining configuration steps by referencing the recommendations and tools in sections 4.1.1, 4.1.2, and 4.4.

SC Series host settings not detailed in this paper are provided in the document, *Dell Storage Center with Red Hat Enterprise Linux (RHEL) 6x Best Practices*. Refer to appendix A.2 for a list of references to SC Series array best practices.

5 Test methodology

Note: The test methodology used in this paper is only valid in the presented topology, components, and test cases. The results may not be directly applicable to environments with other components or variables.

Key testing factors discussed in this section include:

- The test cases assess the impact from the operational and performance perspective of the shared host, host NICs, and shared Ethernet SAN infrastructure using the iSCSI protocol.
- The main objective was to ensure that the Linux hosts with both shared and dedicated NICs and shared Ethernet switches would sustain I/O workloads on PS Series storage when SC Series storage is added.
- SAN component failures were induced to confirm that the solution is highly available, and that a failure on one type of storage did not impact the other.
- The tests focused on validating the capability of the hosts and switches to handle both PS Series and SC Series storage with respect to network adapters on the host, switch hardware, buffer settings on the switch, Ethernet flow control, and resiliency of the storage network.

A baseline test running a specific I/O workload was performed with PS Series storage in a standalone environment. The standalone environment consisted of the Linux hosts accessing only PS Series storage arrays. The test was then repeated in both dedicated and shared coexistence environments (including PS Series and SC Series storage) with both storage types sharing the Ethernet iSCSI SAN to ensure the data could be compared directly without introducing variables beyond those specified in the test plan. A high availability test for link failures, switch failure, and other factors was carried out to prove solution robustness.

In order to determine the relative performance of each SAN design, the performance tool vdbench was used to capture throughput values at four distinct I/O workloads:

- 8 KB transfer size, random I/O, 70% read
- 64 KB transfer size, random I/O, 70% read
- 256 KB transfer size, sequential I/O, 100% read
- 256 KB transfer size, sequential I/O, 100% write

These I/O workloads were first run for one hour each in a series with Linux host servers accessing PS Series volumes only in the iSCSI SAN network to establish a baseline. Next, the same I/O workloads were run for one hour each with each of the Linux hosts accessing either PS Series or SC Series volumes sharing the same iSCSI SAN network. Finally, the same I/O workloads were run for one hour each with both of the Linux hosts accessing both PS Series and SC Series volumes using the same host NICs with both storage platforms sharing the same iSCSI SAN network. The results for both shared and dedicated tests were compared against the baseline for any change in performance.

IOPS, latency measured in milliseconds (ms), throughput measured in megabytes per second (MBPS), retransmits (an indicator of network congestion), and other statistics (such as switch packet drops, discards, and pause frames) were collected from host, switch, and storage ports. These parameters were carefully analyzed to ensure that the host and iSCSI SAN network performance were not adversely impacted with both PS Series and SC Series storage being accessed simultaneously.

5.1 Test environment

The test environment consisted of the following:

- Two Dell PowerEdge™ R620 servers running RHEL 7.1
- Three Dell PS6210 storage arrays (72 x 15K disks)
- Two Dell SC8000 storage controllers with three Dell SC220 enclosures (72 x 15K disks)

Each PowerEdge R620 server had RHEL 7 installed locally and native Linux iSCSI software initiators were used to connect to PS Series and SC Series volumes. These volumes were mapped to the Linux host as raw LUNs. Each Linux host had 8 LUNs (volumes) from PS Series storage arrays and 8 LUNs (volumes) from SC Series storage accessible to itself. For the baseline test in a standalone environment, the Linux hosts were connected to volumes on PS Series storage arrays only. In a shared environment, the Linux hosts were connected to volumes on PS Series and SC Series storage arrays to test the addition of SC Series storage to an existing PS Series SAN environment.

5.2 I/O performance testing

Testing scenarios for this paper are as follows:

- Baseline test: The Linux hosts were connected to volumes on PS Series storage arrays only.
- Shared coexistence environment test: Both Linux hosts were connected to both PS Series and SC Series storage arrays to show the effects of adding SC Series storage to an existing PS Series SAN environment, which includes hosts accessing both storage platforms.
- Dedicated coexistence environment test: One Linux host was connected to volumes on PS Series storage only, and the other Linux host was connected to volumes on the SC Series storage only. This test shows the effects of adding SC Series storage traffic to the existing PS Series iSCSI SAN infrastructure only.

5.2.1 I/O performance results and analysis: shared hosts

Performance analysis in the shared-host environment is designed to show the impact to the PS Series storage environment when SC Series storage is added to the existing environment and existing hosts access storage from both the PS Series and SC Series storage platforms over the same iSCSI adapters and iSCSI infrastructure.

The following figures show the baseline IOPS and latency information gathered on the PS Series environment and how it compares after the introduction of the SC Series storage to the environment. Figure 5 and Figure 6 show the baseline information on the left and the shared environment data on the right, as a percentage of the baseline. As shown in the charts, there was no significant decrease in IOPS nor increase in latency with the introduction of the SC Series storage platform to the environment.

Figure 5 Shared host performance data: 8K and 64K

Figure 6 Shared host performance data: 256K read and write

5.2.2 I/O performance results and analysis: dedicated hosts

Performance analysis in the dedicated host environment is designed to show the impact to the PS Series storage environment when a dedicated SC Series storage environment is added to the same iSCSI infrastructure.

Figure 7 shows the baseline IOPS and latency information gathered on the PS Series environment on the left side. The right side shows the PS Series environment IOPS and latency, as a percentage of the baseline, with the same workload running on the SC Series environment that shares the iSCSI infrastructure.

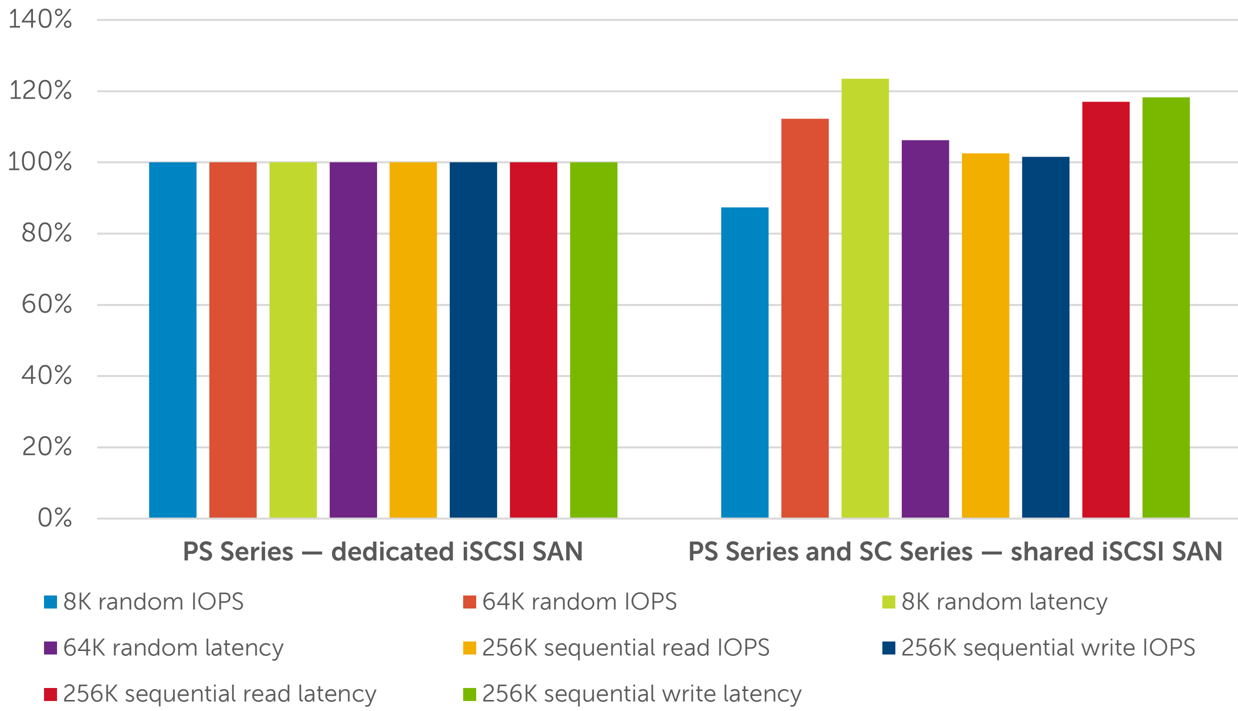

Figure 7 Dedicated host performance data

5.3 High availability testing

High availability was demonstrated by showing performance during an iSCSI switch failure. In this test scenario, a Linux host was connected to both PS Series and SC Series storage platforms. The Linux host had a total of 16 LUNs connected during the test, 8 LUNs from the PS Series and 8 LUNs from the SC Series.

The Linux host started a vdbench 256K write workload to all 16 LUNs. Roughly midpoint in the hour-long test, one of the two iSCSI switches was powered off, simulating a switch failure. The output chart in Figure 8 shows a drop in IOPS and a spike in latency at the time of the failure, as would be expected. Also, as expected, the latency quickly recovers to roughly double the latency observed prior to the failure. Additionally, at the time of the failure, IOPS encounter a momentary drop, but quickly recover to roughly half of the pre-failure IOPS. Overall, the performance observed during the switch failure is consistent with reducing the paths over which storage traffic is passed, by half.

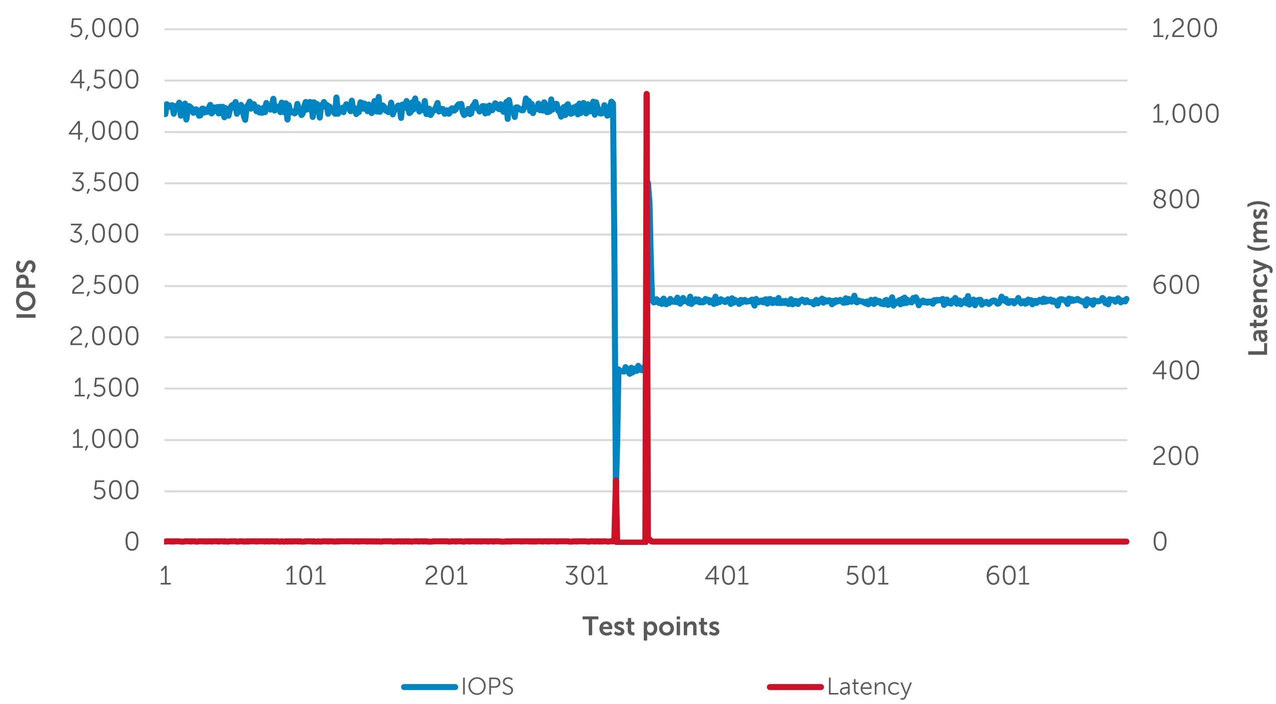

Figure 8 High availability test

6 Best practice recommendations

Always check the *Dell Storage Compatibility Matrix* to ensure that SAN components (such as switches and host NICs) are supported for both PS and SC Series deployments. For full support, the switches used for array connectivity must be validated for use with each solution individually.

6.1 Switch fabric

Place all storage interfaces (including storage controller ports and host ports) in the default (native) VLAN when the iSCSI SAN infrastructure is shared. If a non-default VLAN is assigned, set the ports as **untagged** or put them in **access** mode.

Use the Dell switch configuration guides for PS Series SANs to configure the switch fabric. These guides contain optimal settings for Jumbo frames, flow control, and other common settings recommended for Dell storage.

A switch interconnect (such as LAG, stack, VLT, or VPC) must be configured to maintain a single layer 2 fabric topology so that PS Series arrays can share the same infrastructure.

6.2 Host connectivity

Hosts must be configured for IPv4 connectivity. IPv6 is not supported at this time for shared connectivity.

Enable Jumbo frames (MTU of 9000 or greater) on host NICs used for storage connectivity to allow for greater throughput with large I/O request (block) sizes.

6.3 Storage

PS Series storage will auto-negotiate Jumbo frame size. For SC Series storage, enable Jumbo frames on each iSCSI SAN port through Storage Center. Refer to the SC Series administration guide on the Knowledge Center at the Customer Portal (login required) for the configuration steps.

7 Conclusion

As a result of testing and analysis shown in this document, introducing SC Series storage into an existing PS Series storage iSCSI environment, whether in a dedicated or shared host configuration, will not significantly affect the performance of the PS Series storage. Understanding the effects of a coexisting storage environment is key to enabling future growth and expansion of existing PS Series environments.

A Additional resources

A.1 Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

For additional support information on specific array models, see the following table.

Dell Storage	Online support	Email	Phone support (US only)
SC Series and Compellent	https://customer.compellent.com	support@compellent.com	866-EZ-STORE (866-397-8673)
SCv Series	http://www.dell.com/support	Specific to service tag	800-945-3355
XC Series	http://www.dell.com/support	Specific to service tag	800-945-3355
PS Series (EqualLogic)	http://eqlsupport.dell.com	eqlx-customer-service@dell.com	800-945-3355

Dell TechCenter is an online technical community where IT professionals have access to numerous resources for Dell software, hardware and services.

Storage Solutions Technical Documents on Dell TechCenter provide expertise that helps to ensure customer success on Dell Storage platforms.

A.2 Related documentation

See the following referenced or recommended Dell publications:

- *Dell PS Series Configuration Guide*: http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/19852516
- *Dell Storage Compatibility Matrix*: http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/19856862.aspx
- *Red Hat Enterprise Linux 6 or 7 Host Configuration* http://en.community.dell.com/techcenter/extras/m/white_papers/20438884
- *RHEL 6.x Best Practices for Dell Compellent Storage Center* http://en.community.dell.com/techcenter/extras/m/white_papers/20437964.aspx
- *Switch Configuration Guides for PS Series or SC Series SANs*: http://en.community.dell.com/techcenter/storage/w/wiki/4250.switch-configuration-guides-for-ps-series-or-sc-series-sans