ABSTRACT
This white paper provides technical information and best practices that should be considered when planning or designing deployment of ScaleIO for VMware environment. This guide also includes different performance tunings that can be applied before or after the deployment of ScaleIO.

September, 2016
The information in this publication is provided “as is.” EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC®, EMC, the EMC logo, ScaleIO, SRM, and VIPR are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners. © Copyright 2016 EMC Corporation. All rights reserved. Published in the USA. 09/16 white paper H14684.1

EMC believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

EMC is now part of the Dell group of companies.
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision History</td>
<td>4</td>
</tr>
<tr>
<td>Executive Summary</td>
<td>5</td>
</tr>
<tr>
<td>Audience</td>
<td>5</td>
</tr>
<tr>
<td><strong>ScaleIO Overview</strong></td>
<td>5</td>
</tr>
<tr>
<td>ScaleIO Architecture for VMware Environments</td>
<td>7</td>
</tr>
<tr>
<td>Advantages of ScaleIO in VMware Environments</td>
<td>8</td>
</tr>
<tr>
<td>Resource pools can be shared across different ESX Clusters</td>
<td>8</td>
</tr>
<tr>
<td>No rigid hardware requirements</td>
<td>9</td>
</tr>
<tr>
<td>Scalability</td>
<td>9</td>
</tr>
<tr>
<td>Deployment flexibility</td>
<td>9</td>
</tr>
<tr>
<td>No performance bottlenecks</td>
<td>9</td>
</tr>
<tr>
<td><strong>ScaleIO Networking in a VMware Environment</strong></td>
<td>10</td>
</tr>
<tr>
<td>Traffic Types</td>
<td>10</td>
</tr>
<tr>
<td>VMware implementations</td>
<td>11</td>
</tr>
<tr>
<td>Network requirements for ScaleIO deployment</td>
<td>11</td>
</tr>
<tr>
<td><strong>ScaleIO Installation on ESX Servers</strong></td>
<td>12</td>
</tr>
<tr>
<td>Deployment Prerequisites</td>
<td>12</td>
</tr>
<tr>
<td>Deployment via Plugin</td>
<td>13</td>
</tr>
<tr>
<td>Registering ScaleIO plug-in and uploading the OVA template</td>
<td>13</td>
</tr>
<tr>
<td>Installing the SDC on ESX hosts</td>
<td>15</td>
</tr>
<tr>
<td>Configure ScaleIO SDS</td>
<td>15</td>
</tr>
<tr>
<td>Troubleshooting ScaleIO installation plugin</td>
<td>18</td>
</tr>
<tr>
<td>Manual Installation</td>
<td>19</td>
</tr>
<tr>
<td>Scenarios where combination of ScaleIO plugin and manual installation is needed</td>
<td>19</td>
</tr>
<tr>
<td>Deploying the ScaleIO virtual machine (SVM)</td>
<td>20</td>
</tr>
<tr>
<td>Configure the UUID on virtual machines</td>
<td>21</td>
</tr>
<tr>
<td>Installing the SDC directly on the ESX host</td>
<td>22</td>
</tr>
<tr>
<td><strong>Accessing ScaleIO Volumes for VMware Environments</strong></td>
<td>22</td>
</tr>
<tr>
<td>Performance tunings for VMware Environments</td>
<td>24</td>
</tr>
<tr>
<td>ScaleIO specific system changes</td>
<td>24</td>
</tr>
<tr>
<td>Read Ram cache settings for SDS</td>
<td>25</td>
</tr>
<tr>
<td>VMware specific changes</td>
<td>25</td>
</tr>
</tbody>
</table>
OPTIMIZING THE SCALEIO VIRTUAL MACHINE (SVM) ................................................................. 25
OPTIMIZING ESX .................................................................................................................. 25
OPTIMIZING VM GUESTS .................................................................................................. 26
NETWORK TUNINGS ........................................................................................................... 28
CONCLUSION ......................................................................................................................... 30
REFERENCES ...................................................................................................................... 31

REVISION HISTORY

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Author</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov 2015</td>
<td>1.0</td>
<td>Vibhor Gupta</td>
<td>Initial Document</td>
</tr>
<tr>
<td>Aug 2016</td>
<td>2.0</td>
<td>Vibhor Gupta</td>
<td>Updates for ScaleIO 2.x</td>
</tr>
<tr>
<td>Sept 2016</td>
<td>2.0.1</td>
<td>Jason Brown</td>
<td>Added trademarks and document part #</td>
</tr>
</tbody>
</table>
EXECUTIVE SUMMARY

Dell® EMC® ScaleIO® is an industry leading Software-defined Storage (SDS) solution that enables customers to extend their existing virtual infrastructure into a high performing virtual SAN. ScaleIO creates a server SAN using industry-standard servers with direct attached storage (DAS). It can be deployed for hosting storage across a single or multiple VMware® ESX® clusters just like a traditional SAN either from an external pool of disks or from the DAS already present on the ESX hosts. Unlike solutions that simply pool DAS within a cluster, ScaleIO truly delivers the enterprise shared storage of a SAN but scales well beyond the capabilities of legacy frame based solutions. DAS can be clustered together into a storage system that services I/O requests using massive parallel processing. ScaleIO can scale from as little as three ESX hosts to thousands of ESX hosts. Increasing and decreasing, both the capacity and the IOPs can be done “on the fly” by adding or removing ESX hosts with minimal impact on the user or the application.

The goal of this white paper is to provide the best practices to install ScaleIO both via plug-in and manually in a VMware environment. This white paper also describes the different performance tunings that should be applied to achieve the optimal performance for certain workloads. This guide is intended to provide details on

- ScaleIO Deployment best practices via ScaleIO VMware vSphere® plugin
- Troubleshooting ScaleIO plugin
- ScaleIO Deployment best practices for manual installation
- Performance tunings for optimal performance

This paper does not intend to provide an overview of ScaleIO architecture. Please refer to the EMC ScaleIO Architecture white paper for further details.

AUDIENCE

This white paper is intended for internal Dell EMC technical employees such as Pre-sales Engineers, Professional Service engineers and the external customers and partners who are responsible for deploying and managing enterprise storage.

It is assumed that the reader has an understanding and working knowledge of the following:

- ScaleIO components, architecture, commands and features
- VMware components, architecture, commands and features

SCALEIO OVERVIEW

The management of large-scale, rapidly growing infrastructures is a constant challenge for many data center operation teams and it is not surprising that data storage is at the heart of these challenges. The traditional dedicated SAN and dedicated workloads cannot always provide the scale and flexibility needed. A storage array can’t borrow capacity from another SAN if demand increases and can lead to data bottlenecks and a single point of failure. When delivering Infrastructure-as-a-Service (IaaS) or high performance applications, delays in response are simply not acceptable to customers or users.

Dell EMC ScaleIO is software that creates a server-based SAN from local application server storage (local or network storage devices). ScaleIO delivers flexible, scalable performance and capacity on demand. ScaleIO integrates storage and compute resources, scaling to thousands of servers.

Unlike some of the other hypervisor-based solutions in which the storage is not spanned across the ESX cluster boundaries, ScaleIO is a SAN in true sense. In an ESX environment, ScaleIO allows native sharing of storage resources across multiple ESX clusters. This allows better performance for the applications, as the scaling across the ESX boundaries enables aggregation of the performance of more nodes and or devices across multiple clusters. Sharing resources across the boundaries eliminates storage silos and provide better utilization of the resources.

For ESX environments, ScaleIO provides a vSphere web plugin to automate installation and management of multiple ScaleIO systems, making it easier to manage. ScaleIO fully integrates with the VMware vRealize® suite using EMC ViPR®, giving customers a single pane of management for their entire ESX environment.

As an alternative to traditional SAN infrastructures and other Software Defined Storage (SDS) vendors, ScaleIO is hardware-agnostic, supports physical and/or virtual application servers, and has been proven to deliver significant TCO savings vs. traditional SAN.
Massive Scale - ScaleIO is designed to massively scale from three to thousands of nodes. Unlike most traditional storage systems, as the number of storage devices grows, so do throughput and IOPS. The scalability of performance is linear with regard to the growth of the deployment. Whenever the need arises, additional storage and compute resources (i.e., additional servers and/or drives) can be added modularly so that resources can grow individually or together to maintain balance. Storage growth is therefore always automatically aligned with application needs.

Extreme Performance - Every server in the ScaleIO system is used in the processing of I/O operations, making all I/O and throughput accessible to any application within the cluster. Such massive I/O parallelism eliminates bottlenecks. Throughput and IOPS scale in direct proportion to the number of servers and local storage devices added to the system, improving cost/performance rates with growth. Performance optimization is automatic; whenever rebuilds and rebalances are needed, they occur in the background with minimal or no impact to applications and users. The ScaleIO system autonomously manages performance hot spots and data layout.

Compelling Economics - As opposed to traditional Fibre Channel SANs, ScaleIO has no requirement for a Fibre Channel fabric between the servers and the storage and no dedicated components like HBAs. There are no “forklift” upgrades for end-of-life hardware. You simply remove failed disks or outdated servers from the cluster. It creates a software-defined storage environment that allows users to exploit the unused local storage capacity on any server. Thus ScaleIO can reduce the cost and complexity of the solution resulting in typically greater than 60 percent TCO savings vs. traditional SAN.

Unparalleled Flexibility - ScaleIO provides flexible deployment options. With ScaleIO, you are provided with two deployment options. The first option is called “two-layer” and is when the application and storage are installed in separate servers in the ScaleIO cluster. This provides efficient parallelism and no single points of failure. The second option is called “hyper-converged” and is when the application and storage are installed on the same servers in the ScaleIO cluster. This creates a single-layer architecture and provides the lowest footprint and cost profile. ScaleIO provides unmatched choice for these deployment options. ScaleIO is infrastructure agnostic making it a true software-defined storage product. It can be used with mixed server brands, operating systems (physical and virtual), and storage media types (HDDs, SSDs, and PCIe flash cards). In addition, customers can also use OpenStack commodity hardware for storage and compute nodes.

Supreme Elasticity - With ScaleIO, storage and compute resources can be increased or decreased whenever the need arises. The system automatically rebalances data “on the fly” with no downtime. Additions and removals can be done in small or large increments. No capacity planning or complex reconfiguration due to interoperability constraints is required, which reduces complexity and cost. The ScaleIO system reconfigures itself as the underlying resources change; data is rearranged and spread evenly on the servers to optimize performance and enhance resilience. All of this happens automatically without operator intervention and therefore eliminates the need for costly and disruptive data migrations.

Essential Features for Enterprises and Service Providers - ScaleIO offers a set of features that gives you complete control over performance, capacity and data location. For both private cloud data centers and service providers, these features enhance system control and manageability—ensuring that quality of service (QoS) is met. With ScaleIO, you can limit the amount of performance—IOPS or bandwidth—that selected customers can consume. The limiter allows for resource distribution to be imposed and regulated,
preventing application “hogging” scenarios. Data masking can be used to provide added security for sensitive customer data. ScaleIO offers instantaneous, writeable snapshots for data backups.

For improved read performance, DRAM caching enables you to improve read access by using SDS server RAM. Fault sets – a group of SDS that are likely to go down together, such as SDSs residing on nodes in the same physical rack – can be defined to ensure data mirroring occurs outside the group, improving business continuity. You can create volumes with thin provisioning, providing on-demand storage as well as faster setup and startup times.

ScaleIO also provides multi-tenant capabilities via protection domains and storage pools. Protection domains allow you to isolate specific servers and data sets. This can be done at the granularity of a single customer so that each customer can be under a different SLA. Storage pools can be used for further data segregation, tiering, and performance management. For example, data that is accessed very frequently can be stored in a flash-only storage pool for the lowest latency, while less frequently accessed data can be stored in a low-cost, high-capacity pool of spinning disks.

Just like any other SDS vendor, virtual machines hosted on ScaleIO can be replicated to a remote site for operational and disaster recovery. For more details on ScaleIO replication, please refer to the Protecting ScaleIO Data using RecoverPoint for Virtual Machines and Protecting Virtual Machine hosted on ScaleIO Using VMware vSphere Replication white papers.

SCALEIO ARCHITECTURE FOR VMWARE ENVIRONMENTS

A ScaleIO storage system consists of multiple instances of at least three main components working together in a clustered configuration:

- **ScaleIO Data Client (SDC):** SDC is a lightweight block device driver that exposes the ScaleIO shared volumes to the application. The SDC runs on the same server as the application.

  For VMware deployments, the SDC is installed directly on the ESX server and provides LUNs to be used as guest storage.

- **ScaleIO Data Server (SDS):** SDS is a lightweight software component that is installed on each server which contributes to the ScaleIO storage. SDS interfaces with the host’s volume manager and owns the local storage that contributes to the ScaleIO storage pools. The role of the SDS is to perform the back-end IO operations as requested by an SDC.

  For VMware deployments, the SDS is installed by deploying a preconfigured virtual machine known as the ScaleIO Virtual Machine (SVM).

- **Metadata Manager (MDM) and TieBreaker (TB):** A repository of storage metadata that manages the ScaleIO system and contains data required for system operation. MDM manages metadata such as device mappings, volume information, snapshots, device allocations, used and remaining capacity, errors and failures. The MDM also monitors the state of the storage system and is responsible for initiating system rebuilds and rebalances. MDM interacts asynchronously with SDC and SDS using a separate data path and will not impact their performance.

  Every ScaleIO storage system has a minimum of 3 MDMs: a Master MDM (active), one or more Slave MDMs (passive) and a TieBreaker (TB) to resolve any ‘split-brain’ I/O scenarios. MDM uses an Active/Passive methodology with a TieBreaker component where the master MDM is always active and the slaves are passive. In ScaleIO 2.0 a deployment can be configured with a total of 5 MDMs (1 Primary MDM, 2 Slave MDMs and 2 TBs).

  As with the SDS, in a VMware deployment the MDM and TB are installed with the ScaleIO Virtual Machine.

Figure 2 shows the ScaleIO ESX host architecture.
As mentioned before, SDS and MDM/TB are installed in a ScaleIO Virtual Machine (SVM) and SDC is installed directly on the ESX server (or even potentially on a guest operating system). The LUNs in the figure above can be formatted with VMFS and then exposed using the ESX hypervisor to the virtual machine, or they can be used as an RDM device. When the LUNs are used as an RDM device, the VMFS layer is omitted, and the guest operating system applies its own filesystem or structure to the presented volume.

ADVANTAGES OF SCALEIO IN VMWARE ENVIRONMENTS

There are different types of software defined storage (SDS) in the market for ESX environment, but most don’t have the required functionality to be classified as a server SAN. Unlike ScaleIO, other solutions in the market today don’t have key SAN functionality such as a) resource sharing and storage management for multiple vSphere clusters, b) consistent IOPS and latency at variable workload c) high availability and consistently low rebuild times to be classified as a Server SAN. Most solutions are in fact just hypervisor based DAS pools with all inherent DAS issues still present. The section below describes the advantages of ScaleIO over such DAS based offerings in a VMware environment.

RESOURCE POOLS CAN BE SHARED ACROSS DIFFERENT ESX CLUSTERS

Unlike other SDS offerings, ScaleIO allows native sharing of storage resources across multiple ESX clusters. By spanning storage resources across multiple ESX clusters, ScaleIO eliminates storage silos which reduces the wastage of the hardware resources. ScaleIO distributes the I/O automatically and seamlessly among a larger pool of shared storage which can result in higher performance and increases ease of use.

This leads to lower TCO, because hardware is more effectively utilized; and lower OPEX, because management overhead is reduced.
For more details on ScaleIO architecture advantages over traditional SAN and other SDS offerings, please refer to the “Dell EMC ScaleIO for ESX Environments: Architecture Matters” white paper hosted on emc.com.

**NO RIGID HARDWARE REQUIREMENTS**

Unlike other SDS vendors, ScaleIO allows customers to use the latest server hardware available. Hardware can be sourced from multiple suppliers, or engineered in-house. Maintenance and procurement are simplified.

Also in contrast to other SDS server solutions, which requires that a configuration to be either all flash or spinning media with an SSD cache, ScaleIO allows customers to create pools of storage using different media types, and allows customers to grow CPU and memory resources independently of storage.

**SCALABILITY**

ScaleIO can scale from a minimum of 3 nodes to a maximum of 1024 nodes within a single cluster. No other SDS vendor provides such a massive scalability.

Scalability in ScaleIO is very granular which can range from a single disk or node to multiple nodes at a time. This provides ScaleIO an advantage over other SDS vendors where the capacity can only be scaled in a pre-defined set number of nodes. This usually leads to either under or over provision of resources.

**DEPLOYMENT FLEXIBILITY**

Unlike other SDS server offerings, which are tied to their hypervisor and cannot be deployed beyond their virtualized environment, ScaleIO enable customers to share the resources outside the ESX platforms and applications, even if the storage infrastructure is built on top of ESX.

ScaleIO also allows the customers to scale the storage and compute resources independently. This allows infrastructure to align with application performance requirements. Other SDS vendors are less flexible in a sense that if you add storage, you must also add some amount of compute.

**NO PERFORMANCE BOTTLENECKS**

In a ScaleIO system, the data is striped across all the servers in a balanced manner such that the IOPS and throughput are accessible to any application within the cluster. The IOPS and throughput are directly proportional to the number of servers and or local storage devices added to the system. ScaleIO allow customers to add the storage devices or servers on the fly to optimize the performance based on their needs.
The performance optimization is automatic; whenever rebuilds and rebalances are needed, they occur in the background with minimal or no impact to applications and users. Since the data volumes are always balanced across the nodes, the performance of the ScaleIO system is always predictable.

Some of the other SDS vendors, keep a local copy of the data, which ties the performance of the application to the hardware resources of that server. The performance of such SDS solutions fluctuates in case when the application has to failover to some other node due to a failure or in scenarios when the application doesn’t fit in a single server due to compute memory and storage limitations.

ScaleIO also has an advantage over other SDS vendors in terms of memory requirement. ScaleIO components i.e. SDS, SDC and MDM are very lightweight and utilizes very little memory and CPU to operate, leaving ample resources for the applications to consume. Many of the other SDS offerings utilizes more than half of the system resources leaving very little for the application.

In a test conducted by an independent, 3rd party vendor named StorageReview, ScaleIO has outperformed all the other SDS offerings in terms of highest IOPS, lowest latency, lowest memory and lowest CPU utilization. For more details, follow the ScaleIO articles on http://www.storagereview.com.

SCALEIO NETWORKING IN A VMWARE ENVIRONMENT

The software components that make up ScaleIO (the SDCs, SDSs, and MDMs) converse with each other in a predictable way. Architects designing a ScaleIO deployment should be aware of these traffic patterns in order to make informed choices about the network layout. Default TCP port numbers and instructions for changing them (if needed) are listed in the ScaleIO User Guide.

TRAFFIC TYPES

The following network traffic types constitute a ScaleIO system.

- **ScaleIO Data Client (SDC) to ScaleIO Data Server (SDS)**
  
  Traffic between the SDCs and the SDSs forms the bulk of “front end” storage traffic. Front end storage traffic includes all read and write traffic arriving at or originating from a client. This network has a high throughput requirement.

  The traffic between the SDC(s) and SDS(s) forms the bulk of “front end” storage traffic. This network is used just for the ScaleIO read and writes operations. This network has a high throughput requirement. If there is a multitenancy requirement, ScaleIO SDC to SDS traffic can be isolated using VLANs and network firewalls.

  **Note:** This network can also be configured using the manual installation or post installation using the ScaleIO installation plugin.

- **ScaleIO Data Server (SDS) to ScaleIO Data Server (SDS)**
  
  Traffic between SDSs forms the bulk of “back end” storage traffic. Back end storage traffic includes writes that are mirrored between SDSs, rebalance traffic, and rebuild traffic. This network has a high throughput requirement.

  Although not required, there may be situations where isolating front-end and back-end traffic for the storage network may be ideal. This may be true in two-layer deployments where the storage and server teams act independently.

  **Note:** This network can also be configured using the manual installation or post installation using the ScaleIO installation plugin.

- **Meta Data Manager (MDM) to Meta Data Manager (MDM)**

  MDMs are used to coordinate operations inside the cluster. They issue directives to ScaleIO to rebalance, rebuild, and redirect traffic. MDMs are redundant, and must communicate with each other to maintain a shared understanding of data layout. MDMs also establish the notion of “quorum” in ScaleIO.

  MDMs do not carry or directly interfere with I/O traffic. MDMs do not require the same level of throughput required for SDS or SDC traffic. MDM to MDM traffic requires a reliable, low latency network. MDM to MDM traffic is considered “back end” storage traffic.

  **Note:** ScaleIO 1.32 and earlier require MDM and SDS traffic to be on the same network. ScaleIO 2.0 supports the use of one or more networks dedicated to traffic between MDMs. In either case, at least two 10 gigabit links should be used for each network connection.

- **Meta Data Manager (MDM) to ScaleIO Data Client (SDC)**

10
The master MDM must communicate with SDCs in the event that data layout changes. This can occur because the SDSs that host storage for the SDCs are added, removed, placed in maintenance mode, or go offline. Communication between the Master MDM and the SDCs is asynchronous.

MDM to SDC traffic requires a reliable, low latency network. MDM to SDC traffic is considered “front end” storage traffic.

- **Meta Data Manager (MDM) to ScaleIO Data Server (SDS)**
  The master MDM must communicate with SDSs to issue rebalance and rebuild directives. MDM to SDS traffic requires a reliable, low latency network. MDM to SDS traffic is considered “back end” storage traffic.

- **Management traffic**
  This network is optional but recommended. These IP addresses can be used to provide access to ScaleIO management applications like ScaleIO Gateway (REST Gateway, Installation Manager, and SNMP trap sender), traffic to and from the Light Installation Agent (LIA), and reporting or management traffic to the MDMs (such as syslog for reporting and LDAP for administrator authentication).

  It should be noted that the management network does not come in the control plane data, which means that the ScaleIO system does not require nor use this particular network interface for any ScaleIO internal tasks.

**VMWARE IMPLEMENTATIONS**

VMware-based networking provides all the options of a physical switch, but gives more flexibility within network design. To make use of virtual networking, virtual network configurations must be consistent with physical network devices connected to the virtual structure. Just as would be necessary without network virtualization, uplinks to physical switches must take into account redundancy and bandwidth. We recommend determining traffic patterns in advance, in order to prevent bandwidth starvation.

- **ScaleIO Data Client**
  SDC is installed directly into the ESX as a kernel module. SDC leverages VMware kernel ports (VMkernel NICs) and does not require any network configuration profiles to communicate to SDS or MDM.

- **VMKernel Port**
  VMkernel Port is often used for VMware vSphere® vMotion®, storage network, fault tolerance, and management. We recommend having a higher priority on network traffic over virtual machine traffic, for example virtual machine port groups or user-level traffic.

  **Note:** VMkernel needs to be manually configured on ESX host to use IPv6. IPv6 is not supported in converged VMware environments.

- **Virtual Machine Port Group**
  Virtual Machine Port Group can be separate or joined. For example, you can have three virtual machine port groups on the same VLAN. They can be segregated onto separate VLANs, or depending on the number of NICs, they can be on different networks.

- **VMware Advantages over a Standard Switch**
  NetIOC provides the ability to prioritize certain types of traffic, for example – storage traffic can be given a higher priority than other types of traffic. This will only work with VMware distributed switches and not standard switches.

  This can also be done with Datacenter Bridging (also known as Priority Flow Control), and could be configured with standard QoS; however not all switches support these features.

**NETWORK REQUIREMENTS FOR SCALEIO DEPLOYMENT**

The vSphere web client server must have access to the host on which the PowerShell script will be used. The management network on all of the ESXs that are part of the ScaleIO system must have the following items configured:

- **Virtual Machine Port Group or VMkernel Network Adapters.**
– For installation via web plugin, it is required to use Virtual Port Group with the same name for all the ESX hosts.

- When using distributed switches, the vSphere Distributed Switch (vDS) must have the following items configured:
  – VMkernel port (necessary only if using a single network)
  – dvPortGroup for virtual machines

- If using only a single network (management), you must manually configure the following:
  – vSwitch
  – VMkernel Port
  – Virtual Machine Port Group

For more details on the ScaleIO networking for VMware, please refer to the ScaleIO Networking Best Practices Guide.

**SCALEIO INSTALLATION ON ESX SERVERS**

In a VMware environment, the ScaleIO vSphere plug-in deployment wizard is used to install the ScaleIO Data Server (SDS) and the Metadata Manager (MDM) component on a dedicated ScaleIO virtual machine (SVM), whereas the ScaleIO Data Client (SDC) is installed directly on the ESX hosts. There are two ways to deploy ScaleIO for VMware:

- Deployment via ScaleIO plugin
- Manual installation

**DEPLOYMENT PREREQUISITES**

Before deploying ScaleIO on ESX environment, ensure compliance with the following prerequisites:

The ESX host should meet the following hardware requirements:

<table>
<thead>
<tr>
<th>Table 1. ScaleIO deployment prerequisites</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Component</strong></td>
</tr>
<tr>
<td>Processor</td>
</tr>
<tr>
<td>Physical memory</td>
</tr>
<tr>
<td>Disk space</td>
</tr>
<tr>
<td>Supported Hypervisor</td>
</tr>
<tr>
<td>Java™ requirements</td>
</tr>
</tbody>
</table>

ESX hosts that are selected to have either MDM, TieBreaker, or SDS component installed on them, must have a defined local datastore, with a minimum of 10 GB free space (to be used for the SVM).

- If the ESX is only being used as an SDC, there is no need for this datastore.

Each SDS needs a minimum of 1 storage device for a total of 3 SDS, that all meet the following prerequisites:

- A minimum of 100 GB available storage capacity per storage device.
- The devices must be free of partitions.
• If a device has a datastore on it, before adding the device, you must either remove the datastore, or use the plug-in **Advanced settings** option to enable VMDK creation.

• If the device has the ESX operating system on it, you must use the plug-in **Advanced settings** option to enable VMDK creation. It is important to note that by enabling this option instead of creating RDM device, a VMDK will be created which is time consuming. Please refer to VMDK vs. RDM comparison section for more details.

The host from which you run the PowerShell (.ps1) script must have the following prerequisites:

• Runs on Windows, with Java installed.

• VMware vSphere® PowerCLI™ (not Windows PowerShell) is installed and of the matching version with the VMware vCenter®.

• VMware vCenter 5.5 requires PowerCLI 5.5 and vCenter 6.0 requires PowerCLI 6.0.

• Has incoming and outgoing communication access to the vCenter instance.

To deploy ScaleIO on ESX 6.0 servers, it is recommended to disable the new VMware security features which can prevent the ScaleIO deployment. Make sure to disable the SSH lockout and the timeout features, by performing one of the following:

• Connecting to ESX remotely via vSphere client.

• Connecting to ESX remotely via vSphere web client.

• Connecting to ESX remotely via SSH connection.

• Connecting to ESX locally (either on-site or via console).

Please follow the ScaleIO deployment guide to find the commands for disabling SSH lockout and the timeout features.

**DEPLOYMENT VIA PLUGIN**

The ScaleIO plugin for vSphere enables you to perform all the ScaleIO activities in a simple and efficient manner for all the ESX nodes in the vCenter. The plug-in can be used for provisioning ScaleIO nodes and on an existing ScaleIO cluster adding additional SDS nodes.

**Best practice:** It is recommended to use ScaleIO plugin for the deployment and provisioning process. There are some scenarios when the combination of manual installation and plugin is required. Please refer to the Manual installation section for further details.

There are 3 main tasks in deploying ScaleIO:

• Registering the ScaleIO plug-in and uploading the OVA template.

• Installing the SDC on ESX hosts.

• Deploying ScaleIO.

**REGISTERING SCALEIO PLUG-IN AND UPLOADING THE OVA TEMPLATE**

This section will describe the steps and best practices to register the ScaleIO plugin for vCenter. Download the ScaleIO image. Extract it to find the OVA file.

• From PowerCLI, run the following script, which is extracted from the ScaleIO download package:

  ScaleIOPluginSetup-2.0-XXX.X.ps1
- Enter the vCenter name or IP address, user name, and password.
- Select option 1 to Register ScaleIO Plugin.
- Read the upgrade notice, and enter y to continue.
- For Select Registration Mode, choose Standard.

Note: You can use the Advanced option to install the plug-in using a ScaleIO Gateway from a previous installation or using your own web service. In either case, you must place this version’s plugin.zip file (EMC-ScaleIO-vSphere-web-plugin-2.0-XXX.X.zip) in your resources folder before running the installation. To use a previous ScaleIO Gateway, the resource folder is ScaleIO Gateway installation folder\webapps\root\resources.

Scenarios in which your local machine have a firewall enabled and doesn’t allow you to start the HTTP service, it is recommended to use Advanced option to install the plug-in.

- Log in to the VMware vSphere® Web Client. If you are already logged in, log out, and then log in again.
- In the PowerCLI window, press Enter to finish the plug-in download and return to the menu, do not exit the script.
- Verify that the ScaleIO plug-in is registered on the vCenter.
  - Login to vCenter and click on Home. You should see the EMC ScaleIO plugin there. See Figure 4.

Note: If the plugin is missing, please refer to the Troubleshooting ScaleIO Installation Plug-in section.

To upload the OVA template, perform the following:

- From the PowerCLI script, select 3 to Create SVM template.
- Enter the parameters prompted by the plug-in.
**Best practice:** For faster deployment in large-scale environment (100 nodes or more), it is recommended to use the OVA to create SVM templates, as many as eight datastores. To do so, enter the datastore names, and when you are done, leave the next line blank. It is recommended to check the names of the datastore on VMware vCenter client to avoid any errors.

- When the process is complete, enter 4 to exit the plug-in script.

**INSTALLING THE SDC ON ESX HOSTS**

From version 1.31 and beyond, it is a prerequisite to install SDC component on all of the ESX hosts within the ScaleIO system, regardless of what role the host is playing. To install the SDC component, perform the following:

- From the Basic tasks section of the EMC ScaleIO screen, click **Install SDC on ESX**.
- Select the ESX hosts, and enter the root password for each host.
- Click Install. The status appears in the dialog.
- When finished, click Finish.
- Restart each ESX host.

**Note:** Must restart the ESX host before proceeding.

**CONFIGURE SCALEIO SDS**

This section describes the best practices to deploy ScaleIO in the VMware environment. Before deploying ScaleIO, it is recommended to configure advanced installation options.

VMware allows users to have two different formats for storage: virtual machine disk (VMDK) or raw device mapping (RDM). It's important to decide which option is preferable prior to the ScaleIO deployment:

<table>
<thead>
<tr>
<th>Table 2. RDM vs. VMDK</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RDM (Highly recommended)</strong></td>
</tr>
<tr>
<td><strong>VMDK</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

**Note:** Installation using VMDK takes significantly more time as compared to RDM.

With ScaleIO version 2.x, there are two new features in the advanced settings:

- **Allow the takeover of devices with existing signature:** This feature enables the reuse of devices that were part of a previous ScaleIO system, and were not removed from that system properly. This will revert to not selected after use, and must be reactivated.

- **Allow using non-local datastores for ScaleIO gateway:** This feature enables the deployment of the Gateway SVM on non-local datastores (such as remote storage), in case of insufficient space on the local datastore, or if no datastore can be found.

**Best practice:** If the ScaleIO is being deployed for a very large system (100 or more nodes), it is recommended to increase the parallelism limit (default 100) thus speeding the installation process. However, this is dependent on the processing power of the vCenter system.

To deploy a new ScaleIO system, follow the steps below:

- Click on **Deploy ScaleIO environment**. In the Select Installation type screen, there are options of either creating a new ScaleIO system or to add servers to the existing ScaleIO system.
You can also deploy the ScaleIO Gateway for a ScaleIO system that was registered before. Select this option to use the wizard to prepare the environment without performing ScaleIO configuration (Protection Domains, Storage Pools, etc.).

- If you selected **Create a new ScaleIO system**, review and agree to the license terms, then click **Next**.
- In the **Create New System** screen, enter System name and the admin password.
- **Add ESX hosts** to the ScaleIO cluster. To deploy ScaleIO, minimum of three ESX hosts should be selected.
- In the Select management component screen, there are options of creating a 3-node cluster or a 5-node cluster. With a 3-node cluster, select an ESX host each for an Initial Master MDM, Manager MDM and TieBreaker MDM. For a 5-node cluster, you can configure a single Master MDMs, two Slave MDMs and two TieBreaker.
- Configure Performance profile: This screen allows you to select the ScaleIO components that needs to be configured for a high performance. This step is optional and the performance profile can be activated after the installation as well. It should be noted that the performance profile and sizing affects the ScaleIO Virtual Machine (SVM) memory consumption. Please refer to the **performance tuning** section for more details.
- **Configure protection domain**. There should be at least one protection domain.
- **Configure storage pools**. There should be at least one storage pool. Select to which protection domain to add the storage pool to.
  
  - To enable zero padding, select **Enable zero padding**. Zero padding must be enabled for use with RecoverPoint replication. Zero padding must also be enabled to allow background device scanning as well.  
    
    **Best Practice**: It is recommended to enable zero padding at the time of installation. The zero padding policy cannot be changed after the addition of the first device to a specific Storage Pool.

- **Create fault sets**. This step is optional.
  
  - If creating a fault set, there should be a minimum of three fault sets created.

- Configure the following for every ESX host or SVM that contributes to the ScaleIO storage.
  
  - Whether the ESX host is acting as an SDS.
  
  - If the SVM is an SDS, select a Protection Domain (required) and Fault Set (optional).
  
  - If the SDS has any flash devices, select **Optimize for Flash** to optimize ScaleIO efficiency for the flash devices.

- **Assign ESX host devices to ScaleIO SDS component**
  
  - Select storage devices to add to a single SDS. A minimum of 1 device per SDS is needed.
  
  - **Replicate selections**. Select devices for the other SDSs by replicating selections made in the Select device tab. To replicate device selection, all of the following conditions should be met:

    - The number of devices on each ESX must be identical.

    - Source and target devices must be identical in the following ways: a) both are SSD or non-SSD, b) both have datastores on them or do not, c) both are roughly the same size (within 20%), and d) both are connected via a RAID controller or directly attached.

    - At least one of the following conditions must be met: a) both SDSs are in the same Protection Domain, b) both SDSs are in different Protection Domains, but with the same list of Storage Pools, or c) the target SDS is in a Protection Domain with only one Storage Pool.

**Note**: If the devices cannot be selected (greyed out), please hover the mouse over the devices to read the message.
If local RDM is not connected via RAID, select **Enable RDM on non-parallel SCSI controller**.

- The physical device does not support RDM. **Enable VMDK**
- The device already has a datastore, and the device isn’t being completely used. The excess area that is not already being used will be added as a ScaleIO device. **Remove Datastore** or **Enable VMDK**

**Note:** By default, RDM is always enabled and it is the preferred way of installation. In order to use RDM, a disk must be attached to a RAID controller. However, ESX does not perform any validation, whether the RAID controller exists and it will allow the RDM creation which might cause the ESX host to crash.

To avoid that, ScaleIO has a solution to identify the disc type, which in case of RAID controller is usually "parallel SCSI".

However, some of the vendors and OEMs are presenting it in a different way, therefore if the user is sure that the connection type is not "parallel SCSI" but the controller is still there, the user can bypass the verification by disabling this option in advanced settings.

- Select ESXs to add as SDCs to ScaleIO system. For each ESX host to be added to an SDC, perform the following:
  - Select the SDC check box.
  - Type the ESX root password.
  - Disable LUN comparison regardless of the existence of other operating systems.

**Best practice:** In general, in environments where the SDC is installed on ESX and also on Windows/Linux hosts, you should set this to **Disabled**.

- **Configure Upgrade Components**
  - Configure the ScaleIO Gateway, by performing the following:
    - Select an ESX to host the Gateway virtual machine. A unique SVM will be created for the Gateway.
    - Enter and confirm a password for the Gateway admin user. This password need not be same as the ScaleIO cluster password.
    - Configure the Lightweight Installation Agent (LIA) on the SVMs. Enter the password. This password need not be same as the ScaleIO cluster password.

- **Select OVA template.**
  - Select the template to use to create the ScaleIO virtual machines (SVM). The default is EMC ScaleIO SVM Template. If you uploaded a template to multiple datastores, you can select them all, for faster deployment.
  - Enter and confirm the new root password that will be used for all the SVMs to be created. This will be the password of the ScaleIO virtual machines.

**Note:** ScaleIO does not support customer template for ScaleIO installation and hence it is not recommended. However, if custom template is used, please ensure it is compatible with the ScaleIO installation plug-in and ScaleIO MDM.

- **Configure Network.** There are options to use either a single network for management and data transfer, or to use separate networks. Please refer to the [ScaleIO Networking Best Practices Guide](#) to learn more about how to configure different VMware networks.
  - The management network, used to connect and manage the SVMs, is normally connected to the client management network, a 1 GB network.
  - The data network is internal, enabling communication between the ScaleIO components, and is generally a 10GB network.
  - To use one network (IPv4 or IPv6) for management and data, select a management network and proceed to the next step.
  - To use separate networks (IPv4 or IPv6), select a management network label and one or two data network labels. If the data network already exists, select it from drop-down box. Otherwise, configure the data network by clicking Create new network. Enter the information in the wizard and it will automatically create the following data network:
- vSwitch
- VMkernel Port
- Virtual Machine Port Group
- VMkernel Port Binding

**Best practice:** These are the recommended steps while configuring the network.

- Separating the networks is recommended for security and efficiency.
- It is recommended to check the selected networks must have communication with all of the system nodes. In some cases, while the wizard does verify that the network names match, this does not guarantee communication, as the VLAN IDs may have been manually altered.

- **Configure SVM.** For each SVM, perform the following.
  - Enter the IP address, Subnet mask, and the Default Gateway.
  - You can select a datastore to deploy SVM, or allow automatic selection.

- Review the changes and click Finish to begin deployment or click Back to make changes.

During the deployment process, following operations can be performed:

- View progress
- View logs
- **Pause** the deployment. After pause, the following operations can be performed
  - **Continue**—Click Continue deployment.
  - **Abort**—Click Abort.
  - **Roll back all deployment activities**—Click Cancel and Rollback. Rollback cannot be canceled once started.
  - **Roll back failed tasks only**—Click Rollback failed tasks. Rollback cannot be canceled once started.

- When deployment is complete, click **Finish**

If a task failed, you can look at the error in show detail. Click on **Continue** to try again until all tasks are complete.

**TROUBLESHOOTING SCALEIO INSTALLATION PLUGIN**

This section describes some of the things to be careful of and the troubleshooting best practices for the ScaleIO Plugin.

- Check that the 64-bit version of Java is installed. Java 32-bit version doesn’t work.
- Install the correct version of PowerCLI (5.5 or 6.0 depending on your vCenter version).
- Check that you are running 64-bit PowerCLI version. The 32-bit version doesn’t work.
- If there are certificate errors while installing the plugin, disable them using the following command
  ```powershell
  Set-PowerCLIConfiguration -InvalidCertificateAction Ignore
  ```
- If the host from which the PowerCLI is running from being multi-homed (has more than one IP address), select the **Advanced** mode and enter the IP address that the vCenter is able to connect to.
- ScaleIO Plug-in installation requires a logoff and login to VMware vSphere to complete the following:
The script starts an internal web server to host the plugin that only runs while the script is running.

The vCenter will only try to download and install the plugin during the login process, so the web server needs to be up during that time.

- If the plugin successfully installs and you can see it in the vSphere client but not in the vCenter WebUI, you may need to restart the WebUI. It is also recommended to restart vSphere-client services.

- For deeper troubleshooting, check the logs on vCenter. Login via ssh and find the Virgo log:
  - For 5.5: `/storage/log/vmware/vsphere-client/logs # grep -i vsphere_client_virgo.log`
  - For 6.0: `/var/log/vmware/vsphere-client/logs # grep -i scaleio vsphere_client_virgo.log`

- Another common issue is that the vCenter server and the server running the plugin installation are not able to communicate properly, which impacts the installation of the ScaleIO plugin on the vCenter server. If the installation of the plugin fails and the log gives no indication of the communication errors, look at the vCenter to ensure that there is no partially installed plugin. It is recommended to unregister the plugin, reregister and run the installation again.

  - Clean up the vSphere plugin:
    - Unregister the plugin.
    - Delete the files under "C:\Windows\System32\config\systemprofile\AppData\Roaming\VMware\scaleio"
    - Stop the service vSphere Web Client on vCenter.
      - Delete the files of the plugin in the folder "C:\ProgramData\VMware\vSphere Web Client\vc-packages\vsphere-client-serenity"
    - Start the service vSphere Web Client on vCenter.
  - Register the plugin again.

- If performance of the ScaleIO plugin is slower than expected, it is recommended to increase the memory assigned to the VMware vCenter Server Virtual Appliance (vCSA).

  The process of memory allocation is different in vSphere vCenter version 6.0. vCSA 6.0 now includes a built-in dynamic memory reconfiguration process that automatically runs at boot up. It is recommended by VMware in the vSphere performance best practice guide to monitor the vSphere web client memory usage after the plugin is installed. Even with 16GB of RAM assigned to the VCSA, it only automatically assigned <1GB. To increase the memory for example from 1GB to 2GB, execute the following command:

  ```
  cloudvm-ram-size -C 2048 vsphere-client
  ```

  and then restart the vSphere client:

  ```
  service vsphere-client restart
  ```

MANUAL INSTALLATION

This section describes when and how to install ScaleIO manually in a VMware environment. It is highly recommended to install ScaleIO using the VMware deployment plugin, however there are some scenarios where the manual installation is preferred.

SCENARIOS WHERE COMBINATION OF SCALEIO PLUGIN AND MANUAL INSTALLATION IS NEEDED

Manual installation of ScaleIO is not recommended. ScaleIO vSphere web-plugin should be used whenever possible and only in some scenarios where ScaleIO plugin or combination of plugin and manual installation cannot be done, then only manual installation should be used.

Scenarios where both ScaleIO web plugin and manual installation is required are:
1. When customer wants to “passthrough” the raid controller to the ScaleIO virtual machine (SVM). This scenario requires manual installation.

2. The combination of the ScaleIO web plugin and manual installation is preferred when there are more than 2 NICs or if vLAN tagging is being used from the network perspective. Please refer to the ScaleIO Networking Best Practices document for more details.

3. The combination of the ScaleIO web plugin and manual installation is preferred when the SVM's needs more than 1 network and static routes are required to support the data traffic.

Before starting, obtain the IP addresses of all of the nodes to be installed, and an additional IP address for the MDM cluster, as described in the following table:

**Table 3. ScaleIO network configuration**

<table>
<thead>
<tr>
<th>Number of networks</th>
<th>IP addresses required</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 network</td>
<td>• 1 for vmnic</td>
</tr>
<tr>
<td>(Management network)</td>
<td>• 1 for SVM</td>
</tr>
<tr>
<td>2 networks</td>
<td>• 2 for vmnic (1 for Mgmt., 1 for data)</td>
</tr>
<tr>
<td></td>
<td>• 2 for SVM (1 for Mgmt., 1 for SVM)</td>
</tr>
<tr>
<td>3 networks</td>
<td>• 3 for vmnic (1 for Mgmt., 2 for data)</td>
</tr>
<tr>
<td></td>
<td>• 3 for SVM (1 for Mgmt., 2 for data)</td>
</tr>
</tbody>
</table>

**DEPLOYING THE SCALEIO VIRTUAL MACHINE (SVM)**

This section describes as how to deploy the SVM and prepare it for the installation of ScaleIO. It is required that the SDC is installed directly on ESX and all the other components are installed on ScaleIO virtual machines (SVM).

**Note:** Before deploying ScaleIO, it is recommended to ensure that the networks for each ESX host which are part of the ScaleIO system have a properly defined Port Group and VMkernel.

To deploy the SVM on the ESX, perform the following:

- Use the vSphere client to connect to a vCenter.
- Select File > Deploy OVF Template
- Enter the full path to the OVA, click Next, accept all default values and click Finish.
- Clone the SVM to each ESX host.
- On each ESX physical host, configure the network and NTP (and DNS, if necessary).
  - Using the console, start the SVM and login. The default user name is root and the default password is admin.
  - Configure the network:
    - IP address
    - Gateway address
    - DNS
    - NTP server
- On each SVM in the system, install the following ScaleIO components. The SVM doesn't contain pre-installed packages. The packages are located under /root/install directory and must be installed manually.
  - Extract LIA installation:
    `/opt/emc/scaleio/lia/cfg/GPG/RPM-GPG-KEY-ScaleIO`
  - Import the key:
rpm --import RPM-GPG-KEY-ScaleIO

- Install the MDM components:
  - For both cluster and single mode, run the following command, from the node which is to act as MDM:
    ```
    rpm -i <full rpm file path>
    ```
  - For cluster mode, continue here:
    - Install the second MDM by repeating the previous step on a different server.
    - On a third server, install the TieBreaker, by running the following command
      ```
      rpm -i <full rpm file path>
      username: admin
      password: admin
      ```
    - Install Call-Home component (optional), by running the following command on each MDM node:
      ```
      rpm -i <full rpm file path>
      ```
    - Install the SDS component on every server that will contribute storage devices to the ScaleIO system by running the following command
      ```
      rpm -i <full rpm file path>
      ```

  **Note:** When using fast devices (SSD), you can optimize SDS parameters by replacing the previous command with the following:

  ```
  CONF=IOPS rpm -i <full rpm file path>
  ```

**CONFIGURE THE UUID ON VIRTUAL MACHINES**

To configure the virtual machine, you must enable the UUID attribute on every virtual machine, by performing the following:

- Start the vSphere client, and log in to a vCenter Server.
- Select Virtual Machines and Templates.
- From the **Virtual Machines** tab, right-click the virtual machine, and choose **Power > Power Off**.
- Right click the virtual machine, and click **Edit Settings**.
- From the Options tab, Select Advanced > General in the settings column.
- Click Configuration Parameters.
- Click Add Row, and enter the following:
  - In the **Name** column, enter the disk.EnableUUID.
  - In the **Value** column, enter **TRUE**.
- Click OK.
- Power on the virtual machine.
- Repeat the procedure for every SVM in the ScaleIO system.
INSTALLING THE SDC DIRECTLY ON THE ESX HOST

To install the SDC component on an ESX host, perform the following:

- Set the acceptance level of your host to **PartnerSupported**, by typing:
  
  ```
  esxcli software acceptance set --level=PartnerSupported
  ```

- Install the SDC VIB, by typing the following:
  
  ```
  esxcli software vib install -d <full_path_to_VIB>
  ```

- Reboot the ESX host. The SDC will not automatically boot at this point. It is required to update the SDC GUID and MDM IP address parameters:

  - **GUID**, for example: 12345678-90AB-CDEF-1234-567890ABCDEF

  **Note**: GUID (globally unique identifier) is a 128-bit structure that is used when there are multiple systems or clients generating IDs that needs to be unique. GUID should be unique to avoid collision. Please avoid using the same GUID with the difference of some digits. Please search the internet to create random GUIDs. One such example is [http://www.guidgen.com/](http://www.guidgen.com/)

  - **MDM IP addresses.** It is required to define multiple MDM clusters, each with multiple IP addresses.
    - Separate the IP address of the same MDM cluster with a “,” symbol.
    - Separate the multiple MDM cluster with “+” symbol.

- To update the SDC parameter, type the following command:
  
  ```
  esxcli system module parameters set -m scini -p "IoctlIniGuidStr=<GUID>\nIoctlMdmIPStr=<MDM_IPs>"
  ```

- Back up the parameters, by typing the following:
  
  ```
  /bin/auto-backing.sh
  ```

- Load the SDC module, by typing the following:
  
  ```
  esxcli system module load -m scini
  ```

ACCESSING SCALEIO VOLUMES FOR VMWARE ENVIRONMENTS

This section describes the steps to access the ScaleIO volumes after the installation is completed.

- **Login to ScaleIO Primary MDM cluster by executing the following command:**
  
  ```
  scli --login --username <username>
  ```

- **Create ScaleIO volume by typing the following command:**

  ```
  Syntax
  scli --add_volume {((--protection_domain_id <ID> | --protection_domain_name <NAME>)
  storage_pool_name <NAME> | --storage_pool_id <ID>) --size_gb <SIZE> [--volume_name <NAME>]}
  [Options] [Obfuscation Options] [Use RAM Read Cache Options]
  ```

  Example: `scli --add_volume --volume_name test_vol --protection_domain_name pd1 --storage_pool_name ssd --size_gb 8`
- Map the ScaleIO volume to the SDC node by typing the following command:

```bash
scli --map_volume_to_sdc (--volume_id <ID> | --volume_name <NAME>) (--sdc_id <ID> | --sdc_name <NAME> | --sdc_guid <GUID> | --sdc_ip <IP>) [--allow_multi_map]
```

Example: `scli -map_volume_to_sdc -volume_name test_vol -sdc_id fd25fe30000000008`

- Create VMware datastores on the ESX host where the volume is mapped. Follow these steps to create a datastore.
  - Login to VMware vSphere web client.
  - Click on vCenter Inventory Lists.
  - Click on Datastores under Resources.
  - Click on Create new datastore icon.
  - Select the DataCenter and the ESX host on which datastore needs to be created.
  - Select the type of file system (VMFS or NFS).
  - Select the Device from the Name and device selection.

Note: ScaleIO does not push any information to VMware vSphere regarding the type of discs/volume, hence all the volumes will show up as a HDD even though the volume created is a SSD.

  - Select VMFS version.
  - Partition the datastore size and click Finish.

- After creating a datastore, mount the datastore as a new drive on a Guest OS to access the ScaleIO volume.
  - Select Guest OS on the ESX host where the datastore is created.
  - Click on Manage and then Edit.
  - Under New Device, select New hard disk.
Note: Please refer to Performance tunings section for more details on creating a new hard disk for optimized performance.

PERFORMANCE TUNINGS FOR VMWARE ENVIRONMENTS

With the latest release of ScaleIO 2.x, the performance enhancement parameters have been built-in which can be turned on or off with a single command. This reduces the number of manual tasks that were advises in the previous versions of ScaleIO. This section will describe the performance tunings for the following components:

- ScaleIO specific system changes
- VMware specific changes
- Network tunings

Best practice: It is recommended to make these configuration changes before running any application to avoid any restart of the system that may lead to downtime of the application.

SCALEIO SPECIFIC SYSTEM CHANGES

With ScaleIO version 2.x, the ScaleIO specific performance parameters have been built-in. The users may configure high_performance profile which will change the default parameters. For ScaleIO versions prior to 2.x, the users were instructed to modify the SDS and MDM configuration files located at /opt/emc/scaleio/<sds/mdm>/cfg/conf.txt. However, these changes are no longer required. In some cases, the changes made to the ScaleIO 2.x configuration file will no longer effect the ScaleIO system. The difference between the default and high_performance profile is the amount of server resources i.e. CPU and memory that are being consumed. A high_performance profile will always utilize more resources.

- To change the performance profile from default to high_performance, execute the following command:

  scli --set_performance_parameters --all_sds --all_sdc --apply_to_mdm --profile high_performance

- To change the profile back to default, execute the following command:

  scli --set_performance_parameters --all_sds --all_sdc --apply_to_mdm --profile default

- To view the current settings, execute the following command:

  scli --query_performance_parameters
To list the performance parameters for different components, execute the following commands

- **MDM:**
  
  ```bash
  scli --query_performance_parameters --print_all
  ```

- **SDS:**
  
  ```bash
  scli --query_performance_parameters --sds_name <NAME> --print_all
  ```

- **SDC:**
  
  ```bash
  scli --query_performance_parameters --sdc_name --print_all
  ```

### READ RAM CACHE SETTINGS FOR SDS

In ScaleIO version 2.x, the Read RAM Cache is disabled by default. In the previous versions of ScaleIO, it was enabled by default. Recommendations regarding Read RAM Cache:

- It is recommended to leave Read RAM Cache disabled for SSD/Flash pools.
- For HDD pools, Read RAM Cache can improve the performance.
  
  - If the physical node is storage only or in other words, if ScaleIO is used in a 2-layer architecture, it is recommended to use as much of the server DRAM as possible.
  
  - If the physical node is used for both the storage and compute, i.e. ScaleIO is sharing the server with other applications, it is recommended to increase the default size of the DRAM, leaving enough resources for the application.

### VMWARE SPECIFIC CHANGES

This section will describe the changes that are specific to the ESX environment.

### OPTIMIZING THE SCALEIO VIRTUAL MACHINE (SVM)

- For 2-tier implementation, where the SDS and SDC are running on separate servers, it is recommended to assign all vCPUs for the ScaleIO virtual machine (SVM).

- If SDS and SDCs are running on the same node, in general, 4vCPU and 4GB of memory is sufficient, however, for optimal performance it is recommended to increase the vCPU to 8 and memory to 4GB for each SVM.

  **Note:** On servers with >8 cores per socket, it is recommended to use 1 socket and 8 cores. More than 8 cores for the SVM is discouraged and seemed to unnecessarily retain CPU that provided little if any performance boost.

### OPTIMIZING ESX

To get better IOPS, latency, and bandwidth, perform the following operations on all of the ESX nodes.

To improve the I/O concurrency, increase per device queue length (which can be lowered by ESX to 32). Execute the following `esxcli` command:

```bash
esxcli storage core device set -d <DEVICE_ID> -o <QUEUE_LENGTH>
```
The queue length can vary from 32-256 (default 32)

**Example:**

```
esxcli storage core device set -d eui.16bb852c56d3b93e3888003b00000000 -O 256
```

**Note:** It has been noted that this setting is not persistent and cannot be retained after a reboot of the ESX host. To make this setting persistent, it is recommended to add a similar script (persistentLunDepth.sh) as a cron job.

```bash
for disk in `esxcli storage core device list | grep eui | awk -F\" '{ print $2 }' | awk -F\" '{ print $1 }\'`
  do
echo $disk
  `esxcli storage core device set --device=$disk -O 256`
  `esxcli storage core device set --device=$disk --max-queue-depth=16384`
done
```

- Firstly, add the cron job to the root crontab:
  - Edit /var/spool/cron/crontabs/root
  - Add the following line:
    ```
    */5 * * * /full/path/to/script arguments/with/full/path
    ```
  - For example:
    ```
    */5 * * * /vmfs/volumes/5757341a-18001fde-9108-54ab3a16e160/persistentLunDepth.sh
    ```

  **Note:** By default, the root cron is read only. In vi, even as root, one has to force-save-exit `:wq!` in order to save the changes.

- Next, restart the cron job to make the custom script work. However, crontab is also not persistent and the settings are not retained even after the reboot. To re-generate the cron job when ESX server reboots, we need to edit the `/etc/rc.local` file. Hence it is a three-step process
  - Kill the running crond
  - Edit `/etc/rc.local` and restart crond
  - Execute "auto-backup.sh" to make the changes persistent

To kill the running cron job, execute the following commands:

- "cat /var/run/crond.pid"
  - That will print the process number of the running crond, such as 12345
  - Kill <pid>, where pid is the output of the above command

Edit the `/etc/rc.local` by appending the following three commands:

```
/bin/kill $(cat /var/run/crond.pid)
/bin/echo '5 0 * * * /full/path/to/script arguments/with/full/path >>
/var/spool/cron/crontabs/root
/bin/busybox crond
```

Run the command "auto-backup.sh" so that the changes made to `/etc/rc.local` survives a reboot.

**OPTIMIZING VM GUESTS**

To achieve optimal performance from the ScaleIO system it is important to have the right settings for the Guest OS.

**Note:** These settings are for Linux virtual machines only.
• Create a file of any name (suggestion ‘sio_perf.conf’) in the /etc/modprobe.d/ directory with this line:

```bash
options vmw_pvscsi cmd_per_lun=254 ring_pages=32
```

Alternatively, append the following to the kernel boot argument. (for example, on Red Hat Enterprise Linux edit /etc/grub.conf or on Ubuntu edit /boot/grub/grub.cfg):

```bash
vmw_pvscsi.cmd_per_lun=254
vmw_pvscsi.ring_pages=32
```

• Reboot the virtual machine.

• On guest VM, make sure to use paravirtual scsi controller, which does not have queue limitations. To create the **paravirtual scsi controller**, perform the following operations:
  
  – Right click on Guest OS -> Manage -> Edit -> Under the drop down for New Device, select **Paravirtual SCSI controller**

  ![Paravirtual SCSI controller setup](image)

  – For each new hard disk, assign a separate Paravirtual SCSI controller. If using more than 4 volumes, the Paravirtual SCSI controllers need to be shared:
NETWORK TUNINGS

This section describes the method of enabling Jumbo Frames (MTU 9000) for the ESX hosts, SVM and the guest OS.

For **ESX hosts**, perform the following for all the NICs in the ScaleIO system.

**Note**: Prior to activating MTU settings on the logical level, it is required to check if the Jumbo Frames (MTU 9000) is enabled on the physical ports that are connected to the server. In case of MTU size mismatch, there is a possibility of network disconnected and packet drops.

- To change the MTU size on the ESX hosts, type the following command:

  ```
esxconf-vswitch -m 9000 <vSwitch>
```

  ```
esxconf-vmknic -m 9000 <vSwitch>
```

- Create VMKernel with Jumbo Frames support by typing the following commands:

  ```
esxconf-vswitch -d

esxconf-vswitch -A vmkernel# vSwitch#

esxconf-vmknic -a -i <ip address> -n <netmask> -m 9000 <portgroup name>
```

**Note**: If you change Jumbo Frames on an existing vSwitch, the packet size for VMkernel does not change, and therefore, the existing VMkernel must be deleted and a new one should be created

- Execute the following command to check the updated MTU size:

  ```
esxconf-vswitch -l
```

Example:

Figure 7) ESX MTU 9000
For SVM, execute the following command for all the NICs in the ScaleIO system.

- Append `MTU="9000"` to the following files:

  `/etc/sysconfig/network/ifcfg-ethX`

  Restart the SVM network by typing the following command:

  `service network restart`

Example:

```
```

For Guest Operating System, execute the following command for all the NICs in the ScaleIO system.

**Note:** Please check the Network settings for the different guest OS, this section describes the Jumbo Frame settings for RHEL 6.5 only.

- Append `MTU="9000"` to the following files:

  `/etc/sysconfig/network-scripts/ifcfg-ethX`

- Restart the guest OS network by typing the following command:

  `service network restart`

Example:
After enabling Jumbo Frame (MTU 9000) for all ESX nodes, SVM and guest OS, test the network configuration by typing the following command:

Linux: `ping -M do -s 8972 <Destination_IP_Address>`

ESX: `vmkping -d -s 8972 <Destination_IP_Address>`

Example

```
[root@A59T6292 ~]# ping -M do -s 8972 10.10.221.23
PING 10.10.221.23 (10.10.221.23) 8972(9000) bytes of data.
6980 bytes from 10.10.221.23: icmp_seq=1 ttl=64 time=0.636 ms
6980 bytes from 10.10.221.23: icmp_seq=2 ttl=64 time=0.183 ms
6980 bytes from 10.10.221.23: icmp_seq=3 ttl=64 time=0.235 ms
6980 bytes from 10.10.221.23: icmp_seq=4 ttl=64 time=0.125 ms
6980 bytes from 10.10.221.23: icmp_seq=5 ttl=64 time=0.236 ms
6980 bytes from 10.10.221.23: icmp_seq=6 ttl=64 time=0.151 ms
6980 bytes from 10.10.221.23: icmp_seq=7 ttl=64 time=0.236 ms
6980 bytes from 10.10.221.23: icmp_seq=8 ttl=64 time=0.237 ms
6980 bytes from 10.10.221.23: icmp_seq=9 ttl=64 time=0.235 ms
6980 bytes from 10.10.221.23: icmp_seq=10 ttl=64 time=0.119 ms
6980 bytes from 10.10.221.23: icmp_seq=11 ttl=64 time=0.154 ms
6980 bytes from 10.10.221.23: icmp_seq=12 ttl=64 time=0.231 ms
6980 bytes from 10.10.221.23: icmp_seq=13 ttl=64 time=0.159 ms
6980 bytes from 10.10.221.23: icmp_seq=14 ttl=64 time=0.239 ms
6980 bytes from 10.10.221.23: icmp_seq=15 ttl=64 time=0.236 ms
^C
-- 10.10.221.23 ping statistics --
15 packets transmitted, 15 received, 0% packet loss, time 14264ms
rtt min/avg/max/mdev = 0.119/0.227/0.636/0.118 ms
```

**Best Practice:** The overall performance difference between the Standard and the Jumbo Frames is on an average ~10%. MTU setting is an error prone procedure which may cause downtime. Customers should enable MTU 9000 after performing appropriate checks.

**CONCLUSION**

This guide provides all the information that is required to deploy and optimize ScaleIO for VMware environment. The best practices and recommendations provided in this white paper are the techniques and settings that has been proved through the internal testing and the recommendations provided by ScaleIO engineering. After reading this guide, you should be able to deploy and configure ScaleIO for ESX hosts, provision ScaleIO volumes and optimize all the architecture layers i.e. ScaleIO, networking and guest operating system for the optimal use.

Although this guide attempts to generalize many best practices as recommended by ScaleIO, different conditions with different considerations may apply in different environments. For a specific use case, it is recommended to contact a representative from EMC, VMware or ScaleIO.
REFERENCES
ScaleIO User Guide
ScaleIO Installation Guide
ScaleIO ECN community
VMware vSphere 5.5 Documentation Center
VMware vSphere 6.0 Documentation Center
ScaleIO - Network Best Practices Guide
Protecting ScaleIO Data using RecoverPoint for Virtual Machines
Protecting Virtual Machine hosted on ScaleIO Using VMware vSphere Replication
EMC ScaleIO Design Considerations and Best Practices