Storage Profiles

Storage Profiles

To allow flexibility in defining the number of storage disks, roles and usage of these disks, and other storage parameters, you can create and use storage profiles. A storage profile encapsulates the storage requirements for one or more service profiles. LUNs configured in a storage profile can be used as boot LUNs or data LUNs, and can be dedicated to a specific server. You can also specify a local LUN as a boot device. However, LUN resizing is not supported. The introduction of storage profiles allows you to do the following:

- Configure multiple virtual drives and select the physical drives that are used by a virtual drive. You can also configure the storage capacity of a virtual drive.
- Configure the number, type and role of disks in a disk group.
- Associate a storage profile with a service profile.

You can create a storage profile both at an org level and at a service-profile level. A service profile can have a dedicated storage profile as well as a storage profile at an org level.

Cisco Boot Optimized M.2 RAID Controller

Beginning with 4.0(4a), Cisco UCS Manager supports Cisco boot optimized M.2 RAID controller (UCS-M2-HWRAID), which is based on Marvell® 88SE92xx PCIe to SATA 6Gb/s controller. It is supported on the following servers:

- Cisco UCS C225 M6 Server
- Cisco UCS C245 M6 Server
- Cisco UCS C220 M6 Server
- Cisco UCS C240 M6 Server
- Cisco UCS C220 M5 Server
- Cisco UCS C240 M5 Server
- Cisco UCS C480 M5 Server
- Cisco UCS B200 M5 Server
- Cisco UCS B480 M5 Server

The following M.2 drives are managed by the Cisco boot optimized M.2 RAID controller:

- 240GB M.2 6G SATA SSD
- 960GB M.2 6G SATA SSD

The Cisco boot optimized M.2 RAID controller supports only RAID1/JBOD (default - JBOD) mode and only UEFI boot mode.

Limitations of Cisco boot optimized M.2 RAID controller

- Existing LUN migration is not supported.
- **Local Disk Configuration** policy is not supported.
- The number of LUNs that can be created is limited to one because creating a single LUN uses the entire disk capacity.
- LUN is created using the **Local LUN** tab (see) under storage profile and not using the controller definitions.
- You cannot mix different capacity M.2 drives.
- You cannot rename an orphan virtual drive on a blade or a rack server.

Disk Groups and Disk Group Configuration Policies

You can select and configure the disks to be used for storage. A logical collection of these physical disks is called a disk group. Disk groups allow you to organize local disks. The storage controller controls the creation and configuration of disk groups.

A disk group configuration policy defines how a disk group is created and configured. The policy specifies the RAID level to be used for the disk group. It also specifies either a manual or an automatic selection of disks for the disk group, and roles for disks. You can use a disk group policy to manage multiple disk groups. However, a single disk group can be managed only by one disk group policy.

A hot spare is an unused extra disk that can be used by a disk group in the case of failure of a disk in the disk group. Hot spares can be used only in disk groups that support a fault-tolerant RAID level.

Virtual Drives

A disk group can be partitioned into virtual drives. Each virtual drive appears as an individual physical device to the Operating System.

All virtual drives in a disk group must be managed by using a single disk group policy.

Configuration States

Indicates the configuration states of a virtual drive. Virtual drives can have the following configuration states:

- Applying—Creation of the virtual drive is in progress.
- Applied—Creation of the virtual drive is complete, or virtual disk policy changes are configured and applied successfully.
- Failed to apply—Creation, deletion, or renaming of a virtual drive has failed due to errors in the underlying storage subsystem.
- Orphaned—The service profile that contained this virtual drive is deleted or the service profile is no longer associated with a storage profile.

Note Orphaned LUNs cannot be used for booting OS. Although an image can be installed on these LUNs, booting from these drives will fail. To use any specific orphaned LUN, you must reassociate the storage profile, which will return it to the “Equipped” presence state.

- Not in use—The service profile that contained this virtual drive is in the disassociated state.

Deployment States

Indicates the actions that you are performing on virtual drives. Virtual drives can have the following deployment states:

- No action—No pending work items for the virtual drive.
- Creating—Creation of the virtual drive is in progress.
- Deleting—Deletion of the virtual drive is in progress.
- Modifying—Modification of the virtual drive is in progress.
- Apply-Failed—Creation or modification of the virtual drive has failed.

Operability States

Indicates the operating condition of a virtual drive. Virtual drives can have the following operability states:

- Optimal—The virtual drive operating condition is good. All configured drives are online.
- Degraded—The virtual drive operating condition is not optimal. One of the configured drives has failed or is offline.
- Cache-degraded—The virtual drive has been created with a write policy of **write back** mode, but the BBU has failed, or there is no BBU.

Note This state does not occur if you select the **always write back** mode.

- Partially degraded—The operating condition in a RAID 6 virtual drive is not optimal. One of the configured drives has failed or is offline. RAID 6 can tolerate up to two drive failures.
- Offline—The virtual drive is not available to the RAID controller. This is essentially a failed state.
- Unknown—The state of the virtual drive is not known.

Presence States

Indicates the presence of virtual drive components. Virtual drives have the following presence states:

- Equipped—The virtual drive is available.
- Mismatched—A virtual drive deployed state is different from its configured state.
- Missing—Virtual drive is missing.

RAID Levels

The RAID level of a disk group describes how the data is organized on the disk group for the purpose of ensuring availability, redundancy of data, and I/O performance.

The following are features provided by RAID:

- Striping—Segmenting data across multiple physical devices. This improves performance by increasing throughput due to simultaneous device access.
- Mirroring—Writing the same data to multiple devices to accomplish data redundancy.
- Parity—Storing of redundant data on an additional device for the purpose of error correction in the event of device failure. Parity does not provide full redundancy, but it allows for error recovery in some scenarios.
- Spanning—Allows multiple drives to function like a larger one. For example, four 20 GB drives can be combined to appear as a single 80 GB drive.

The supported RAID levels include the following:

- Disable Local Storage—(Supported for PCH SSD Controller Definition) This disk policy mode is to disable the SATA AHCI Controller. This mode can be set only when disks are not present under the SATA AHCI controller. To re-enable this controller and to bring the controller back to its default value (AHCI), you can select No RAID mode or No Local Storage mode.
- No Local Storage—(Supported for PCH SSD Controller Definition) For a diskless server or a SAN only configuration. If you select this option, you cannot associate any service profile which uses this policy with a server that has a local disk.
- RAID 0 Striped— (Supported for PCH SSD Controller Definition) Data is striped across all disks in the array, providing fast throughput. There is no data redundancy, and all data is lost if any disk fails.
- RAID 1 Mirrored— (Supported for PCH SSD Controller Definition) Data is written to two disks, providing complete data redundancy if one disk fails. The maximum array size is equal to the available space on the smaller of the two drives.
- Any Configuration—(Supported for PCH SSD Controller Definition) For a server configuration that carries forward the local disk configuration without any changes.
- No RAID—(Supported for PCH SSD Controller Definition) Allthe disks can be used individually without interdependency similarto JBOD disks. If you choose No RAID and you apply this policyto a server that already has an operating system with RAID storageconfigured, the system does not remove the disk contents.Therefore, there may be no visible differences on the server afteryou apply the No RAID mode. This can lead to a mismatchbetween the RAID configuration in the policy and the actual diskconfiguration shown in the Inventory > Storage tab for the server.To make sure that any previous RAID configuration informationis removed from a disk, apply a scrub policy that removes all diskinformation after you apply the No RAID configuration mode.
- RAID 5 Striped Parity— (Not supported for PCH SSD Controller Definition) Data is striped across all disks in the array. Part of the capacity of each disk stores parity information that can be used to reconstruct data if a disk fails. RAID 5 provides good data throughput for applications with high read request rates.

  RAID 5 distributes parity data blocks among the disks that are part of a RAID-5 group and requires a minimum of three disks.
- RAID 6 Striped Dual Parity— (Not supported for PCH SSD Controller Definition) Data is striped across all disks in the array and two sets of parity data are used to provide protection against failure of up to two physical disks. In each row of data blocks, two sets of parity data are stored.

  Other than addition of a second parity block, RAID 6 is identical to RAID 5 . A minimum of four disks are required for RAID 6.
- RAID 10 Mirrored and Striped— (Not supported for PCH SSD Controller Definition) RAID 10 uses mirrored pairs of disks to provide complete data redundancy and high throughput rates through block-level striping. RAID 10 is mirroring without parity and block-level striping. A minimum of four disks are required for RAID 10.
- RAID 50 Striped Parity and Striped— (Not supported for PCH SSD Controller Definition) Data is striped across multiple striped parity disk sets to provide high throughput and multiple disk failure tolerance.
- RAID 60 Striped Dual Parity and Striped— (Not supported for PCH SSD Controller Definition) Data is striped across multiple striped dual parity disk sets to provide high throughput and greater disk failure tolerance.

Note Some Cisco UCS servers require a license for certainRAID configuration options. When Cisco UCS Managerassociates a service profile containing this local diskpolicy with a server, Cisco UCS Manager verifies thatthe selected RAID option is properly licensed. If thereare issues, Cisco UCS Manager displays a configurationerror during the service profile association. For RAID license information for a specific Cisco UCSserver, see the Hardware Installation Guide for that server.

Automatic Disk Selection

When you specify a disk group configuration, and do not specify the local disks in it, Cisco UCS Manager determines the disks to be used based on the criteria specified in the disk group configuration policy. Cisco UCS Manager can make this selection of disks in multiple ways.

When all qualifiers match for a set of disks, then disks are selected sequentially according to their slot number. Regular disks and dedicated hot spares are selected by using the lowest numbered slot.

The following is the disk selection process:

1. Iterate over all local LUNs that require the creation of a new virtual drive. Iteration is based on the following criteria, in order:
   a. Disk type
   b. Minimum disk size from highest to lowest
   c. Space required from highest to lowest
   d. Disk group qualifier name, in alphabetical order
   e. Local LUN name, in alphabetical order
2. Select regular disks depending on the minimum number of disks and minimum disk size. Disks are selected sequentially starting from the lowest numbered disk slot that satisfies the search criteria.

Note If you specify **Any** as the type of drive, the first available drive is selected. After this drive is selected, subsequent drives will be of a compatible type. For example, if the first drive was SATA, all subsequent drives would be SATA. Cisco UCS Manager Release 2.5 supports only SATA and SAS.

Cisco UCS Manager Release 2.5 does not support RAID migration.

3. Select dedicated hot spares by using the same method as normal disks. Disks are only selected if they are in an **Unconfigured Good** state.
4. If a provisioned LUN has the same disk group policy as a deployed virtual drive, then try to deploy the new virtual drive in the same disk group. Otherwise, try to find new disks for deployment.

Supported LUN Modifications

Some modifications that are made to the LUN configuration when LUNs are already deployed on an associated server are supported.

The following are the types of modifications that can be performed:

- Creation of a new virtual drive.
- Deletion of an existing virtual drive, which is in the orphaned state.
- Non-disruptive changes to an existing virtual drive. These changes can be made on an existing virtual drive without loss of data, and without performance degradation:
  - Policy changes. For example, changing the write cache policy.
  - Modification of boot parameters

The removal of a LUN will cause a warning to be displayed. Ensure that you take action to avoid loss of data.

Unsupported LUN Modifications

Some modifications to existing LUNs are not possible without destroying the original virtual drive and creating a new one. All data is lost in these types of modification, and these modifications are not supported.

Disruptive modifications to an existing virtual drive are not supported. The following are unsupported disruptive changes:

- Any supported RAID level change that can be handled through reconstruction. For example, RAID0 to RAID1.
- Increasing the size of a virtual drive through reconstruction.
- Addition and removal of disks through reconstruction.

Destructive modifications are also not supported. The following are unsupported destructive modifications:

- RAID-level changes that do not support reconstruction. For example, RAID5 to RAID1.
- Shrinking the size of a virtual drive.
- RAID-level changes that support reconstruction, but where there are other virtual drives present on the same drive group.
- Disk removal when there is not enough space left on the disk group to accommodate the virtual drive.
- Explicit change in the set of disks used by the virtual drive.

Disk Insertion Handling

When the following sequence of events takes place:

1. The LUN is created in one of the following ways:

a. You specify the slot specifically by using a local disk reference

b. The system selects the slot based on criteria specified by you

2. The LUN is successfully deployed, which means that a virtual drive is created, which uses the slot.
3. You remove a disk from the slot, possibly because the disk failed.
4. You insert a new working disk into the same slot.

The following scenarios are possible:

- Non-Redundant Virtual Drives, on page 8
- Redundant Virtual Drives with No Hot Spare Drives, on page 8
- Redundant Virtual Drives with Hot Spare Drives, on page 8
- Replacing Hot Spare Drives, on page 9
- Inserting Physical Drives into Unused Slots, on page 9

Non-Redundant Virtual Drives

For non-redundant virtual drives (RAID 0), when a physical drive is removed, the state of the virtual drive is **Inoperable**. When a new working drive is inserted, the new physical drive goes to an **Unconfigured Good** state.

For non-redundant virtual drives, there is no way to recover the virtual drive. You must delete the virtual drive and re-create it.

Redundant Virtual Drives with No Hot Spare Drives

For redundant virtual drives (RAID 1, RAID 5, RAID 6, RAID 10, RAID 50, RAID 60) with no hot spare drives assigned, virtual drive mismatch, virtual drive member missing, and local disk missing faults appear until you insert a working physical drive into the same slot from which the old physical drive was removed.

If the physical drive size is greater than or equal to that of the old drive, the storage controller automatically uses the new drive for the virtual drive. The new drive goes into the **Rebuilding** state. After rebuild is complete, the virtual drive goes back into the **Online** state.

Redundant Virtual Drives with Hot Spare Drives

For redundant virtual drives (RAID 1, RAID 5, RAID 6, RAID 10, RAID 50, RAID 60) with hot spare drives assigned, when a drive fails, or when you remove a drive, the dedicated hot spare drive, if available, goes into the **Rebuilding** state with the virtual drive in the **Degraded** state. After rebuilding is complete, that drive goes to the **Online** state.

Cisco UCSM raises a disk missing and virtual drive mismatch fault because although the virtual drive is operational, it does not match the physical configuration that Cisco UCSM expects.

if you insert a new disk in the slot with the disk missing, automatic copy back starts from the earlier hot spare disk to the newly inserted disk. After copy back, the hot spare disk is restored. In this state all faults are cleared.

If automatic copy back does not start, and the newly inserted disk remains in the **Unconfigured Good**, **JBOD**, or **Foreign Configuration** state, remove the new disk from the slot, reinsert the earlier hot spare disk into the slot, and import foreign configuration. This initiates the rebuilding process and the drive state becomes **Online**. Now, insert the new disk in the hot spare slot and mark it as hot spare to match it exactly with the information available in Cisco UCSM.

Replacing Hot Spare Drives

If a hot spare drive is replaced, the new hot spare drive will go to the **Unconfigured Good**, **Unconfigured Bad**, **JBOD**, or **Foreign Configuration** state.

Cisco UCSM will raise a virtual drive mismatch or virtual drive member mismatch fault because the hot spare drive is in a state different from the state configured in Cisco UCSM.

You must manually clear the fault. To do this, you must perform the following actions:

1. Clear the state on the newly inserted drive to **Unconfigured Good**.
2. Configure the newly inserted drive as a hot spare drive to match what is expected by Cisco UCSM.

Inserting Physical Drives into Unused Slots

If you insert new physical drives into unused slots, neither the storage controller nor Cisco UCSM will make use of the new drive even if the drive is in the **Unconfigured Good** state and there are virtual drives that are missing good physical drives.

The drive will simply go into the **Unconfigured Good** state. To make use of the new drive, you will need to modify or create LUNs to reference the newly inserted drive.

Virtual Drive Naming

When you use UCSM to create a virtual drive, UCSM assigns a unique ID that can be used to reliably identify the virtual drive for further operations. UCSM also provides the flexibility to provide a name to the virtual drive at the time of service profile association. Any virtual drive without a service profile or a server reference is marked as an orphan virtual drive.

In addition to a unique ID, a name is assigned to the drive. Names can be assigned in two ways:

- When configuring a virtual drive, you can explicitly assign a name that can be referenced in storage profiles.
- If you have not preprovisioned a name for the virtual drive, UCSM generates a unique name for the virtual drive.

You can rename an orphan virtual drive drives on a blade or a rack server that are not referenced by any service profile or server.

Note The renaming an orphan virtual drive is not supported on the Cisco boot optimized M.2 Raid controller (UCS-M2-HWRAID).

LUN Dereferencing

A LUN is dereferenced when it is no longer used by any service profile. This can occur as part of the following scenarios:

- The LUN is no longer referenced from the storage profile
- The storage profile is no longer referenced from the service profile
- The server is disassociated from the service profile
- The server is decommissioned

When the LUN is no longer referenced, but the server is still associated, re-association occurs.

When the service profile that contained the LUN is disassociated, the LUN state is changed to **Not in use**.

When the service profile that contained the LUN is deleted, the LUN state is changed to **Orphaned**.

Controller Constraints and Limitations

- The following table provides the maximum supported virtual drives for servers:

Servers/Storage Controllers	Maximum Virtual Drives
UCSB-MRAID12G-M6	16
UCSC-C220-M6, UCSC-C240-M6, UCSC-C225-M6, UCSC-C245-M6	32
UCSC-C240-M5, UCSC-C480-M5	32
UCS-S3260-M5, UCSC-C3X60-M4, UCSC-C3K-M4	64
UCSC-C240-M4, UCSC-C240-M3, UCSC-C24-M3	24
UCSB-MRAID12G	16
UCS-M2-HWRAID	2
For all other servers.	18

Note

- Storage controllers support the check max feature.
- When servers with multiple storage controllers are managed by the same storage profile, the maximum virtual drives are limited to the maximum value supported by the server.
- UCS-MSTOR-M2 and UCS-MSTOR-SD controllers are not supported on M6 servers.

- The following table shows the maximum supported storage controllers for the Cisco UCS C225 M6 Server is as follows:

Table 1: Maximum Supported Storage Controllers: Cisco UCS C225 M6 Server

Servers/Storage Controllers	Maximum Virtual Drives
Cisco UCS C225 M6 Server	• UCS C225 M6SX and UCS C245 M6SX in C225-SFF (10 front SAS/SATA drives) • 2 M.2 2280 Drives on UCS-M2-HWRAID • Direct Attached NVMe drives (10 NVMe drives in the front)

- The following table shows the maximum supported storage controllers for the Cisco UCS C245 M6 Server.

Table 2: Maximum Supported Storage Controllers: Cisco UCS C245 M6 Server

Servers/Storage Controllers	Maximum Virtual Drives
Cisco UCS C245 M6 Server	• Dual UCS C245 M6SX 16 SAS/SATA HDD • UCS C245 M6SX Plus 28 SAS/SATA HDD • 2 M.2 2280 Drives on UCS-M2-HWRAID • Directly Attached NVMe on rear risers(up to 4 NVMe SSD)

- The following table shows supported controller and driver configurations for the storage drives on the Cisco UCS C225 M6 Server.

	Storage Controller	Front Cage Support			Single CPU
	Number of SFF HDD/SAS SSD	Number of NVMe Drives	NVMe Drive connectivity		

	Storage Controller	Front Cage Support			Single CPU
C225-SFF (10front)	UCS C225 M6SX or UCS C245 M6SX in C225-SFF	Up to 10	Up to 4	PCIe Gen4 x2	10 SAS
C225-NVMe (10 front)	Direct Attach to CPU	Not Supported	Up to 10	PCIe Gen4 x2	10 NVMe

- The following table provides the maximum supported storage drives for the Cisco UCS C245 M6 Server :

Servers/Storage Controllers	Maximum Virtual Drives
UCS Cisco UCS C245 M6 x 28 HDD/SDD backplane Up to 24 x 2.5-inch 12-Gbps Front load HDDs or SSDs and 4 rear hot-swappable 2.5-inch NVMe drives, Total of 8 (4 front +4 rear) NVMe SSDs	Dual UCS C245 M6SX 12 SAS3 drives (12 per controller)
Cisco UCS C245 M6 x 24 HDD/SDD backplane	UCS C245 M6SX Plus 24 SAS3 drives
UCS-M2-HWRAID M.2 modules with RAID 1 support	1
Only UCS-M2-HWRAID M.2 module support on 4 Front NVMe and 4 Rear NVMe drives	1

- In Cisco UCS Manager Release 2.2(4), blade servers do not support drives with a block size of 4K, but rack-mount servers support such drives. If a drive with a block size of 4K is inserted into a blade server, discovery fails and the following error message appears:`Unable to get Scsi Device Information from the system`.
- In Cisco UCS Manager Release 3.1(2) and later releases, RAID Controller that does not support Out of band inventory (OOB) in C240 M4, M5, and M6 servers, display Operability as `NA` and Drive State as `Unknown`.

Configuring Storage Profiles

Configuring a Disk Group Policy

You can choose to configure a disk group policy through automatic or manual disk selection. Configuring a disk group involves the following:

1. Setting the RAID Level, on page 13

2. Automatically Configuring Disks in a Disk Group, on page 14 or Manually Configuring Disks in a Disk Group, on page 16

Note If you have a setup with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then you can configure the disk only manually.

3. Configuring Virtual Drive Properties, on page 18

Guidelines for sucessful creation of LUNs using JBOD or UG drives:

1. When the drive state is UG and is in the disk group policy, and if Use JBOD is set to:
   - Yes—Both JBOD and UG drives can be used based on the drive slot ordering.
   - No—Only UG drives can be used.
2. When drive state is JBOD and is in the disk group policy, and if Use JBOD is set to:
   - Yes—Both JBOD and UG drives can be used based on the drive slot ordering.
   - No—Only UG drives can be used.
3. When the drive state is JBOD or UG and is in the disk group policy, and if Use JBOD is set to:
   - Yes—Both JBOD and UG drives can be used.
   - No—Only UG drives can be used.

Note The UCS Manager disk selection is based on the sequential slot number, irrespective of the drive state.

Setting the RAID Level

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org# **create disk-group-config-policy** *disk-group-name*	Creates a disk group configuration policy with the specified name and enters disk group configuration policy mode.
Step 3	UCS-A /org/disk-group-config-policy* # **set raid-level** *raid-level*	Specifies the RAID level for the disk group configuration policy. The RAID levels that you can specify are: • raid-0-striped

	Command or Action	Purpose
		• raid-1-mirrored • raid-10-mirrored-and-striped • raid-5-striped-parity • raid-6-striped-dual-parity • raid-50-striped-parity-and-striped • raid-60-striped-dual-parity-and-striped **Note** The Cisco boot optimized M.2 RAID controller (UCS-M2-HWRAID) supports only RAID1.
Step 4	UCS-A /org/disk-group-config-policy* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to set the RAID level for a disk group configuration policy.

```
UCS-A# scope org
UCS-A /org # create disk-group-config-policy raid5policy
UCS-A /org/disk-group-config-policy* # set raid-level raid-5-striped-parity
UCS-A /org/disk-group-config-policy* # commit-buffer
```

What to do next

Automatically or manually configure disks as part of the disk group configuration policy.

Automatically Configuring Disks in a Disk Group

You can allow UCSM to automatically select and configure disks in a disk group.

When you create a disk group with RAID 1 policy and configure four disks for it, a RAID1E configuration is created internally by the storage controller.

If you have a set-up with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then go to Manually Configuring Disks in a Disk Group, on page 16.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org# **enter disk-group-config-policy** *disk-group-name*	Enters disk group configuration policy mode for the specified disk group name.

	Command or Action	Purpose
Step 3	UCS-A /org/disk-group-config-policy* # **enter disk-group-qual**	Enters disk group qualification mode. In this mode, UCSM automatically configures disks as part of the specified disk group.
Step 4	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set drive-type** *drive-type*	Specifies the drive type for the disk group. You can select: • HDD • SSD • Unspecified **Note** If you specify **Unspecified** as the type of drive, the first available drive is selected. After this drive is selected, subsequent drives will be of a compatible type. For example, if the first was SSD, all subsequent drives would be SSD.
Step 5	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set min-drive-size** *drive-size*	Specifies the minimum drive size for the disk group. Only disks that match this criteria will be available for selection. The range for minimum drive size is from 0 to10240 GB. You can also set the minimum drive size as **Unspecified**. If you set the minimum drive size as **Unspecified**, drives of all sizes will be available for selection.
Step 6	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set num-ded-hot-spares** *hot-spare-num*	Specifies the number of dedicated hot spares for the disk group. The range for dedicated hot spares is from 0 to 24 hot spares. You can also set the number of dedicated hot spares as **Unspecified**. If you set the number of dedicated hot spares as **Unspecified**, the hot spares will be selected according to the disk selection process.
Step 7	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set num-drives** *drive-num*	Specifies the number of drives for the disk group. The range for drives is from 0 to 24 drives for Cisco UCS C240, C220, C24, and C22 servers. For all other servers, the limit is 16 drives per server.. You can also set the number of drives as **Unspecified**. If you set the number of drives as **Unspecified**, the number of drives will be selected according to the disk selection process.

	Command or Action	Purpose
Step 8	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set num-glob-hot-spares** *hot-spare-num*	Specifies the number of global hot spares for the disk group. The range for global hot spares is from 0 to 24 hot spares. You can also set the number of global hot spares as **Unspecified**. If you set the number of global hot spares as **Unspecified**, the global hot spares will be selected according to the disk selection process.
Step 9	UCS-A /org/disk-group-config-policy/disk-group-qual* # **set use-remaining-disks** {**no** | **yes**}	Specifies whether the remaining disks in the disk group policy should be used or not. The default value for this command is **no**.
Step 10	UCS-A /org/disk-group-config-policy/disk-group-qual* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to automatically configure disks for a disk group configuration policy.

```
UCS-A# scope org
UCS-A /org # enter disk-group-config-policy raid5policy
UCS-A /org/disk-group-config-policy* # enter disk-group-qual
UCS-A /org/disk-group-config-policy/disk-group-qual* # set drive-type hdd
UCS-A /org/disk-group-config-policy/disk-group-qual* # set min-drive-size 1000
UCS-A /org/disk-group-config-policy/disk-group-qual* # set num-ded-hot-spares 2
UCS-A /org/disk-group-config-policy/disk-group-qual* # set num-drives 7
UCS-A /org/disk-group-config-policy/disk-group-qual* # set num-glob-hot-spares 2
UCS-A /org/disk-group-config-policy/disk-group-qual* # set use-remaining-disks no
UCS-A /org/disk-group-config-policy/disk-group-qual* # commit-buffer
```

```
UCS-A# scope org
UCS-A /org # enter disk-group-config-policy raid5policy
UCS-A /org/disk-group-config-policy* # enter disk-group-qual
UCS-A /org/disk-group-config-policy/disk-group-qual* # set drive-type ssd
UCS-A /org/disk-group-config-policy/disk-group-qual* # set min-drive-size 1000
UCS-A /org/disk-group-config-policy/disk-group-qual* # set num-ded-hot-spares 2
UCS-A /org/disk-group-config-policy/disk-group-qual* # set num-drives 7
UCS-A /org/disk-group-config-policy/disk-group-qual* # commit-buffer
```

What to do next

Configure Virtual Drives.

Manually Configuring Disks in a Disk Group

You can manually configure disks for a disk group.

When you create a disk group with RAID 1 policy and configure four disks for it, a RAID 1E configuration is created internally by the storage controller.

Cisco boot optimized M.2 RAID controller (UCS-M2-HWRAID) supports only RAID1.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org# **enter disk-group-config-policy** *disk-group-name*	Enters disk group configuration policy mode for the specified disk group name.
Step 3	UCS-A /org/disk-group-config-policy* # **create local-disk-config-ref** *slot-num*	Creates a local disk configuration reference for the specified slot and enters local disk configuration reference mode. **Note** M.2 drives typically have Slot IDs = 253, 254.
Step 4	UCS-A /org/disk-group-config-policy/local-disk-config-ref *# **set role** *role*	Specifies the role of the local disk in the disk group. You can select: • ded-hot-spare: Dedicated hot spare • glob-hot-spare: Global hot spare • normal **Note** If you have a setup with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then select normal. Selecting any other value results in configuration error.
Step 5	UCS-A /org/disk-group-config-policy/local-disk-config-ref *# **set span-id** *span-id*	Specifies the ID of the span group to which the disk belongs. Disks belonging to a single span group can be treated as a single disk with a larger size. The values range from 0 to 8. For RAID-10, RAID-50, and RAID-60, minimum 2 spans are required and maximum 8 spans are supported. You can also set the Span ID as **Unspecified** when spanning information is not required.

	Command or Action	Purpose
		Note • In Cisco UCS Release 2.5, you can have a maximum of 4 span groups. • If you have a setup with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then this field does not apply. Select the Span ID field as **Unspecified**. Selecting any value results in configuration error.
Step 6	UCS-A /org/disk-group-config-policy/local-disk-config-ref *# **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to manually configure disks for a disk group configuration policy.

```
UCS-A# scope org
UCS-A /org # enter disk-group-config-policy raid5policy
UCS-A /org/disk-group-config-policy* # create local-disk-config-ref 1
UCS-A /org/disk-group-config-policy/local-disk-config-ref *# set role ded-hot-spare
UCS-A /org/disk-group-config-policy/local-disk-config-ref* # set span-id 1
UCS-A /org/disk-group-config-policy/local-disk-config-ref *# commit-buffer
```

What to do next

Configure Virtual Drive Properties.

Configuring Virtual Drive Properties

All virtual drives in a disk group must be managed by using a single disk group policy.

If you try to associate to a server that does not support these properties, a configuration error will be generated.

Only the following storage controllers support these properties:

- LSI 6G MegaRAID SAS 9266-8i
- LSI 6G MegaRAID SAS 9271-8i
- LSI 6G MegaRAID 9265-8i
- LSI MegaRAID SAS 2208 ROMB
- LSI MegaRAID SAS 9361-8i

For the LSI MegaRAID SAS 2208 ROMB controller, these properties are supported only in the B420-M3 blade server. For the other controllers, these properties are supported in multiple rack servers.

Note If you have a setup with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then:

- You can create only one virtual drive
- For **strip-size**, select **64KB** or **32KB**. Selecting any other value results in configuration error.
- For **access-policy**, **read-policy**, **write-cache-policy**, **io-policy**, and **drive-cache**, select **platform-default**. Selecting any other value results in configuration error.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org# **scope disk-group-config-policy** *disk-group-name*	Enters disk group configuration policy mode for the specified disk group name.
Step 3	UCS-A /org/disk-group-config-policy* # **create virtual-drive-def**	Creates a virtual drive definition and enters the virtual drive definition mode.
Step 4	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set access-policy** *policy-type*	Specifies the access policy. This can be one of the following: • blocked • platform-default • read-only: • read-write
Step 5	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set drive-cache** *state*	Specifies the state of the drive cache. This can be one of the following: • enable • disable • no-change • platform-default **Important** In Cisco UCS Release 2.5, the drive cache state cannot be changed. It will remain as **platform-default**, irrespective of the drive cache state that you select.

	Command or Action	Purpose
Step 6	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set io-policy** *policy-type*	Specifies the I/O policy. This can be one of the following: • cached • direct • platform-default
Step 7	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set read-policy** *policy-type*	Specifies the read policy. This can be one of the following: • normal • platform-default • read-ahead
Step 8	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set strip-size** *strip-size*	Specifies the strip size. This can be one of the following: • 64 KB • 128 KB • 256 KB • 512 KB • 1024 KB • platform-default
Step 9	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set write-cache-policy** *policy-type*	Specifies the write-cache-policy. This can be one of the following: • always-write-back • platform-default • write-back-good-bbu • write-through
Step 10	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **commit-buffer**	Commits the transaction to the system configuration.
Step 11	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **show**	Displays the configured virtual drive properties.

Example

This example shows how to configure virtual disk properties:

```
UCS-A# scope org
UCS-A /org # scope disk-group-config-policy raid0policy
UCS-A /org/disk-group-config-policy # create virtual-drive-def
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set access-policy read-write
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set drive-cache enable
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set io-policy cached
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set read-policy normal
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set strip-size 1024
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set write-cache-policy write-through
UCS-A /org/disk-group-config-policy/virtual-drive-def* # commit-buffer
UCS-A /org/disk-group-config-policy/virtual-drive-def # show

Virtual Drive Def:
    Strip Size (KB): 1024KB
    Access Policy: Read Write
    Read Policy: Normal
    Configured Write Cache Policy: Write Through
    IO Policy: Cached
    Drive Cache: Enable
UCS-A /org/disk-group-config-policy/virtual-drive-def #
```

What to do next

Create a Storage Profile

Creating a Storage Profile

You can create a storage profile at the org level and at the service-profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **create storage-profile** *storage-profile-name*	Creates a storage profile with the specified name at the org level and enters storage-profile configuration mode.
Step 3	UCS-A /org/storage-profile* # **commit-buffer**	Commits the transaction to the system configuration.
Step 4	(Optional) UCS-A /org* # **enter service-profile** *service-profile-name*	Enters the specified service profile.
Step 5	(Optional) UCS-A /org/service-profile* # **create storage-profile-def**	Creates a storage profile at the service-profile level.

	Command or Action	Purpose
Step 6	UCS-A /org/service-profile/storage-profile-def* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to create a storage profile at the org level.

```
UCS-A# scope org
UCS-A /org # create storage-profile stp2
UCS-A /org/storage-profile* # commit-buffer
```

This example shows how to create a storage profile at the service-profile level.

```
UCS-A# scope org
UCS-A /org* # enter service-profile sp1
UCS-A /org/service-profile* # create storage-profile-def
UCS-A /org/service-profile/storage-profile-def* # commit-buffer
```

What to do next

Create Local LUNs

Deleting a Storage Profile

You can delete a storage profile that was created at the org level or at the service-profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **delete storage-profile** *storage-profile-name*	Deletes the storage profile with the specified name at the org level.
Step 3	(Optional) UCS-A /org # **scope service-profile** *service-profile-name*	Enters the specified service profile.
Step 4	(Optional) UCS-A /org/service-profile # **delete storage-profile-def**	Deletes the dedicated storage profile at the service-profile level.

Example

This example shows how to delete a storage profile at the org level.

```
UCS-A # scope org
UCS-A /org #  delete storage-profile stor1
```

This example shows how to delete a storage profile at the service-profile level.

```
UCS-A # scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # delete storage-profile-def
```

Local LUNs

Creating Local LUNs

You can create local LUNs within a storage profile at the org level and within a dedicated storage profile at the service-profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **enter storage-profile** *storage-profile-name*	Enters storage-profile mode for the specified storage profile.
Step 3	UCS-A /org/storage-profile* # **create local-lun** *lun-name*	Creates a local LUN with the specified name.
Step 4	UCS-A /org/storage-profile/local-lun* # **set auto-deploy** {**auto-deploy** | **no-auto-deploy**}	Specifies whether the LUN should be auto-deployed or not.
Step 5	UCS-A /org/storage-profile/local-lun* # **set disk-policy-name** *disk-policy-name*	Specifies the name of the disk policy name for this LUN.
Step 6	UCS-A /org/storage-profile/local-lun* # **set order** *order-num*	Specifies the order of this LUN. The order can range from 1 to 64. You can also specify that the order should be **lowest-available** for the system to automatically assign the lowest available order to the LUN. Multiple LUNs referenced by a storage profile must have unique names and unique orders.
Step 7	UCS-A /org/storage-profile/local-lun* # **set expand-to-avail** {**no** | **yes**}	Specifies whether the LUN should be expanded to the entire available disk group. For each service profile, only one LUN can be configured to use this option.
Step 8	UCS-A /org/storage-profile/local-lun* # **set size** *size*	Specifies the size of this LUN in GB.

	Command or Action	Purpose
		Note In a setup with the Cisco boot optimized M.2 Raid controller, you do not have to specify a size. The system uses the full disk capacity to create the LUN, irrespective of the size specified. **Note** You do not need to specify a LUN size while claiming an orphaned LUN.
Step 9	UCS-A /org/storage-profile/local-lun* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to configure a local LUN within a storage profile at the org level.

```
UCS-A# scope org
UCS-A /org # enter storage-profile stp2
UCS-A /org/storage-profile* # create local-lun lun2
UCS-A /org/storage-profile/local-lun* # set disk-policy-name dpn2
UCS-A /org/storage-profile/local-lun* # set order 2
UCS-A /org/storage-profile/local-lun* # set size 1000
UCS-A /org/storage-profile/local-lun* # commit-buffer
```

```
UCS-A# scope org
UCS-A /org # enter storage-profile stp2
UCS-A /org/storage-profile* # create local-lun lun2
UCS-A /org/storage-profile/local-lun* # set auto-deploy no-auto-deploy
UCS-A /org/storage-profile/local-lun* # set disk-policy-name dpn2
UCS-A /org/storage-profile/local-lun* # set expand-to-avail yes
UCS-A /org/storage-profile/local-lun* # set size 1000
UCS-A /org/storage-profile/local-lun* # commit-buffer
```

This example shows how to configure a local LUN within a dedicated storage profile at the service-profile level.

```
UCS-A# scope org
UCS-A /org* # enter service-profile stp1
UCS-A /org/service-profile* # enter storage-profile-def
UCS-A /org/service-profile/storage-profile-def # create local-lun lun1
UCS-A /org/service-profile/storage-profile-def/local-lun* # set disk-policy-name dpn1
UCS-A /org/service-profile/storage-profile-def/local-lun* # set order 1
UCS-A /org/service-profile/storage-profile-def/local-lun* # set size 1000
UCS-A /org/service-profile/storage-profile-def/local-lun* # commit-buffer
```

```
UCS-A# scope org
UCS-A /org # enter service-profile sp1
UCS-A /org/service-profile* # enter storage-profile-def
UCS-A /org/service-profile/storage-profile-def # create local-lun lun1
UCS-A /org/service-profile/storage-profile-def/local-lun* # set auto-deploy no-auto-deploy
UCS-A /org/service-profile/storage-profile-def/local-lun* # set disk-policy-name dpn1
UCS-A /org/service-profile/storage-profile-def/local-lun* # set expand-to-avail yes
```

```
UCS-A /org/service-profile/storage-profile-def/local-lun* # set size 1000
UCS-A /org/service-profile/storage-profile-def/local-lun* # commit-buffer
```

What to do next

Associate a Storage Profile with a Service Profile

Reordering Local LUNs In a Storage Profile

You can use the **set order** command to change the local LUN visibility order to the server. This operation will reboot the server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **enter storage-profile** *storage-profile-name*	Enters storage-profile mode for the specified storage profile.
Step 3	UCS-A /org/storage-profile # **enter local-lun** *lun-name*	Enters local-lun mode for the specified local LUN.
Step 4	UCS-A /org/storage-profile/local-lun* # **set disk-policy-name** *disk-policy-name*	Specifies the name of the disk policy name for this LUN.
Step 5	UCS-A /org/storage-profile/local-lun # **set order** *order-num*	Specifies the order of this LUN. The order can range from 1 to 64. You can also specify that the order should be **lowest-available** for the system to automatically assign the lowest available order to the LUN.
Step 6	UCS-A /org/storage-profile/local-lun* # **set size** *size*	Specifies the size of this LUN in GB.
Step 7	UCS-A /org/storage-profile/local-lun # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to reorder a local LUN within a storage profile at the org level.

```
UCS-A# scope org
UCS-A /org # enter storage-profile stp1
UCS-A /org/storage-profile* # enter local-lun lun1
UCS-A /org/storage-profile/local-lun* # set disk-policy-name dpn1
UCS-A /org/storage-profile/local-lun* # set order 1
UCS-A /org/storage-profile/local-lun* # set size 10
UCS-A /org/storage-profile/local-lun* # exit
UCS-A /org/storage-profile* # enter local-lun lun2
```

```
UCS-A /org/storage-profile/local-lun* # set disk-policy-name dpn2
UCS-A /org/storage-profile/local-lun* # set order 2
UCS-A /org/storage-profile/local-lun* # set size 10
UCS-A /org/storage-profile/local-lun* # exit
UCS-A /org/storage-profile* # commit-buffer

UCS-A /org/storage-profile # show configuration

 enter storage-profile stp1
     enter local-lun lun1
         set auto-deploy auto-deploy
         set disk-policy-name dpn1
         set order 1
         set size 10
     exit
     enter local-lun lun2
         set auto-deploy auto-deploy
         set disk-policy-name dpn2
         set order 2
         set size 10
     exit
     set descr ""
 exit

UCS-A /org/storage-profile # enter local-lun lun1
UCS-A /org/storage-profile/local-lun # set order 2
UCS-A /org/storage-profile/local-lun* # exit
UCS-A /org/storage-profile* # enter local-lun lun2
UCS-A /org/storage-profile/local-lun* # set order 1
UCS-A /org/storage-profile/local-lun* # exit
UCS-A /org/storage-profile* # commit-buffer
UCS-A /org/storage-profile # show configuration

 enter storage-profile stp1
     enter local-lun lun1
         set auto-deploy auto-deploy
         set disk-policy-name dpn1
         set order 2
         set size 10
     exit
     enter local-lun lun2
         set auto-deploy auto-deploy
         set disk-policy-name dpn2
         set order 1
         set size 10
     exit
     set descr ""
 exit
```

Deleting Local LUNs In a Storage Profile

When a LUN is deleted, the corresponding virtual drive is marked as orphan after the virtual drive reference is removed from the server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **enter storage-profile** *storage-profile-name*	Enters storage-profile mode for the specified storage profile.
Step 3	(Optional) UCS-A /org/storage-profile* # **show local-lun**	Displays the local LUNs in the specified storage profile.
Step 4	UCS-A /org/storage-profile* # **delete local-lun** *lun-name*	Deletes the specified LUN.
Step 5	UCS-A /org/storage-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to delete a LUN in a storage profile.

```
UCS-A # scope org
UCS-A /org # enter storage-profile stp2
UCS-A /org/storage-profile # show local-lun


Local SCSI LUN:

    LUN Name   Size (GB)   Order            Disk Policy Name Auto Deploy

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

    luna       1           2                raid0            Auto Deploy

    lunb       1           1                raid0            Auto Deploy

UCS-A /org/storage-profile # delete local-lun luna
UCS-A /org/storage-profile* # commit-buffer
UCS-A /org/storage-profile* # show local-lun


Local SCSI LUN:

    LUN Name   Size (GB)   Order            Disk Policy Name Auto Deploy

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

    lunb       1           1                raid0            Auto Deploy
```

LUN Set

LUN Set

Beginning with release 4.0(2a), Cisco UCS Manager provides the ability to configure a range of disk slots into individual RAID0 LUNs using LUN Set option.

The following guidelines should be considered while creating a LUN Set:

- Only SSD and HDD types of disks are allowed.
- Upto 60 disks are allowed in one range.
- You cannot add the same set of disks in range under two different LUN Set configurations.
- If a disk is set in the disk slot range of LUN Set, then you cannot configure the same disk set in Local LUN configuration under the same storage policy. Similarly, if a disk is set in Local LUN configuration, then you cannot use the same disk in the disk slot range of LUN Set.
- The server, in which the LUN Set is configured, should support OOB storage operations.
- You cannot configure a Local Disk Policy along with a Storage Policy in the same Service Profile.
- You cannot have the same name for a Local LUN and LUN Set.
- In S-series server PCH controllers, slots 201 and 202 do not support LUN Set.

Limitations of LUN Set

Cisco UCS Manager has the following limitations with LUN Set:

- You cannot claim orphaned Local LUNs into a LUN Set.
- Once created, you cannot modify a LUN Set. You should delete and create a new LUN Set with desired parameters.
- OS boot is not supported from LUN Set.

Creating a LUN Set

You can create a LUN Set within a storage profile at the org level and within a dedicated storage profile at the service-profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **enter storage-profile** *storage-profile-name*	Enters storage-profile mode for the specified storage profile.
Step 3	UCS-A /org/storage-profile* # **create lun-set** *lun-set-name*	Creates a LUN Set with the specified name.

	Command or Action	Purpose
Step 4	UCS-A /org/storage-profile/lun-set* # **set disk-slot-range** *disk-slot-range*	Specifies the slot range for the disk.
Step 5	UCS-A /org/storage-profile/lun-set* # **create virtual-drive-def**	Enters the virtual drive configuration settings mode.
Step 6	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set access-policy** {**blocked** | **platform-default** | **read-only** | **read-write**}	Specifies the type of access allowed.
Step 7	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set drive-cache** {**disable** | **enable** | **no-change** | **platform-default**}	Specifies the type of drive cache.
Step 8	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set io-policy** {**cached** | **direct** | **platform-default**}	Specifies the type of Input/Output Policy.
Step 9	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set read-policy** {**normal** | **platform-default** | **read-ahead**}	Specifies the read-ahead cache mode.
Step 10	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set security** {**no** | **yes**}	Configure this option to secure a virtual drive.
Step 11	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set strip-size** {**1024kb** | **128kb** | **16kb** | **256kb** | **32kb** | **512kb** | **64kb** | **8kb** | **platform-default**}	Specifies the portion of the striped data segment that resides on each physical disk.
Step 12	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **set write-cache-policy** {**always-write-back** | **platform-default** | **write-back-good-bbu** | **write-through**}	Specifies the type of write policy.
Step 13	UCS-A /org/storage-profile/lun-set/virtual-drive-def* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a LUN set and configure the virtual drive settings:

```
UCS-A# scope org
UCS-A/org # enter storage-profile stroageprofile1
UCS-A/org/storage-profile # create lun-set lunset1
UCS-A/org/storage-profile/lun-set* # set disk-slot-range 2
```

```
UCS-A/org/storage-profile/lun-set* # create virtual-drive-def
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set access-policy read-write
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set drive-cache enable
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set io-policy direct
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set read-policy read-ahead
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set security yes
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set strip-size 512kb
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # set write-cache-policy platform-default
UCS-A/org/storage-profile/lun-set/virtual-drive-def* # commit-buffer
```

What to do next

Associate the Storage Profile with a Service Profile

Deleting a LUN Set

You can delete a LUN Set within a storage profile at the org level and within a dedicated storage profile at the service-profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **enter storage-profile** *storage-profile-name*	Enters storage-profile mode for the specified storage profile.
Step 3	UCS-A /org/storage-profile* # **delete lun-set** *lun-set-name*	Deletes the LUN Set with the specified name.
Step 4	UCS-A /org/storage-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a LUN set:

```
UCS-A# scope org
UCS-A/org # enter storage-profile stroageprofile1
UCS-A/org/storage-profile # delete lun-set lunset1
UCS-A/org/storage-profile* # commit-buffer
```

Configuring Aero Controllers

Autoconfiguration Mode for Storage Controllers

Cisco UCS C220M6/C240M6 C-series M6 servers support PCIe SAS316-port storage controllers for Direct Attached Storage. Controllers support an Autoconfiguration mode in which the state of a newly inserted disk is automatically moved to the Unconfigured-Good state.

Because of this, you can choose whether or not to use Autoconfiguration by creating a Storage Profile and associating it with the server. The default is that the automatic configuration feature is disabled, which retains the drive state when the server is rebooted.

If Autoconfiguration is used, you must select a drive state from one of the following:

- Unconfigured-Good
- JBOD
- RAID0 (RAID0 WriteBack)

This is because the controller firmware changes the behavior of systemPD to EPD-PT. EPD-PT is internally a RAID0 volume without any drive DDF metadata. The controller stores the metadata for identifying it as a RAID0 volume. The EPD-PT drives are considered as JBOD drives so the drive status is reported as JBOD and online.

Controller supports the following models:

- UCSC-RAID-M6T
- UCSC-RAID-M6HD
- UCSC-RAID-M6SD
- UCSX-X10C-RAIDF

The table below shows the behavior of Autoconfiguration in different scenarios.

Autoconfig Mode	Reboot/OCR	Hotplug	User Action
Unconfigured-Good (OFF)	• All Unconfigured-Good drives remain Unconfigured-Good. • All previously configured JBOD remain JBOD.	• Inserted drive remains Unconfigured-Good. • JBOD from a different server remains Unconfigured-Good on this controller.	Disabling Autoconfig has no impact on the existing configuration Any JBOD device remains as JBOD across controller boot. Any Unconfigured-Good remains unconfiguredgood across controller boot.
JBOD	• All Unconfigured-Good are converted to JBOD.	Newly inserted unconfigured device is converted to JBOD.	All Unconfigured-Good drives (non-user created) on the controller while running Autoconfig is converted to JBOD. User created Unconfigured-Good drive remains Unconfigured-Good until next reboot. During reboot Unconfigured-Good gets converted to JBOD.

Autoconfig Mode	Reboot/OCR	Hotplug	User Action
RAID0 (RAID0 WriteBack)	• All Unconfigured-Good converted to RAID0 WriteBack.	Newly inserted unconfigured device is converted to RAID0 WriteBack.	All Unconfigured-Good drives (non-user created) on the controller while running Autoconfig is converted to RAID0 WriteBack. User created Unconfigured-Good remains Unconfigured-Good across controller reboot. Any RAID0 WriteBack device remains as RAID0 WriteBack across controller reboot.

Selecting EPD-PT (JBOD) as the default configuration does not retain the Unconfigured-Good state across host reboot. The drive state can be retained by disabling the automatic configuration feature. If the Autoconfig option is used, the default automatic configuration will always mark a drive as Unconfigured-Good.

When Autoconfig is selected, then the drive is configured to the desired drive state, the JBOD and unconfigured drives will set the drive state accordingly on the next controller boot or OCR,

The following table shows sample use cases for different Autoconfig scenarios.

Use Case Scenario	Autoconfig Option
Using the server for JBOD Only (for example: Hyper converged, Hadoop data node etc)	JBOD
Using the server for RAID volume (for example: SAP HANA database)	Unconfigured-Good
Using the server for Mixed JBOD and RAID volume	Unconfigured-Good
Using the server for per drive RAID0 WriteBack (for example: Hadoop data node)	RAID0 WriteBack

Creating an Autoconfiguation Profile

You can include the storage Autoconfiguration (Auto Config) mode option in your storage profile and unconfigure it when no longer needed. Changes will take effect on the next system boot. Auto Config for storage is only available on Cisco UCS M6 servers with Aero controllers.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type / as the *org-name* .
Step 2	UCS-A/org# **scope storage-profile** *profile-name*	Enters the storage profile for the specified profile.
Step 3	UCS-A/org/storage-profile# **show detail expand**	Shows a detailed view of the Storage Profile. If Auto Config Mode has not been enabled for this storage profile, or no Aero controller is present, you should not see an entry for Auto Config Mode. If Auto Config is not configured, inserted devices will retain their state on system reboot.
Step 4	UCS-A/org/storage-profile# **set auto-config-mode** jbod | raid-0 | unconfigured-good | unspecified	Enables Auto Config Mode and sets the disk configuration mode to the desired state. If no further parameters are specified, all inserted devices will be tagged as Unconfigured Good on reboot. Enter unconfigured if you wish to disable Auto Config mode.
Step 5	UCS-A/org/storage-profile# **commit-buffer**	Commits the transaction to the system configuration.

PCH Controller Definitions

PCH SSD Controller Definition

Cisco UCS Manager Platform Controller Hub (PCH) Solid State Drive (SSD) Controller Definition provides a local storage configuration in storage profiles where you can configure all the disks in a single RAID or in a JBOD disk array.

The PCH Controller Definition configuration provides the following features:

- Ability to configure a single LUN RAID across two internal SSDs connected to the onboard PCH controller
- A way to configure the controller in two modes: AHCI (JBOD) and SWRAID (RAID).
- Ability to configure the PCH storage device in an Embedded Local LUN and Embedded Local Disk boot policy so precision control for boot order is achieved even with the presence of other bootable local storage devices in the server. Do not use the Local LUN or the Local JBOD options to boot from PCH disks
- Scrub policy support for the internal SSD drives. This is applicable only for the SWRAID mode. This does not apply for the AHCI and NORAID of PCH Controller modes.*See the UCS Manager Server Management Guide.*

- Firmware upgrade support for the internal SSD drives. Disk firmware upgrade is supported only when the PCH Controller is in SWRAID mode. It is not supported for AHCI mode.

You can configure PCH controller SSDs in a storage profile policy. You can enable or disable protect configuration which saves the LUN configuration even after a service profile disassociation. You choose a controller mode. The PCH controller configuration supports only these two RAID options: RAID0 and RAID1. Use No RAID configuration option for AHCI mode where all the disks connected to the controller configured as JBOD disks. The configuration deployment happens as part of the storage profile association to a service profile process.

Cisco UCS Manager supports PCH managed internal SSDs in the following M4 servers:

- UCSC-C240-M4L
- UCSC-C240-M4SX

Cisco UCS Manager supports the following PCH managed SSDs on the M.2 card for all M5 servers:

- 240GB M.2 6G SATA SSD
- 960GB M.2 6G SATA SSD

Note You cannot have software RAID configuration in the controller definition and legacy boot mode configuration in boot policy together in M5 servers. Only UEFI boot mode is supported with software RAID configuration in the controller definition. This condition is applicable even if the drives are not used as boot drive.

For the PCH Controller Definition configuration in a Cisco UCS Manager boot policy two new devices exist to select: PCH LUN and PCH Disk. EmbeddedLocalLun represents the boot device in SWRAID mode and EmbeddedLocalDisk represent the boot devices in AHCI mode.

The system uses the same scrub policy is used to scrub supported SSDs. If the scrub is Yes, configured LUNs are destroyed as part of disassociation or re-discovery. If the scrub is No, configured LUNs are saved during disassociation and re-discovery.

Cisco UCS Manager supports firmware upgrade for the internal SSDs only when the PCH Controller is in SWRAID mode. It is not supported in AHCI mode.

FCH Controller Configuration

Fusion Controller Hub (FCH) SSD Controller Definition provides a local storage configuration in storage profiles for AMD based Cisco UCS C125 M5 Server. For AMD processor based servers, the PCH controller is referred to as FCH controller. The controller type remains as PCH in the Cisco UCS Manager GUI.

The FCH Controller works the same as PCH Controller except for the following differences:

- FCH is only in AHCI (JBOD) mode.

Note Cisco UCS Manager GUI shows the RAID support as **RAID0, RAID1**, however, Cisco UCS C125 M5 Server supports only AHCI mode.

Note You must re-acknowledge the server if you remove or insert disks managed by the PCH Controller.

- There are two FCH controllers:
  - First PCH controller manages SATA disks in the front panel (in the absence of a separate PCIe storage controller)
  - Second PCH controller manages the M.2 SSDs

Note For Cisco UCS C125 M5 Server, the PCH IDs are 3 and 4.

Note Further information and procedures related to PCH controller in this document are applicable to both Intel and AMD based servers.

Creating a Storage Profile PCH Controller Definition

You can create a PCH controller definition under a storage profile at the org level or at the service profile level.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*. **Note** This task assumes the storage profile is at the org level. If the storage profile is at the service profile level, see the example below for the steps to scope to the storage profile definition under the service profile.
Step 2	UCS-A /org # **scope storage-profile** *storage-profile-name*	Enters storage-profile configuration mode for the selected storage profile.
Step 3	UCS-A /org/storage-profile # **create controller-def** *controller-definition-name*	Creates a PCH controller definition with the specified name and enters controller-definition configuration mode.
Step 4	UCS-A /org/storage-profile/controller-def* # **create controller-mode-config**	Creates a PCH controller configuration and enters controller-mode configuration mode.

	Command or Action	Purpose
Step 5	UCS-A /org/storage-profile/controller-def/controller-mode-config* # **set protect-config** {**yes**|**no**}	Specifies whether the server retains the configuration in the PCH controller even if the server is disassociated from the service profile.
Step 6	UCS-A /org/storage-profile/controller-def/controller-mode-config* # **set raid-mode** {**any-configuration** | **disable-local-storage** |**no-local-storage** | **no-raid** | **raid-0-striped** | **raid-1-mirrored** | **raid-5-striped-parity** | **raid-50--striped-parity-and-striped** | **raid-6-striped-dual-parity** | **raid-60-striped-dual-parity-and-striped** | **raid-10-mirrored-and-striped**}	Specifies the raid mode for the PCH controller.
Step 7	UCS-A /org/storage-profile/controller-def/controller-mode-config* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to add a PCH controller definition called "raid1-controller" with raid mode set to RAID 1 Mirrored to the org-level storage profile named "storage-profile-A".

```
UCS-A# scope org /
UCS-A /org # scope storage-profile storage-profile-A
UCS-A /org/storage-profile # create controller-def raid1-controller
UCS-A /org/storage-profile/controller-def* # create controller-mode-config
UCS-A /org/storage-profile/controller-def/controller-mode-config* # set protect-config yes
UCS-A /org/storage-profile/controller-def/controller-mode-config* # set raid-mode
raid-1-mirrored
UCS-A /org/storage-profile/controller-def/controller-mode-config* # commit buffer
```

This example shows how to scope to the service profile called "Service-Profile1", create a storage profile, then create a PCH controller definition called "Raid60Ctrlr" within that storage profile. The controller definition has protection mode off and uses RAID 60 Striped Dual Parity and Striped.

```
UCS-A /org/service-profile # scope org /
UCS-A /org # scope service-profile Service-Profile1
UCS-A /org/service-profile # create storage-profile-def
UCS-A /org/service-profile/storage-profile-def* # create controller-def Raid60Ctrlr
UCS-A /org/service-profile/storage-profile-def/controller-def* # create controller-mode-config
UCS-A /org/service-profile/storage-profile-def/controller-def/controller-mode-config* # set
 protect-config no
UCS-A /org/service-profile/storage-profile-def/controller-def/controller-mode-config* # set
 raid-mode raid-60-striped-dual-parity-and-striped
UCS-A /org/service-profile/storage-profile-def/controller-def/controller-mode-config* #
commit-buffer
```

Deleting a Storage Profile PCH Controller Definition

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*. **Note** This task assumes the storage profile is at the org level. If the storage profile is at the service profile level, see the example below for the steps to scope to the storage profile definition under the service profile.
Step 2	UCS-A /org # **scope storage-profile** *storage-profile-name*	Enters storage-profile configuration mode for the selected storage profile.
Step 3	UCS-A /org/storage-profile # **delete controller-def** *controller-definition-name*	Deletes a PCH controller definition with the specified name.
Step 4	UCS-A /org/storage-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to delete a PCH controller definition called "raid1-controller" from the org-level storage profile named "storage-profile-A".

```
UCS-A# scope org
UCS-A /org # scope storage-profile storage-profile-A
UCS-A /org/storage-profile # delete controller-def raid1-controller
UCS-A /org/storage-profile* # commit-buffer
```

Migrating an M.2 Module

Migrating an M.2 module in SWRAID

Perform this procedure to migrate an M.2 module in SWRAID mode to a destination server:

Before you begin

Only UEFI boot mode is supported with software RAID configuration in the controller definition. This condition is applicable even if the drives are not used as boot drive. Ensure that the source and destination server boot mode is set to UEFI and controller definition is configured as same SWRAID (R0/R1).

Procedure

Step 1 Gracefully shut down the server.

Step 2 Physically remove the M.2 module.

The boot mode in the source server for SWRAID M.2 controller configuration in the source server has to be UEFI. Configure the boot policy of destination server with UEFI boot parameters under embedded disk.

Step 3 Insert the disk in the M.2 module in the destination server.

Step 4 Power on the server.

Step 5 Re-acknowledge the server.

Migrating an M.2 Module in AHCI Mode

Perform this procedure to migrate an M.2 module in NORAID mode to a destination server:

Before you begin

- If the source server is in legacy boot mode, ensure that the destination server is also in legacy boot mode and controller definition is configured as **NORAID**.
- If the source server is in UEFI boot mode, ensure that the destination server is also in UEFI boot mode and controller definition is configured as **NORAID**.

Procedure

Step 1 Gracefully shut down the server.

Step 2 Physically remove the M.2 module.

Step 3 Do one of the following:

- If the disk under M.2 controller had boot mode as UEFI on the source server, configure the boot policy of the destination server with UEFI boot parameters.
- If the disk under M.2 controller had boot mode as legacy on the source server, configure the boot policy of the destination server as legacy mode

Step 4 Insert the M.2 module in the destination server.

Step 5 Power on the server.

Step 6 Re-acknowledge the server.

Note If the disk is faulty, the server shows the disk status as **Not Detected**. Perform Replacing a Faulty M.2 Disk, on page 40 to replace the faulty disk.

Migrating a SWRAID Disk

Perform this procedure to migrate a M.2 disk in SWRAID mode to a destination server:

Before you begin

Only UEFI boot mode is supported with software RAID configuration in the controller definition. This condition is applicable even if the drives are not used as boot drive. Ensure that the source and destination server boot mode is set to UEFI and controller definition is configured as same SWRAID (R0/R1).

Procedure

Step 1 Gracefully shut down the server.

Step 2 Physically remove the M.2 module and extract the disk.

If the disk is used as SWRAID in the source server the boot mode has to be UEFI and configure boot policy of destination server with UEFI boot parameters under embedded disk.

Step 3 Insert the disk in the M.2 module in the destination server.

Step 4 Power on the server.

Step 5 Re-acknowledge the server.

Note The **Drive State** of the disk should show as **Online**. If the disk is faulty, the sever fails to detect the disk or the **Drive State** shows as **BAD** (or **FAILED**) instead of **Online**. Perform Replacing a Faulty M.2 Disk, on page 40 to replace the faulty disk.

Migrating a JBOD Disk in AHCI Mode

Perform this procedure to migrate a JBOD disk in NORAID mode to a destination server:

Before you begin

- If the source server is in legacy boot mode, ensure that the destination server is also in legacy boot mode and controller definition is configured as **NORAID**.
- If the source server is in UEFI boot mode, ensure that the destination server is also in UEFI boot mode and controller definition is configured as **NORAID**.

Procedure

Step 1 Gracefully shut down the server.

Step 2 Physically remove the module and extract the M.2 disk.

Step 3 Do one of the following:

- If the disk under M.2 controller had boot mode as UEFI on the source server, configure the boot policy of the destination server with UEFI boot parameters.
- If the disk under M.2 controller had boot mode as legacy on the source server, configure the boot policy of the destination server as legacy mode

Step 4 Insert the M.2 disk in the M.2 module on the destination server.

Step 5 Power on the server.

Step 6 Re-acknowledge the server.

Replacing a Faulty M.2 Disk

Perform this procedure to replace a faulty M.2 disk.

Before you begin

Ensure that the SWRAID controller definition is configured and the replacement disk formatted empty drive.

Procedure

Step 1 Gracefully power down the server.

Step 2 Physically remove the faulty M.2 drive. Use the **Serial Number** and **Disk Slot** to identity the faulty disk.

Step 3 Insert the replacement M.2 drive.

Step 4 Power on the server.

Step 5 Wait for the disk to rebuild and then re-acknowledge the server.

Note SWRAID rebuild may take anywhere between 35 to 75 minutes depending on the disk size, disk speed, OS content, and other parameters.

AHCI is a NORAID configuration and hence rebuild is not applicable.

Note After replacing the faulty M.2 drive, the operability state and drive-state of the drive in other slot change to Degraded and Rebuilding. To bring back the drive to normal state, decommission and recommission the blade.

Associating a Storage Profile with a Service Profile

A storage profile created under org can be referred by multiple service profiles, and a name reference in service profile is needed to associate the storage profile with a service profile.

Important Storage profiles can be defined under org and under service profile (dedicated). Hence, a service profile inherits local LUNs from both possible storage profiles. A service profile can have a maximum of two such local LUNs.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.

	Command or Action	Purpose
Step 2	UCS-A /org # **scope service-profile** *service-profile-name*	Enters the specified service profile mode.
Step 3	UCS-A /org/service-profile # **set storage-profile-name** *storage-profile-name*	Associates the specified storage profile with the service profile. **Note** To dissociate the service profile from a storage profile, use the **set storage-profile-name** command and specify **""** as the storage profile name.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to associate a storage profile with a service profile.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # set storage-profile-name stp2
```

This example shows how to dissociate a service profile from a storage profile.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # set storage-profile-name ""
```

Displaying Details of All Local LUNs Inherited By a Service Profile

Storage profiles can be defined under org and as a dedicated storage profile under service profile. Thus, a service profile inherits local LUNs from both possible storage profiles. It can have a maximum of 2 such local LUNs. You can display the details of all local LUNs inherited by a service profile by using the following command:

Procedure

	Command or Action	Purpose
Step 1	UCS-A /org/service-profile # **show local-lun-ref**	Displays the following detailed information about all the local LUNs inherited by the specified service profile: • **Name**—LUN name in the storage profile. • **Admin State**—Specifies whether a local LUN should be deployed or not. Admin state can be **Online** or **Undeployed**.

	Command or Action	Purpose

		When the local LUN is being referenced by a service profile, if the auto-deploy status is **no-auto-deploy** then the admin state will be **Undeployed**, else it will be **Online**. After the local LUN is referenced by a service profile, any change made to this local LUN's auto-deploy status is not reflected in the admin state of the LUN inherited by the service profile.
		• **RAID Level**—Summary of the RAID level of the disk group used.
		• **Provisioned Size (GB)**—Size, in GB, of the LUN specified in the storage profile.
		• **Assigned Size (MB)**—Size, in MB, assigned by UCSM.
		• **Config State**—State of LUN configuration. The states can be one of the following:
		  • **Applying**—Admin state is online, the LUN is associated with a server, and the virtual drive is being created.
		  • **Applied**—Admin state is online, the LUN is associated with a server, and the virtual drive is created.
		  • **Apply Failed**—Admin stage is online, the LUN is associated with a server, but the virtual drive creation failed.
		  • **Not Applied**—The LUN is not associated with a server, or the LUN is associated with a service profile, but admin state is undeployed.
		  • **Not In Use**—Service profile is using the virtual drive, but the virtual drive is not associated with a server.
		• **Reference LUN**—The preprovisioned virtual drive name, or UCSM-generated virtual drive name.
		• **Deploy Name**—The virtual drive name after deployment.
		• **ID**—Virtual drive ID.

	Command or Action	Purpose
		• **Drive State**—State of the virtual drive. The states are: • **Unknown** • **Optimal** • **Degraded** • **Inoperable** • **Partially Degraded** • **Self Test Failed** **Note** The *Self Test Failed* drive state enables you to monitor the health and performance of the virtual drive. In this drive state: • The existing virtual drive operation or a new virtual drive creation works normally, unless the storage controller fails the virtual drive for any of the legitimate faults. • The degree of the virtual drive failure is not displayed. However, most of the operations such as participation in Boot Order Policy, Secure Erase, and LED are still supported, except for the drive state modification. • The drive can soon become unusable and can result in loss of information.

Example

```
UCS-A /org/service-profile # show local-lun-ref


Local LUN Ref:

    Profile LUN Name Admin State RAID Level             Provisioned Size (GB)  Assigned
Size (MB)   Config State Referenced Lun Deploy Name ID          Drive State

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

    luna             Online      RAID 0 Striped         1
     1024 Applied      luna-1         luna-1      1003        Optimal

    lunb             Online      RAID 0 Striped         1
     1024 Applied      lunb-1         lunb-1      1004        Optimal

UCS-A /org/service-profile #



Local LUN Ref:
    Name             Admin State RAID Level             Provisioned Size (GB)  Assigned
Size (MB)   Config State Referenced Lun Deploy Name ID          Drive State
    ---------------- ----------- ---------------------- ----------------------
-------------------- ------------ -------------- ----------- ----------- -----------
    lun111           Online      RAID 0 Striped         30                     30720
          Applied      lun111-1       lun111-1    1001        Optimal
    lun201           Online      Unspecified            1                      0
          Not Applied
```

Importing Foreign Configurations for a RAID Controller

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope raid-controller** *raid-contr-id* {**sas** | **sata**}	Enters RAID controller chassis mode.
Step 3	UCS-A /chassis/raid-controller # **set admin-state import-foreign-configuration**	Allows import of configurations from local disks that are in the **Foreign Configuration** state.

Example

This example shows how to import foreign configurations from local disks that are in the **Foreign Configuration** state:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope raid-controller 1 sas
UCS-A /chassis/raid-controller # set admin-state import-foreign-configuration
UCS-A /chassis/raid-controller* #
```

Configuring Local Disk Operations

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope raid-controller** *raid-contr-id* {**sas** | **sata**}	Enters RAID controller chassis mode.
Step 3	UCS-A /chassis/raid-controller # **scope local-disk** *local-disk-id*	Enters local disk configuration mode.
Step 4	UCS-A /chassis/raid-controller/local-disk # **set admin-state** {**clear-foreign-configuration** | **dedicated-hot-spare** [*admin-vd-id*] | **prepare-for-removal** | **remove-hot-spare** | **unconfigured-good** | **undo-prepare-for-removal**}	Configures the local disk to one of the following states: • **clear-foreign-configuration**—Clears any foreign configuration that exists in a local disk when it is introduced into a new configuration. • **dedicated-hot-spare**—Specifies the local disk as a dedicated hot spare. The admin virtual drive ID that you can assign ranges from 0 to 4294967295. • **prepare-for-removal**—Specifies that the local disk is marked for removal from the chassis. • **remove-hot-spare**—Specifies that the local disk is no longer a hot spare. Use this only to clear any mismatch faults. • **unconfigured-good**—Specifies that the local disk can be configured. • **undo-prepare-for-removal**—Specifies that the local disk is no longer marked for removal from the chassis.

Example

This example shows how to clear any foreign configuration from a local disk:

```
UCS-A /chassis/raid-controller/local-disk # set admin-state clear-foreign-configuration
```

This example shows how to specify a local disk as a dedicated hot spare:

```
UCS-A /chassis/raid-controller/local-disk* # set admin-state dedicated-hot-spare 1001
```

This example shows how to specify that a local disk is marked for removal from the chassis:

```
UCS-A /chassis/raid-controller/local-disk* # set admin-state prepare-for-removal
```

This example shows how to specify that a local disk is marked for removal as a hot spare:

```
UCS-A /chassis/raid-controller/local-disk* # set admin-state remove-hot-spare
```

This example shows how to specify that a local disk is working, but is unconfigured for use:

```
UCS-A /chassis/raid-controller/local-disk* # set admin-state unconfigured-good
```

This example shows how to specify that a local disk is no longer marked for removal from the chassis:

```
UCS-A /chassis/raid-controller/local-disk* # set admin-state undo-prepare-for-removal
```

Configuring Virtual Drive Properties

All virtual drives in a disk group must be managed by using a single disk group policy.

If you try to associate to a server that does not support these properties, a configuration error will be generated.

Only the following storage controllers support these properties:

- LSI 6G MegaRAID SAS 9266-8i
- LSI 6G MegaRAID SAS 9271-8i
- LSI 6G MegaRAID 9265-8i
- LSI MegaRAID SAS 2208 ROMB
- LSI MegaRAID SAS 9361-8i

For the LSI MegaRAID SAS 2208 ROMB controller, these properties are supported only in the B420-M3 blade server. For the other controllers, these properties are supported in multiple rack servers.

Note If you have a setup with the Cisco Boot Optimized M.2 Raid Controller (UCS-M2-HWRAID), then:

- You can create only one virtual drive
- For **strip-size**, select **64KB** or **32KB**. Selecting any other value results in configuration error.
- For **access-policy**, **read-policy**, **write-cache-policy**, **io-policy**, and **drive-cache**, select **platform-default**. Selecting any other value results in configuration error.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org# **scope disk-group-config-policy** *disk-group-name*	Enters disk group configuration policy mode for the specified disk group name.
Step 3	UCS-A /org/disk-group-config-policy* # **create virtual-drive-def**	Creates a virtual drive definition and enters the virtual drive definition mode.
Step 4	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set access-policy** *policy-type*	Specifies the access policy. This can be one of the following: • blocked • platform-default • read-only: • read-write
Step 5	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set drive-cache** *state*	Specifies the state of the drive cache. This can be one of the following: • enable • disable • no-change • platform-default **Important** In Cisco UCS Release 2.5, the drive cache state cannot be changed. It will remain as **platform-default**, irrespective of the drive cache state that you select.
Step 6	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set io-policy** *policy-type*	Specifies the I/O policy. This can be one of the following: • cached • direct • platform-default
Step 7	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set read-policy** *policy-type*	Specifies the read policy. This can be one of the following: • normal

	Command or Action	Purpose
		• platform-default • read-ahead
Step 8	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set strip-size** *strip-size*	Specifies the strip size. This can be one of the following: • 64 KB • 128 KB • 256 KB • 512 KB • 1024 KB • platform-default
Step 9	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **set write-cache-policy** *policy-type*	Specifies the write-cache-policy. This can be one of the following: • always-write-back • platform-default • write-back-good-bbu • write-through
Step 10	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **commit-buffer**	Commits the transaction to the system configuration.
Step 11	UCS-A /org/disk-group-config-policy/virtual-drive-def* # **show**	Displays the configured virtual drive properties.

Example

This example shows how to configure virtual disk properties:

```
UCS-A# scope org
UCS-A /org # scope disk-group-config-policy raid0policy
UCS-A /org/disk-group-config-policy # create virtual-drive-def
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set access-policy read-write
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set drive-cache enable
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set io-policy cached
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set read-policy normal
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set strip-size 1024
UCS-A /org/disk-group-config-policy/virtual-drive-def* # set write-cache-policy write-through
UCS-A /org/disk-group-config-policy/virtual-drive-def* # commit-buffer
UCS-A /org/disk-group-config-policy/virtual-drive-def # show

Virtual Drive Def:
    Strip Size (KB): 1024KB
```

```
    Access Policy: Read Write
    Read Policy: Normal
    Configured Write Cache Policy: Write Through
    IO Policy: Cached
    Drive Cache: Enable
UCS-A /org/disk-group-config-policy/virtual-drive-def #
```

What to do next

Create a Storage Profile

Deleting an Orphaned Virtual Drive

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope raid-controller** *raid-contr-id* {**sas** | **sata**}	Enters RAID controller chassis mode.
Step 3	(Optional) UCS-A /chassis/raid-controller # **delete virtual-drive id** *virtual-drive-id*	Deletes the orphaned virtual drive with the specified virtual drive ID.
Step 4	(Optional) UCS-A /chassis/raid-controller # **delete virtual-drive name** *virtual-drive-name*	Deletes the orphaned virtual drive with the specified virtual drive name.
Step 5	(Optional) UCS-A /chassis/raid-controller # **scope virtual-drive** *virtual-drive-id*	Enters virtual drive mode for the specified orphaned virtual drive.
Step 6	UCS-A /chassis/raid-controller/virtual-drive # **set admin-state delete**	Deletes the orphaned virtual drive.
Step 7	UCS-A /chassis/raid-controller/virtual-drive # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to delete an orphan virtual drive by specifying the virtual drive ID.

```
UCS-A# scope chassis 1
UCS-A /chassis # scope raid-controller 1 sas
UCS-A /chassis/raid-controller # show virtual-drive

Virtual Drive:
    ID: 1001
    Name: lun111-1
    Block Size: 512
    Blocks: 62914560
    Size (MB): 30720
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 0
    Change Qualifier: No Change
    Config State: Applied
```

```
    Deploy Action: No Action

    ID: 1002
    Name: luna-1
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 1
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

    ID: 1003
    Name: lunb-1
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 2
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

    ID: 1004
    Name: lunb-2
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 3
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

    ID: 1005
    Name: luna-2
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 4
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

...

UCS-A /chassis/raid-controller # delete virtual-drive id 1002
Warning: When committed, the virtual drive will be deleted, which may result in data loss.

UCS-A /chassis/raid-controller # commit-buffer
```

This example shows how to delete an orphan virtual drive by specifying the virtual drive name.

```
UCS-A# scope chassis 1
UCS-A /chassis # scope raid-controller 1 sas
UCS-A /chassis/raid-controller # show virtual-drive

Virtual Drive:
    ID: 1001
```

```
    Name: lun111-1
    Block Size: 512
    Blocks: 62914560
    Size (MB): 30720
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 0
    Change Qualifier: No Change
    Config State: Applied
    Deploy Action: No Action

    ID: 1003
    Name: lunb-1
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 2
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

    ID: 1004
    Name: lunb-2
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 3
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

    ID: 1005
    Name: luna-2
    Block Size: 512
    Blocks: 2097152
    Size (MB): 1024
    Operability: Operable
    Presence: Equipped
    Oper Device ID: 4
    Change Qualifier: No Change
    Config State: Orphaned
    Deploy Action: No Action

...

UCS-A /chassis/raid-controller # delete virtual-drive name lunb-1
Warning: When committed, the virtual drive will be deleted, which may result in data loss.

UCS-A /chassis/raid-controller # commit-buffer
```

This example shows how to delete an orphan virtual drive by setting the admin-state.

```
UCS-A# scope chassis 1
UCS-A /chassis # scope raid-controller 1 sas
UCS-A /chassis/raid-controller # scope virtual-drive 1004
UCS-A /chassis/raid-controller/virtual-drive # set admin-state delete

Warning: When committed, the virtual drive will be deleted, which may result in data loss.

UCS-A /chassis/raid-controller/virtual-drive # commit-buffer
```

Renaming an Orphaned Virtual Drive

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope raid-controller** *raid-contr-id* {**sas** | **sata**}	Enters RAID controller chassis mode.
Step 3	UCS-A /chassis/raid-controller # **scope virtual-drive** *virtual-drive-id*	Enters virtual drive mode for the specified virtual drive.
Step 4	UCS-A /chassis/raid-controller/virtual-drive # **set name** *virtual-drive-name*	Specifies a name for the orphan virtual drive.
Step 5	UCS-A /chassis/raid-controller/virtual-drive # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to specify a name for an orphan virtual drive.

```
UCS-A /chassis # scope raid-controller 1 sas
UCS-A /chassis/raid-controller # scope virtual-drive 1060
UCS-A /chassis/raid-controller/virtual-drive* # set name vd1
UCS-A /chassis/raid-controller/virtual-drive* # commit-buffer
```

Boot Policy for Local Storage

You can specify the primary boot device for a storage controller as a local LUN or a JBOD disk. Each storage controller can have one primary boot device. However, in a storage profile, you can set only one device as the primary boot LUN.

Beginning with 4.0(4a), Cisco UCS Manager supports Cisco boot optimized M.2 Raid controller based off Marvell 88SE92xx PCIe to SATA 6Gb/s controller (UCS-M2-HWRAID). The controller supports only UEFI boot mode.

Local storage option in the boot policy supports the boot from the SATA drives in the Cisco boot optimized M.2 Raid controller.

Also, embedded local storage option in the boot policy supports the boot from the SATA drives in the Cisco boot optimized M.2 Raid controller. The primary and the secondary type boot specifically from the M.2 SATA drives.

Note Cisco UCS S3260 M3 server nodes do not support **Any** as an option for the **Local LUN Image Path** while adding a local LUN to the boot policy through the Cisco UCS Manager GUI. In the Cisco UCS Manager CLI, Cisco UCS S3260 M3 server nodes do not support the **local-any** command option.

Configuring the Boot Policy for a Local LUN

Note In Cisco UCS Manager Release 2.5, you cannot configure JBOD as a boot device.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type **/** as the *org-name* .
Step 2	UCS-A /org # **scope boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create local-lun**	Specifies a local hard disk drive as the local storage.
Step 6	UCS-A /org/boot-policy/storage/local/local-lun # **create local-lun-image-path** {**primary** | **secondary**}	Specifies the boot order for the LUN that you specify. **Important** Cisco UCS Manager Release 2.2(4) does not support **secondary** boot order.
Step 7	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # **set lunname** *lun_name*	Specifies the name of the LUN that you want to boot from.
Step 8	UCS-A /org/boot-policy/storage/local/*local-storage-device* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create a local hard disk drive boot for the policy, specify a boot order and a LUN to boot from, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local* # create local-lun
UCS-A /org/boot-policy/storage/local/local-lun # create local-lun-image-path primary
```

```
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # set lunname luna
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # commit-buffer
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path #
```

What to do next

Include the boot policy in a service profile and template.

Configuring the Boot Policy for a Local JBOD Disk

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type **/** as the *org-name* .
Step 2	UCS-A /org # **scope boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create local-jbod**	Specifies the local JBOD as the local storage.
Step 6	UCS-A /org/boot-policy/storage/local/local-jbod # **create local-lun-image-path***{primary | secondary}*	
Step 7	UCS-A /org/boot-policy/storage/local/local-jbod/local-disk-image-path # **set slotnumber** *slotnumber*	
Step 8	UCS-A /org/boot-policy/storage/local/local-jbod/local-disk-image-path* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create an local JBOD disk drive boot for the policy, specify a boot order and a JBOD to boot from, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local/ # create local-jbod
UCS-A /org/boot-policy/storage/local/local-jbod* # create local-disk-image-path primary
```

```
UCS-A /org/boot-policy/storage/local/local-jbod/local-disk-image-path # set slotnumber 1
UCS-A /org/boot-policy/storage/local/local-jbod/local-disk-image-path* # commit-buffer
UCS-A /org/boot-policy/storage/local/local-jbod/local-disk-image-path #
```

What to do next

Include the boot policy in a service profile and template.

Configuring the Boot Policy for an Embedded Local LUN

Note Specify one bootable LUN as either primary or secondary boot device. If you specify the bootable LUN as both primary and secondary boot devices, the boot policy will result in the service profile configuration error.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type **/** as the *org-name* .
Step 2	UCS-A /org # **scope boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create embedded-local-lun**	Specifies the embedded local LUN as the local storage.
Step 6	UCS-A /org/boot-policy/storage/local/embedded-local-lun* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create an embedded LUN boot for the policy, specify a boot order and a LUN to boot from, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local/ # create embedded-local-lun
UCS-A /org/boot-policy/storage/local/embedded-local-lun* # commit-buffer
UCS-A /org/boot-policy/storage/local/embedded-local-lun #
```

What to do next

Include the boot policy in a service profile and template.

Configuring the Boot Policy for an Embedded Local Disk

Note For Cisco UCS C125 M5 Server, if there is no separate PCIe storage controller, then do not configure boot policy for embedded local disk. Instead, use **Add Local Disk** option.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type **/** as the *org-name* .
Step 2	UCS-A /org # **scope boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create embedded-local-jbod**	Specifies the embedded local JBOD as the local storage.
Step 6	UCS-A /org/boot-policy/storage/local/embedded-local-jbod* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create an embedded JBOD disk drive boot for the policy, specify a boot order and a JBOD to boot from, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local/ # create embedded-local-jbod
UCS-A /org/boot-policy/storage/local/embedded-local-jbod* # commit-buffer
UCS-A /org/boot-policy/storage/local/embedded-local-jbod #
```

What to do next

Include the boot policy in a service profile and template.

Local LUN Operations in a Service Profile

Although a service profile is derived from a service profile template, the following operations can be performed for each local LUN at the individual service profile level:

- Preprovisioning a LUN Name or Claiming an Orphan LUN, on page 57
- Deploying and Undeploying a LUN, on page 58
- Renaming a Service Profile Referenced LUN, on page 59

Note Preprovisioning a LUN name, claiming an orphan LUN, and deploying or undeploying a LUN result in server reboot.

Preprovisioning a LUN Name or Claiming an Orphan LUN

You can preprovision a LUN name or claim an orphan LUN by using the **set ref-name** command. Preprovisioning a LUN name or claiming an orphan LUN can be done only when the admin state of the LUN is **Undeployed**. You can also manually change the admin state of the LUN to **Undeployed** and claim an orphan LUN.

Important This operation will reboot the server.

If the LUN name is empty, set a LUN name before claiming it.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org# **scope service-profile** *service-profile-name*	Enters the specified service profile mode.
Step 3	UCS-A /org/service-profile# **scope local-lun-ref** *lun-name*	Enters the specified LUN.
Step 4	UCS-A /org/service-profile/local-lun-ref# **set ref-name** *ref-lun-name*	Sets the referenced LUN name. If this LUN name exists and the LUN is orphaned, its is claimed by the service profile. If this LUN does not exist, a new LUN is created with the specified name.

- If the LUN exists and is not orphaned, a configuration failure occurs.

- If a LUN is already referred to and the ref-name is changed, it will release the old LUN and will claim or create a LUN with the ref-name. The old LUN is marked as an orphan after the LUN reference is removed from the server.

Example

This examples shows how to preprovision a LUN name.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile* # scope local-lun-ref lun1
UCS-A /org/service-profile/local-lun-ref* # set ref-name lun2
```

Deploying and Undeploying a LUN

You can deploy or undeploy a LUN by using the **admin-state** command. If the admin state of a local LUN is **Undeployed**, the reference of that LUN is removed and the LUN is not deployed.

Important This operation will reboot the server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org# **scope service-profile** *service-profile-name*	Enters the specified service profile mode.
Step 3	UCS-A /org/service-profile# **scope local-lun-ref** *lun-name*	Enters the specified LUN.
Step 4	UCS-A /org/service-profile/local-lun-ref# **set admin-state** {**online** | **undeployed**}	Sets the admin state of the specified LUN to **online** or **undeployed**. If a LUN is already referred to and the admin state is set to **undeployed**, it will release the old LUN. The old LUN is marked as orphan after the LUN reference is removed from the server.

Example

This examples shows how to deploy a LUN.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile* # scope local-lun-ref lun1
UCS-A /org/service-profile/local-lun-ref* # set admin-state online
```

This examples shows how to undeploy a LUN.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile* # scope local-lun-ref lun1
UCS-A /org/service-profile/local-lun-ref* # set admin-state undeployed
```

Renaming a Service Profile Referenced LUN

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org# **scope service-profile** *service-profile-name*	Enters the specified service profile mode.
Step 3	UCS-A /org/service-profile# **scope local-lun-ref** *lun-name*	Enters the specified LUN.
Step 4	UCS-A /org/service-profile/local-lun-ref# **set name**	Renames the referenced LUN.

Example

This examples shows how to rename a LUN referenced by a service profile.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile* # scope local-lun-ref lun1
UCS-A /org/service-profile/local-lun-ref* # set name lun11
```