Notes, cautions, and warnings

NOTE: A NOTE indicates important information that helps you make better use of your product.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.
1 Overview.................................................................................................................................................. 5
   Documentation matrix.................................................................................................................................. 5

2 Accessing the command prompt............................................................................................................. 6
   Using CLI commands from Windows command prompts......................................................................... 6
   Using CLI commands in Linux.................................................................................................................. 6
   Using CLI commands in VMware............................................................................................................. 6

3 Working with the PERC Command Line Interface tool............................................................................. 7
   System commands....................................................................................................................................... 7
   System show commands............................................................................................................................ 7
   Controller commands.................................................................................................................................. 8
   Show and set controller properties commands.......................................................................................... 8
   Controller show commands....................................................................................................................... 12
   Controller background tasks operation commands.................................................................................... 13
   Patrol Read.................................................................................................................................................. 13
   Consistency check....................................................................................................................................... 15
   Controller security commands................................................................................................................... 16
   Flashing controller firmware command.................................................................................................... 17
   Controller cache command....................................................................................................................... 17
   Controller profile commands..................................................................................................................... 17
   HBA controller commands......................................................................................................................... 18
   Drive commands........................................................................................................................................ 20
   Drive show commands.............................................................................................................................. 20
   Missing drives commands......................................................................................................................... 21
   Drive initialization commands................................................................................................................... 22
   Set drive state commands.......................................................................................................................... 22
   Locate drives commands............................................................................................................................ 23
   Prepare to remove drives commands......................................................................................................... 24
   Dimmer switch commands.......................................................................................................................... 24
   Drive security commands........................................................................................................................... 25
   Drive erase commands............................................................................................................................... 25
   Rebuild drives commands.......................................................................................................................... 27
   Drive copyback commands........................................................................................................................ 27
   Hot spare drive commands.......................................................................................................................... 28
   Virtual drives commands............................................................................................................................ 29
   Add virtual drives commands.................................................................................................................... 29
   Delete virtual drives commands................................................................................................................ 32
   Delete Non-RAID disks............................................................................................................................... 33
   Virtual drive show commands.................................................................................................................... 33
   Preserved cache commands....................................................................................................................... 33
   Change virtual drive properties commands.............................................................................................. 34
   Virtual drive initialization commands........................................................................................................ 36
   Virtual drive erase commands.................................................................................................................... 36
You can set up, configure, and manage your Dell PowerEdge RAID Controller (PERC) by using the Command Line Interface (CLI).

**NOTE:** Some features may not be supported on every generation of PERC, or may require a firmware update to enable a feature. See your PERC’s User’s Guide for information on the specific features supported by that controller.

**Topics:**
- Documentation matrix

## Documentation matrix

The documentation matrix provides information on documents that you can refer to for setting up and managing your system.

**Table 1. Documentation matrix**

<table>
<thead>
<tr>
<th>To...</th>
<th>See the...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install your system into a rack</td>
<td>Rack documentation included with your rack solution.</td>
</tr>
<tr>
<td>Set up your system and know the system technical specifications</td>
<td>Getting Started With Your System that shipped with your system or see <a href="http://www.dell.com/poweredgemanuals">www.dell.com/poweredgemanuals</a>.</td>
</tr>
<tr>
<td>Install the operating system</td>
<td>Operating system documentation at <a href="http://www.dell.com/operatingsystemmanuals">www.dell.com/operatingsystemmanuals</a>.</td>
</tr>
<tr>
<td>Configure and log in to iDRAC, set up managed and management system, know the iDRAC features, and troubleshoot by using iDRAC</td>
<td>Integrated Dell Remote Access Controller User’s Guide at <a href="http://www.dell.com/idracmanuals">www.dell.com/idracmanuals</a>.</td>
</tr>
<tr>
<td>Launch, enable, and disable Dell Lifecycle Controller, know the features, use and troubleshoot Dell Lifecycle Controller</td>
<td>Dell Lifecycle Controller User’s Guide at <a href="http://www.dell.com/idracmanuals">www.dell.com/idracmanuals</a>.</td>
</tr>
<tr>
<td>Set up, use, and troubleshoot OpenManage Server Administrator</td>
<td>Dell OpenManage Server Administrator User’s Guide at <a href="http://www.dell.com/openmanagemanuals">www.dell.com/openmanagemanuals</a> &gt; OpenManage Server Administrator.</td>
</tr>
<tr>
<td>Know the features of the storage controller cards, deploy the cards, and manage the storage subsystem</td>
<td>Storage controller documentation at <a href="http://www.dell.com/storagecontrollermanuals">www.dell.com/storagecontrollermanuals</a>.</td>
</tr>
<tr>
<td>Check the event and error messages generated by the system firmware and agents that monitor system components</td>
<td>Dell Event and Error Messages Reference Guide at <a href="http://www.dell.com/openmanagesoftware">www.dell.com/openmanagesoftware</a>.</td>
</tr>
</tbody>
</table>
Accessing the command prompt

Access the CLI in Microsoft Windows, Linux, or VMware operating systems.

**Topics:**
- Using CLI commands from Windows command prompts
- Using CLI commands in Linux
- Using CLI commands in VMware

**Using CLI commands from Windows command prompts**

Ensure that you copy the `perccli.exe` and `perccli64.exe` files to `C:\Windows\System32`.

To access the command prompt in systems using the Microsoft Windows operating system, perform the following procedure:

1. Click **Start** > **Run**.
   - The **Run** window is displayed.
2. In the **Open** field, type `cmd`, and then click **OK**.
   - The **Administrator: Command Prompt** window is displayed, where you can execute the PERC CLI commands.

**Using CLI commands in Linux**

Perform the following procedures to access the command prompt in systems using the Linux operating system:

1. To install the `percli` RPM, run `rpm -ivh <percli-x.xx-x.noarch.rpm>`, or to upgrade the `percli` RPM, run `rpm -Uvh <percli-x.xx-x.noarch.rpm>`.
2. Change directory to `/opt/MegaRAID/perccli`.
3. As a root user, run `./perccli`.

**Using CLI commands in VMware**

Perform the following procedures to access the command prompt in systems using the VMware system:

1. View the list of installed VIB package using the following command: `esxcli software vib list`.
2. Install the VIB package using the command: `esxcli software vib install -v /vmfs/volume/datastore1/vmware-esx-perccli.vib` where `/vmfs/volume/datastore1` is the path detail of the VIB.
3. You can remove the installed VIB by using the command: `esxcli software vib remove -n=vmware-esx-perccli.vib --force`.
4. Run `perccli` by browsing to the following location: `cd /opt/lsi/perccli`. 

---

Accessing the command prompt
Working with the PERC Command Line Interface tool

This chapter describes the commands supported by the PERC Command Line Tool.

**NOTE:** The PERC Command Line Interface (CLI) Tool is not case sensitive.

**CAUTION:** The order in which you specify the command options should be the same as in the User Guide; otherwise, the command will fail.

**NOTE:** The PERC CLI Tool does not support the Snapshot feature.

**Topics:**

- System commands
- Controller commands
- Drive commands
- Virtual drives commands
- Foreign configurations commands
- BIOS-related commands
- Drive group commands
- BBU commands
- Enclosure commands
- PHY commands
- Logging commands
- PERC CLI command examples
- Snapdump commands

**System commands**

In the following sections, syntax is read as follows:

**Table 2. System commands reference table**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Displays information on all controllers present on the host.</td>
</tr>
<tr>
<td>cx</td>
<td>Specifies the controller where ( x ) is the controller index.</td>
</tr>
<tr>
<td>ex</td>
<td>The enclosure ID.</td>
</tr>
<tr>
<td>.&lt;file extension&gt;</td>
<td>Specifies the file required for a particular command.</td>
</tr>
<tr>
<td>sx</td>
<td>The drive slot ID of the controller.</td>
</tr>
</tbody>
</table>

**System show commands**

The PERC Command Line Tool supports the following system show commands:

```bash
perccli show
perccli show all
perccli show ctrlcount
perccli show help
perccli -v
```
The detailed description for each command follows.

**perccli show**

This command shows a summary of controller and controller-associated information for the system. The summary includes the number of controllers, the host name, the operating system information, and the overview of existing configuration.

**perccli show all**

This command shows the list of controllers and controller-associated information, information about the drives that need attention, and advanced software options.

**perccli show ctrlcount**

This command shows the number of controllers detected in the server.

**perccli show help**

This command shows help for all commands at the server level.

**perccli -v**

This command shows the version of the PERC Command Line Tool.

## Controller commands

Controller commands provide information and perform actions related to the specified controller, such as the /c0 controller. The PERC Command Line Tool supports the controller commands described in this section.

### Show and set controller properties commands

<table>
<thead>
<tr>
<th>Commands</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show &lt;properties&gt;</td>
<td>See Table 4. Properties for show and set commands,</td>
<td>Shows specific controller properties.</td>
</tr>
<tr>
<td>set &lt;properties&gt;</td>
<td>See Table 4. Properties for show and set commands,</td>
<td>Sets controller properties.</td>
</tr>
<tr>
<td>show all</td>
<td>all: Shows all properties of the virtual drive. freespacem: Shows the freespace in the controller. See Controller show commands.</td>
<td>Shows physical drive information.</td>
</tr>
</tbody>
</table>

This section provides command information to show and set controller properties.

**NOTE:** You cannot set multiple properties with a single command.

The generalized syntax for show controller properties command is as follows:

**perccli /cx show <property>**

This command shows the current value of the specified property on the specified controller.

General example output:

```
Status Code = 0
Status = Success
Description = None
```
Controller: 0
Property_name = Property_value

You can show the following properties using the `perccli /cx show <property1>|<property2>` command.

NOTE: `/cx` specifies the controller where `x` is the controller index.

```
perccli /cx show abortcconerror
perccli /cx show activityforlocate
perccli /cx show backplane
perccli /cx show badblocks
perccli /cx show batterywarning
perccli /cx show bgirate
perccli /cx show bootwithpinnedcache
perccli /cx show cachebypass
perccli /cx show cacheflushint
perccli /cx show ccrate
perccli /cx show coercion
perccli /cx show ccicompatibilitycheck
perccli /cx show copyback
perccli /cx show ds(dimmerswitch)
perccli /cx show jbd
perccli /cx show loadbalancemode
perccli /cx show maintainpdfailhistory
perccli /cx show ncq
perccli /cx show patrolread
perccli /cx show perfmode
perccli /cx show personality
perccli /cx show pi
perccli /cx show prcorrectunconfiguredareas
perccli /cx show prrate
perccli /cx show rebuildrate
perccli /cx show reconrate
perccli /cx show restorehotspare
perccli /cx show smartpollinterval
perccli /cx show time
perccli /cx show usefdeonlyencrypt
perccli /cx(x|all) show pi
```

```
perccli /cx set <property>=<value>
```

**General example output:**

```
Status Code = 0
Status = Success
Description = None
```

Controller 0, new Property_name = Property_value

The following commands are examples of the properties that can be set using the `perccli /cx set<property>=<value>` command:

```
perccli /cx set abortcconerror=<on|off>
perccli /cx set activityforlocate=<on|off>
perccli /cx set backplane mode=<value> expose=<on/off>
perccli /cx set batterywarning=<on/off>
perccli /cx set bgirate=<value>
perccli /cx set bootwithpinnedcache=<on/off>
perccli /cx set cachebypass=<on/off>
perccli /cx set cacheflushint=<value>
perccli /cx set ccrate=<value>
perccli /cx set coercion=<value>
perccli /cx set ccicompatibilitycheck=[off|seq|conc][delay=value] [starttime=yyyy/mm/dd hh] [excludevd=x-y,z]
perccli /cx set copyback=<on|off> type=ctrl|smartssd|smarthdd|all
perccli /cx set eghs [state=<on|off>] [eug=on|off] [smarter=<on|off>]
perccli /cx set ds(dimmerswitch)=<on|off> type=1|2|3|4
perccli /cx set foreignautoimport=<on/off>
```
perccli /cx set jbod=<on|off>
perccli /cx set loadbalancemode=<on|off>
perccli /cx set maintainpdfailhistory=<on|off>
perccli /cx set reconrate=<value>
perccli /cx set ncq=<on|off>
perccli /cx set patrolread|pr {=on mode=<auto|manual>}|{off}
perccli /cx set perfmode=<value>
perccli /cx set personality=RAID|HBA|eHBA
perccli /cx set pi[state=<on|off>] [import=<on|off>]
perccli /cx set prcorrectunconfiguredareas=<on|off>
perccli /cx set prrate=<value>
perccli /cx set reconstruct=<on|off>
perccli /cx set restorehotspare=<on|off>
perccli /cx set smartpollinterval=<value>
perccli /cx set stoponerror=<on|off>
perccli /cx set usefdeonlyencrytp=<on|off>
perccli /cx set time=<yyyymmdd hh:mm:ss|systemtime>

The following table lists and describes the properties for the show and set commands.

### Table 4. Properties for show and set commands

<table>
<thead>
<tr>
<th>Property name</th>
<th>Set command range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abortcconerror</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>activityforlocate</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>backplane</td>
<td>0: Use autodetect logic of backplanes, such as SGPIO and I2C SEP using GPIO pins. 1: Disable autodetect SGPIO. 2: Disable I2C SEP autodetect. 3: Disable both the autodetects.</td>
<td>Configures enclosure detection on a non-SES/expander backplane.</td>
</tr>
<tr>
<td>expose</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>batterywarning</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>bgirate</td>
<td>0 to 100</td>
<td>Sets background initialization rate in percentage.</td>
</tr>
<tr>
<td>cacheflushint</td>
<td>0 to 255, default value 4</td>
<td>Sets cache flush interval in seconds.</td>
</tr>
<tr>
<td>ccrate</td>
<td>0 to 100</td>
<td>Sets consistency check rate in percentage.</td>
</tr>
<tr>
<td>coercion</td>
<td>0: No coercion 1: 128 MB 2: 1 GB</td>
<td>Sets drive capacity in coercion mode.</td>
</tr>
<tr>
<td>consistencycheck</td>
<td>See Consistency check.</td>
<td>See Consistency check.</td>
</tr>
<tr>
<td>copyback</td>
<td>on</td>
<td>off type = smartssd</td>
</tr>
<tr>
<td>Property name</td>
<td>Set command range</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>all: Copy back enabled for both SSD drives and HDD drives. Example: percll /cx set copyback=on type=all</td>
<td>Enables/disables the commissioning of otherwise incompatible global hot spare drives as Emergency Hot Spare (EHSP) drives.</td>
</tr>
<tr>
<td>eghs</td>
<td>state=on</td>
<td>off: Enables use of hotspare drives for emergency feature. euq=on</td>
</tr>
<tr>
<td>exposeencldevice</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>ds(dimerswitch)</td>
<td>See Dimmer switch commands.</td>
<td>See Dimmer switch commands.</td>
</tr>
<tr>
<td>foreignautoimport</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>jbod</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>NOTE:</strong> Not supported by all controllers.</td>
</tr>
<tr>
<td>loadbalancemode</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>maintainpdfailhistory</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>patrolread</td>
<td>pr</td>
<td>See Patrol Read.</td>
</tr>
<tr>
<td>perfmode</td>
<td>0: Tuned to provide best IOPS, currently applicable to non-FastPath 1: Tuned to provide least latency, currently applicable to non-FastPath</td>
<td>Performance tuning setting for the controller.</td>
</tr>
<tr>
<td>personality</td>
<td>RAID</td>
<td>HBA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pi</td>
<td>[state=&lt;on</td>
<td>off&gt;]</td>
</tr>
<tr>
<td></td>
<td>[import=&lt;on</td>
<td>off&gt;]</td>
</tr>
<tr>
<td>prcorrectunconfiguredareas</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>Property name</td>
<td>Set command range</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>prrate</td>
<td>0 to 100</td>
<td>Sets patrol read rate of the virtual drives in percentage.</td>
</tr>
<tr>
<td>rebuildrate</td>
<td>0 to 100</td>
<td>Sets rebuild rate of the drive in percentage.</td>
</tr>
<tr>
<td>reconrate</td>
<td>0 to 100</td>
<td>Sets reconstruction rate for a drive in percentage.</td>
</tr>
<tr>
<td>restorehotspare</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>smartpollinterval</td>
<td>0 to 65535</td>
<td>Set time for polling of SMART errors in seconds.</td>
</tr>
<tr>
<td>spinupdrivecount</td>
<td>0 to 255</td>
<td>Sets number of drives that are spun up at a time.</td>
</tr>
<tr>
<td>spinupdelay</td>
<td>0 to 255</td>
<td>Sets spin-up delay between a group of drives or a set of drives, in seconds.</td>
</tr>
<tr>
<td>stoponerror</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>time</td>
<td>Valid time in ymmmd hh:mm:ss format or systemtime</td>
<td>Sets the controller time to your input value or the system time (local time in 24-hour format).</td>
</tr>
<tr>
<td>usefdeonlyencrypt</td>
<td>on</td>
<td>off</td>
</tr>
</tbody>
</table>

### Controller show commands

The PERC Command Line Tool supports the following show commands:

- `perccli /cx show`
- `perccli /cx show all`
- `perccli /cx show freespace`
- `perccli /cx show personality`

The detailed description for each command follows.

**perccli /cx show personality**

This command shows the personality set on the controller. eHBA mode lists the personality as eHBA.

Input example:

`perccli /c1 show personality`

**perccli /cx show [jbod]**

This command shows the summary of the controller information. The summary includes basic controller information, foreign configurations, drive groups, virtual drives, physical drives, enclosures, and BBU information. If you use the JBOD option, the command shows all Non-RAID disk(s) displayed in JBOD list. If the physical disk is Non-RAID, its type is set as JBOD and its state as Online.

Input example:

`perccli /c1 show`
perccli /cx show all
This command shows all controller information, which includes basic controller information, bus information, controller status, advanced software options, controller policies, controller defaults, controller capabilities, scheduled tasks, miscellaneous properties, foreign configurations, drive groups, virtual drives, physical drives, enclosures, and BBU information.

Input example:
perccli /c0 show all

**NOTE:** The PCI information displayed as a part of `perccli /cx show` and `perccli /cx show all` commands is not applicable for the FreeBSD operating system. Hence, the PCI information fields are displayed as N/A.

perccli /cx show freespace
This command shows the usable free space on all disk groups in the controller.

Input example:
perccli /c0 show freespace

Controller background tasks operation commands

Rebuild Rate

```
perccli /cx set rebuildrate=<value>
perccli /cx show rebuildrate
```

The detailed description for each command follows.

**perccli /cx set rebuildrate=<value>**
This command sets the rebuild task rate of the specified controller. The input value is in percentage.

Input example:
```
perccli /c0 set rebuildrate=30
```

**NOTE:** A high rebuild rate slows down I/O processing.

**perccli /cx show rebuildrate**
This command shows the current rebuild task rate of the specified controller in percentage.

Input example:
```
perccli /c0 show rebuildrate
```

Patrol Read
The PERC Command Line Tool supports the following patrol read commands:

```
perccli /cx resume patrolread
perccli /cx set patrolread =\{on mode\:<auto|manual>\}|{off}]
perccli /cx set patrolread [starttime<yyyy/mm/dd hh>] [maxconcurrentpd=value]
[inclueseds=<on|off>] [uncfgareas=<on|off>] [excludevd=x-y,z | none]
perccli /cx set patrolread delay=value
perccli /cx show patrolread
perccli /cx start patrolread
perccli /cx stop patrolread
perccli /cx suspend patrolread
```

**NOTE:** A patrol read operation is scheduled for all the physical drives of the controller.

The detailed description for each command follows.
perccli /cx resume patrolread
This command resumes a suspended patrol read operation.
Input example:
perccli /c0 resume patrolread

perccli /cx set patrolread {=on mode=<auto|manual>}|{off}
This command turns the patrol read scheduling on and sets the mode of the patrol read to automatic or manual.
Input example:
perccli /c0 set patrolread=on mode=manual

perccli /cx set patrolread [starttime=< yyyy/mm/dd hh>] | [maxconcurrentpd =<value>] | [includessds=<on|onlymixed|off>] | [uncfgareas=on|off] | [excludevd=x-y,z|none] |
This command schedules a patrol read operation. You can use the following options for patrol read command:

Table 5. Set Patrolread input options

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>starttime</td>
<td>A valid date and hour in 24 hours format.</td>
<td>Sets the start time in yyyy/mm/dd hh format.</td>
</tr>
<tr>
<td>maxconcurrentpd</td>
<td>Valid number of physical drives present.</td>
<td>Sets the number of physical drives that can be patrol read at a single time.</td>
</tr>
<tr>
<td>includessds</td>
<td>—</td>
<td>Include SSDs in the patrol read.</td>
</tr>
<tr>
<td>uncfgareas</td>
<td>—</td>
<td>Include the areas not configured in the patrol read.</td>
</tr>
<tr>
<td>excludevd</td>
<td>—</td>
<td>Excludes virtual drives from the patrol read. To exclude particular virtual drives, you can provide list of virtual drive IDs (x,y, z format) or the range of virtual drives that needs to be excluded from a patrolread (x-y format). If this option is not specified in the command, no virtual drives are excluded.</td>
</tr>
<tr>
<td>none</td>
<td>—</td>
<td>Virtual drives if any were excluded before, will be removed.</td>
</tr>
</tbody>
</table>

**NOTE:** Controller time is taken as a reference for scheduling a patrol read operation.

Input example:
perccli /c0 set patrolread=on starttime=2012/02/21 00

perccli /cx set patrolread [delay=<value>]   
This command delays the scheduled patrol read in hours.
Input example:
perccli /c0 set patrolread delay=30

perccli /cx show patrolread
This command shows the progress on the current patrol read in percentage.
Input example:
perccli /c0 show patrolread

perccli /cx start patrolread
This command starts the patrol read operation. This command starts a patrol read immediately.
Input example:
perccli /c0 start patrolread

**perccli /cx stop patrolread**

This command stops a running patrol read operation.

Input example:
perccli /c0 stop patrolread

**NOTE:** You cannot resume a stopped patrol read.

**perccli /cx suspend patrolread**

This command pauses a running patrol read operation.

Input example:
perccli /c0 suspend patrolread

**NOTE:** You can run this command only when a patrol read operation is running on the controller.

### Consistency check

The PERC Command Line Tool supports the following commands to schedule, perform, and view the status of a consistency check (CC) operation:

```
perccli /cx set consistencycheck|cc=[off|seq|conc][delay=value] starttime=yyyy/mm/dd hh
[excludevd=x-y,z|none]
perccli /cx show cc
perccli /cx show ccrate
```

The detailed description for each command follows.

**perccli /cx set consistencycheck |cc=[off|seq|conc] [delay=value] starttime=yyyy/mm/dd hh [excludevd=x-y,z |none]**

This command schedules a consistency check (CC) operation. You can use the following options with the consistency check command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc</td>
<td>seq: Sequential mode.</td>
<td>Sets CC to either sequential mode, or concurrent mode, or turns off the CC.</td>
</tr>
<tr>
<td></td>
<td>conc: Concurrent mode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>off: Turns off the consistency check.</td>
<td><strong>NOTE:</strong> The concurrent mode slows I/O processing.</td>
</tr>
<tr>
<td>delay</td>
<td>-1 and any integer value.</td>
<td>Delay a scheduled consistency check. The value is in hours. A value of 0 makes the CC runs continuously with no delay (in a loop).</td>
</tr>
<tr>
<td>starttime</td>
<td>A valid date and hour in 24-hours format.</td>
<td>Start time of a consistency check is <em>yyyy/mm/dd hh</em> format.</td>
</tr>
<tr>
<td>excludevd</td>
<td>The range should be less than the number of virtual drives.</td>
<td>Excludes virtual drives from the consistency checks. To exclude particular virtual drives, you can provide list of virtual drive names (Vx,Vy - format) or the range of virtual drives that you want to exclude from a consistency check (Vx-Vy format). If this option is not specified in the command, no virtual drives are excluded.</td>
</tr>
</tbody>
</table>

Working with the PERC Command Line Interface tool 15
**Option** | **Value range** | **Description**
--- | --- | ---
none |  | Virtual drives if any were excluded before, will be removed.

Input example:

```
perccli /c0 set CC=on starttime=2012/02/21 00 excludevd v0-v3
```

**perccli /cx show cc**

This command shows the consistency check schedule properties for a controller.

Input example:

```
perccli /c0 show cc
```

**perccli /cx show ccrate**

This command checks the status of a consistency check operation. The CC rate appears in percentage.

Input example:

```
perccli /c0 show ccrate
```

**NOTE:** A high CC rate slows I/O processing.

---

### Controller security commands

The PERC Command Line Tool supports the following controller security commands:

```
perccli /cx compare securitykey=xxxxxxxxxx
perccli /cx delete securitykey
perccli /cx set securitykey keyid=xxx
perccli /cx set perccli/cx securitykey=xxxxxxxxx keyid=xxx
perccli /cx set securitykey=xxxxxxxx oldsecuritykey=xxxxxxxx [keyid=xxx]
```

The detailed description for each command follows.

**perccli /cx compare securitykey=xxxxxxxxxx**

This command compares and verifies the security key of the controller.

**perccli /cx delete securitykey**

This command deletes the security key of the controller.

Input example:

```
perccli /c0 delete securitykey
```

**perccli /cx set securitykey keyid=xxx**

This command sets the key ID for the controller. The key ID is unique for every controller.

**perccli /cx set perccli/cx securitykey=xxxxxxxxx keyid=xxx**

This command sets the security key for the controller. You can use the following options with the set security key command:

Input example:

```
perccli /c0 set securitykey=Lsi@12345 keyid=1
```
<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Securitykey</td>
<td>Should have a combination of numbers, upper case letters, lower case letters and special characters. Minimum of 8 characters and maximum of 32 characters.</td>
<td>Security key is used to lock the drive.</td>
</tr>
<tr>
<td>keyid</td>
<td>—</td>
<td>Unique ID set for different controllers to help you specify a passphrase to a specific controller.</td>
</tr>
</tbody>
</table>

**perccli /cx set securitykey=xxxxxxxx oldsecuritykey=xxxxxxxx [keyid=xxx]**

This command changes the security key for the controller.

Input example:

```
perccli /c0 set securitykey=Lsi@12345 oldsecuritykey=pass123 keyid=1
```

**Flashing controller firmware command**

The following command flashes the controller firmware:

**perccli /cx download file=filepath [noverchk]**

This command flashes the firmware to the specified adapter from the given file location (filepath is the absolute file path). You can use the following options when you flash the firmware:

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>noverchk</td>
<td>—</td>
<td>The application flashes the controller firmware without checking the version of the firmware image.</td>
</tr>
</tbody>
</table>

**Controller cache command**

The following command flushes the controller cache:

**perccli /cx flush|flushcache**

This command flushes the controller cache.

Input example:

```
perccli /c0 flushcache
```

**Controller profile commands**

The PERC command line tool supports the following profile-related commands:

```
perccli /cx show profile
perccli /cx set profile profileid=<id>
```

The detailed description for each command follows.

**perccli /cx show profile**

This command shows current profile and profile properties.

Input example:
perccli /c1 show profile

**perccli /cx set profile profileid=<id>**

This command sets profile ID. The output contains control ID, status, and description attributes.

Input example:

```
perccli /c1 set profile profileid=<id>
```

**NOTE:** You must reboot the system for profile changes to take effect.

**NOTE:** Profile changes fail if:

- The new profile supports fewer drives than the number of drives supported in the current topology.
- Background operations (rebuild, copy back, full initialization, background initialization, patrol read, cc) are active.
- Background operations start after profile change but before you reboot the system.

### HBA controller commands

**NOTE:** The UEFI version of PERCCLI is not supported on Dell HBA330 or 12Gbps HBA controllers. Support will be added in a future PERCCLI release.

The PERC Command Line Tool supports the following HBA-related commands:

```
perccli /call show
perccli /cx download bios file=<filepath>
perccli /cx download file=<filepath>
perccli /cx/ex/sx start locate
perccli /cx/ex/sx stop locate
perccli /cx/pall show
perccli /cx show
perccli /cx show all
perccli /cx show freespace
perccli /cx show sasadd
perccli [verbose] -h | -help | ?
perccli /cx restart
perccli v
```

**perccli /call show**

This command shows information on all the controllers present on the host.

Input example:

```
perccli /call show
```

**perccli /cx download bios file=<filepath>**

Use this command to update the BIOS component on all supported controllers.

Input example:

```
perccli /c1 download bios file=<filepath>
```

**perccli /cx download file=<filepath>**

Use this command to flash the firmware with the .rom file to a specified adapter from the provided file location (file path is the absolute file path).

Input example:

```
perccli /cx download file=<filepath>
```

**perccli /cx/ex/sx start locate**

Use this command to turn on the drive LED flash to locate physical drives.
Input example:
perccli /c1/e10/s12 start locate

**perccli /cx/ex/sx stop locate**
Use this command to turn off the drive LED flash to locate physical drives.
Input example:
perccli /c1/e10/s12 stop locate

**perccli /cx/pall show**
This command shows the basic PHY layer information on a specified adapter.
Input example:
perccli /c1/pall show

**perccli /cx show**
This command shows the summary of the controller information. The summary includes basic controller information, foreign configurations, drive groups, virtual drives, physical drives, enclosures, and BBU information.
Input example:
perccli /c1 show

**perccli /cx show all <logfile>**
This command shows all of the controller information, including basic controller information, bus information, controller status, advanced software options, controller policies, controller defaults, controller capabilities, scheduled tasks, miscellaneous properties, foreign configurations, drive groups, virtual drives, physical drives, enclosures, and BBU information.
If you use the logfile option in the command syntax, the logs are written to the specified file. If you do not specify the file name, then the logs are written to the percas.log file. If you do not use the logfile option in the command syntax, the entire log output is printed to the console.
Ensure that the filename does not contain a blank space.
Input example:
perccli /c0 show all logfile=log.txt

**perccli /cx show freespace**
This command shows the usable free space in the controller.
Input example:
perccli /c0 show freespace

**perccli /cx show sasadd**
This command displays the SAS address of the specified controller.
Input example:
perccli /c1 show sasadd

**perccli [verbose] -h| -help| ?**
This command displays the perccli help.
Input example:
perccli -h
**perccli /cx restart**

Using this command, you can reset a specific controller or reset all controllers connected to the host. This command resets the chip hardware and reinitializes all the chip information.

Input example:

```
perccli /c1 restart
```

**perccli —v**

This command displays the version of the command line tool.

Input example:

```
perccli —v
```

## Drive commands

This section describes the drive commands, which provide information and perform actions related to physical drives. The following table describes frequently used virtual drive commands:

<table>
<thead>
<tr>
<th>Commands</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
</table>
| set      | missing: Sets the drive status as missing.  
good: Sets the drive status to unconfigured good.  
online: Sets the drive status to offline.  
online: Sets the drive status to online. | Sets physical drive properties. |
| show     | all: shows all properties of the physical drive. See Drive show commands. | Shows virtual drive information. |

### Drive show commands

The PERC Command Line Tool supports the following drive show commands:

```
perccli /cx[/ex]/sx show
perccli /cx[/eall]/sall show
perccli /cx[/ex]/sx|sall show all
perccli /cx[/ex]/sall show jbod
perccli /cx[/ex]/sx show jbod
```

**NOTE:** If enclosures are used to connect physical drives to the controller, specify the enclosure ID in the command. If no enclosures are used, you must specify the controller ID and slot ID.

The detailed description for each command follows.

**perccli /cx[/ex]/sx show**

This command shows the summary of the physical drive for a specified slot in the controller.

Input example:

```
perccli /c0/e0/s4,5 show
```

**perccli /cx[/eall]/sall show**

This command shows the summary information for all the enclosures and physical drives connected to the controller.

Input example:

```
perccli /c0/eall/sall show
```
perccli /cx[/ex]/sx|sall show all

This command shows all information of a physical drive for the specified slot in the controller. If you use the all option, the command shows information for all slots on the controller. x stands for a number, a list of numbers, a range of numbers, or all numbers.

Input examples:

- perccli /c0/e3/s0-3 show all
- perccli /c0/e35/sall show all

**NOTE:** The **perccli /cx/sx show all** command shows tape drives information.

perccli /cx[/eall]/sall show jbod

This command shows the summary information for all the enclosures and physical drives connected to the controller. If you use the JBOD option, the command shows all Non-RAID disk(s) displayed in JBOD list. If physical disk is Non-RAID, type is set as JBOD and state as Online. ID displays the target ID Non-RAID disks.

Input example:

- perccli /c0/eall/sall show jbod

perccli /cx[/ex]/sx show jbod

This command shows the summary of the physical drive for a specified slot in the controller.

Input example:

- perccli /c0/e0/s4,5 show jbod

**Missing drives commands**

The PERC Command Line Tool supports the following commands to mark and replace missing physical drives:

- perccli /cx[/ex]/sx set offline
- perccli /cx[/ex]/sx set missing
- perccli /cx/dall show
- perccli /cx[/ex]/sx insert dg=a array=b row=c
- perccli /cx[/ex]/sx start rebuild

The detailed description for each command follows.

**perccli /cx[/ex]/sx set offline**

This command marks the drive in an array as offline.

**NOTE:** To set a drive that is part of an array as missing, first set it as offline. After the drive is set to offline, you can set the drive to missing.

Input example:

- perccli /c1/e56/s3 set offline

**perccli /cx[/ex]/sx set missing**

This command marks a drive as missing.

Input example:

- perccli /c0/s4 set missing

**perccli /cx/dall show**

This command shows the topology information of the drive group.

Input example:
perccli /c0/dall show

perccli /cx[/ex]/sx insert dg=a array=b row=c
This command replaces the configured drive that is identified as missing. User must manually start the rebuild.
Input example:
perccli /c0/e32/s4 insert dg=2 array=2 row=1

perccli /cx[/ex]/sx start rebuild
This command starts a rebuild operation for a drive.
Input example:
perccli /c0/e32/s4 start rebuild

Drive initialization commands
When you initialize drives, all the data from the drives is cleared. The PERC Command Line Tool supports the following commands to initialize drives:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx[/ex]/sx show initialization</td>
<td>This command shows the current progress of the initialization progress in percentage.</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx start initialization</td>
<td>This command starts the initialization process on a drive.</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx stop initialization</td>
<td>This command stops an initialization process running on the specified drive. A stopped initialization process cannot be resumed.</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.

perccli /cx[/ex]/sx show initialization
This command shows the current progress of the initialization progress in percentage.
Input example:
perccli /c0/e31/s4 show initialization

perccli /cx[/ex]/sx start initialization
This command starts the initialization process on a drive.
Input example:
perccli /c0/e31/s4 start initialization

perccli /cx[/ex]/sx stop initialization
This command stops an initialization process running on the specified drive. A stopped initialization process cannot be resumed.
Input example:
perccli /c0/e56/s1 stop initialization

Set drive state commands
The PERC Command Line Tool supports the following commands to set the status of physical drives:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx[/ex]/sx set jbod</td>
<td>Set JBOD status</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx set good [force]</td>
<td>Set good status</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx set offline</td>
<td>Set offline status</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx set online</td>
<td>Set online status</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sall set jbod</td>
<td>Set JBOD status for all drives</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx-y set jbod</td>
<td>Set JBOD status for drives X-Y</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.
**perccli /cx[/ex]/sx set jbod**
This command converts unconfigured good drive to Non-RAID disks.
Input example:
perccli /c1/e56/s3 set jbod

**perccli /cx[/ex]/sx set good [force]**
This drive changes the drive state to unconfigured good. If the drive has the operating system in it, use the force option.
Input example:
perccli /c1/e56/s3 set good

**perccli /cx[/ex]/sx set offline**
This command changes the drive state to offline.
Input example:
perccli /c1/e56/s3 set offline

**perccli /cx[/ex]/sx set online**
This command changes the drive state to online.
Input example:
perccli /c1/e56/s3 set online

**perccli /cx[/ex]/sall set jbod**
This command converts all unconfigured good drives to Non-RAID disks.
Input example:
perccli /c1/e56/sall set jbod

**perccli /cx[/ex]/sx-y set jbod**
This command converts all the selected unconfigured good drives to Non-RAID disks.
Input example:
perccli /c1/e56/s1-6 set jbod

**Locate drives commands**
The PERC Command Line Tool supports the following commands to locate a drive and activate the physical disk activity LED:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx[/ex]/sx start locate</td>
</tr>
<tr>
<td>perccli /cx[/ex]/sx stop locate</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.

**perccli /cx[/ex]/sx start locate**
This command locates a drive and activates the drive’s LED.
Input example:
perccli /c0/e56/s1 start locate

**perccli /cx[/ex]/sx stop locate**
This command stops a locate operation and deactivates the drive’s LED.
Prepare to remove drives commands

The PERC CLI supports the following commands to prepare the physical drive for removal:

- `perccli /cx[/ex]/sx spindown`
- `perccli /cx[/ex]/sx spinup`

The detailed description for each command follows.

**perccli /cx[/ex]/sx spindown**

This command spins down an unconfigured drive and prepares it for removal. The drive state is unaffiliated and it is marked offline.

Input example:

```
perccli /cx/e34/s4 spindown
```

**perccli /cx[/ex]/sx spinup**

This command spins up a spun-down drive and the drive state is unconfigured good.

Input example:

```
perccli /cx/e34/s4 spinup
```

Dimmer switch commands

The PERC Command Line Tool supports the following command to change the dimmer switch setting. The dimmer switch is the power saving policy on the drives.

**Change drive power settings commands**

**perccli /cx set dimmerswitch|ds=<on|off type=1|2|3|4>**

This command changes the power-saving properties on a controller for drives. See dimmer switch in the following table for values.

Input example:

```
perccli /c0 set ds=on type=1
```

Below is the list of combination of dimmer switch commands:

- `perccli /cx set ds=off type=1|2|3|4`
- `perccli /cx set ds=on type=1|2 [properties]`
- `perccli /cx set ds=on type=3|4 defaultldtype=<val> [properties]`
- `perccli /cx set ds [properties]`

**Table 10. Dimmer switch command options**

<table>
<thead>
<tr>
<th>Options</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dimmerswitch or ds</td>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>type</td>
<td></td>
<td>Specifies the type of drives that the dimmer switch feature is applicable. By default, it is activated for unconfigured drives, hot spare drives and virtual drives.</td>
</tr>
<tr>
<td>defaultldtype</td>
<td></td>
<td>Specifies the default logical drive type that is created by the dimmer switch option; set to none automatically.</td>
</tr>
</tbody>
</table>

1. Unconfigured
2. Hot spare
3. Virtual drive
4. All
Drive security commands

The PERC Command Line supports the following drive security command:

**perccli /cx[/ex]/sx show securitykey keyid**

This command shows the security key and key ID of the controller.

Input example:
```
perccli /c0/s4 show securityKey keyid
```

**perccli /cx[/ex]/sx set security=on**

This command sets the security key on JBOD or Non-RAID disks.

Input example:
```
perccli /c0/e2/s4 set security=on
```

**perccli /cx[/ex]/sx show jbod**

This command shows the summary of the non-RAID disks/JBOD drive for specified slot in the controller.

Input example:
```
perccli /c0/e2/s4 show jbod
```

**perccli /cx[/ex]/sx show jbod all**

This command shows all information of a non-RAID disks/JBOD drive for the specified slot in the controller. The `all` option in the command shows information for all slots on the controller. `x` stands for a number, a list of numbers, a range of numbers, or all numbers.

Input example:
```
perccli /c0/e2/s4 show jbod all
```

Drive erase commands

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/cx</td>
<td>Specifies a controller where <code>x</code> is the controller index.</td>
</tr>
<tr>
<td>/ex</td>
<td>Specifies an enclosure where <code>x</code> is the enclosure device ID.</td>
</tr>
<tr>
<td>/sx</td>
<td>Specifies a physical drive where <code>x</code> is the slot number.</td>
</tr>
</tbody>
</table>

Table 11. Conventions
The PERC Command Line supports the following drive erase commands:

```
perccli /cx[/ex]/sx secureerase[force]
perccli /cx[/ex]/sx stop erase
perccli /cx[/ex]/sx show erase
perccli /cx[/ex]/sx start erase[simple|normal|thorough|standard|threepass|crypto]
    [patternA=<val>] [patternB=<val>]
```

The detailed description for each command follows.

**perccli /cx[/ex]/sx secureerase [force]**

This command erases the drive's security configuration and securely erases data on a drive. You can use the force option as a confirmation to erase the data on the drive and the security information.

Input example:

```
perccli /c0/e25/s1 secureerase
```

**NOTE:** This command deletes data on the drive and the security configuration and this data is no longer accessible. This command is used for SED drives only.

**perccli /cx[/ex]/sx stop erase**

Stops secure erase on non-SED drives.

**perccli /cx[/ex]/sx show erase**

Displays the status as percentage of secure erase completed.

**perccli /cx[/ex]/sx start erase [simple| normal| thorough| standard| threepass| crypto] [patternA=<val>] [patternB=<val>]**

This command securely erases non-SED drives. The drive is written with erase patterns to ensure that the data is securely erased. You can use the following options with the start erase command:

**Table 12. Drive erase command options**

<table>
<thead>
<tr>
<th>Options</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>cx[/ex]/sx</strong></td>
<td>—</td>
<td>- /cx - specifies a controller where X is the controller index</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- /ex - specifies an enclosure where X is the enclosure device ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- /sx - specifies a physical drive where X is the slot number</td>
</tr>
<tr>
<td><strong>erase</strong></td>
<td>simple: Single pass, single pattern write</td>
<td>Secure erase type</td>
</tr>
<tr>
<td></td>
<td>normal: Three pass, three pattern write</td>
<td></td>
</tr>
<tr>
<td></td>
<td>thorough: Nine pass, repeats the normal write three times.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>standard: Applicable only for DFFs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>threepass: Three pass, pass1 random pattern write, pass 2, 3 write zero, verify</td>
<td></td>
</tr>
<tr>
<td><strong>crypto</strong></td>
<td></td>
<td>Applicable only for ISE capable drives.</td>
</tr>
<tr>
<td><strong>patternA</strong></td>
<td>8-bit value</td>
<td>Erase pattern A to overwrite the data.</td>
</tr>
<tr>
<td><strong>patternB</strong></td>
<td>8-bit value</td>
<td>Erase pattern B to overwrite the data.</td>
</tr>
</tbody>
</table>

Input example:

```
perccli /c0/e25/s1 start erase thorough patternA=10010011 patternB=11110000
```
Rebuild drives commands

The following commands rebuild drives in the PERC Command Line Tool:

- `perccli /cx[/ex]/sx pause rebuild`
- `perccli /cx[/ex]/sx resume rebuild`
- `perccli /cx[/ex]/sx show rebuild`
- `perccli /cx[/ex]/sx start rebuild`
- `perccli /cx[/ex]/sx stop rebuild`

**NOTE:** If enclosures are used to connect physical drives to the controller, specify the enclosure ID in the command.

The detailed description for each command follows.

**perccli /cx[/ex]/sx pause rebuild**

This command pauses an ongoing rebuild process. You can run this command only for a drive that is currently rebuilt.

Input example:

```
perccli /c0/s4 pause rebuild
```

**perccli /cx[/ex]/sx resume rebuild**

This command resumes a paused rebuild process. You can run this command only when a paused rebuild process for the drive exists.

Input example:

```
perccli /c0/s4 resume rebuild
```

**perccli /cx[/ex]/sx show rebuild**

This command shows the progress of the rebuild process in percentage.

Input example:

```
perccli /c0/s5 show rebuild
```

**perccli /cx[/ex]/sx start rebuild**

This command starts a rebuild operation for a drive.

Input example:

```
perccli /c0/s4 start rebuild
```

**perccli /cx[/ex]/sx stop rebuild**

This command stops a rebuild operation. You can run this command only for a drive that is currently rebuilt.

Input example:

```
perccli /c0/s4 stop rebuild
```

Drive copyback commands

The PErC Command Line Tool supports the following commands for drive copyback:

- `perccli /cx[/ex]/sx pause copyback`
- `perccli /cx[/ex]/sx resume copyback`
- `perccli /cx[/ex]/sx show copyback`
- `perccli /cx[/ex]/sx start copyback target=e:s`
- `perccli /cx[/ex]/sx stop copyback`

**NOTE:** In the copyback commands, `cx[/ex]/sx` indicates the source drive and `e:s` indicates the target drive.
perccli /cx[/ex]/sx pause copyback
This command pauses a copyback operation. You can run this command only when there is a copyback operation running.
Input example:
perccli /c0/e25/s4 pause copyback

perccli /cx[/ex]/sx resume copyback
This command resumes a paused copyback operation. You can run this command only when there is a paused copyback process for the drive.
Input example:
perccli /c0/e25/s4 resume copyback

perccli /cx[/ex]/sx show copyback
This command shows the progress of the copyback operation in percentage.
Input example:
perccli /c0/e25/s4 show copyback

perccli /cx[/ex]/sx start copyback target=e:s
This command starts a copyback operation for a drive.
Input example:
perccli /c0/e25/s4 start copyback target=25:8

perccli /cx[/ex]/sx stop copyback
This command stops a copyback operation. You can run this command only on drives that have the copyback operation running.
Input example:
perccli /c0/e25/s4 stop copyback

**NOTE:** A stopped rebuild process cannot be resumed.

Hot spare drive commands

The following commands create and delete hot spare drives:

perccli /cx[/ex]/sx add hotsparedrive [DGs=<N|0,1,2...>] [enclaffinity] [nonrevertible]
perccli /cx[/ex]/sx delete hotsparedrive

**NOTE:** If enclosures are used to connect the physical drives to the controller, specify the enclosure ID in the command.

The detailed description for each command follows.

perccli /cx[/ex]/sx add hotsparedrive [DGs=<N|0,1,2...>] [enclaffinity] [nonrevertible]
This command creates a hot spare drive. You can use the following options to create a hot spare drive:

**Table 13. Add hotsparedrive input options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dgs</td>
<td>Valid drive group number</td>
<td>Specifies the drive group to which the hot spare drive is dedicated.</td>
</tr>
<tr>
<td>enclaffinity</td>
<td>Valid enclosure number</td>
<td>Specifies the enclosure with which the hot spare is associated. If this option is specified, affinity is set; if it is not specified, there is no affinity.</td>
</tr>
<tr>
<td>Option</td>
<td>Value range</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>nonrevertible</td>
<td></td>
<td>Sets the drive as a nonrevertible hotspare.</td>
</tr>
</tbody>
</table>

**perccli /cx[/ex]/sx add hotsparedrive**

This command adds a hot spare drive.

Input example:

```bash
perccli /c0/e3/s4,5 add hotsparedrive
```

This command sets the drives /c0/e3/s4,5 as global hot spare.

Input example:

```bash
perccli /c0/e3/s6,8 add hotsparedrive dgs=0,1
```

This command sets /c0/e3/s6,8 as dedicated hot spare for disk groups 0,1.

**perccli /cx[/ex]/sx delete hotsparedrive**

This command deletes a hot spare drive.

Input example:

```bash
perccli /c0/e3/s4,5 delete hotsparedrive
```

### Virtual drives commands

The PERC Command Line Tool supports the following virtual drive commands. The following table describes frequently used virtual drive commands.

**Table 14. Virtual drives commands quick reference table**

<table>
<thead>
<tr>
<th>Commands</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>See Table 16. Add RAID 0 configuration input options.</td>
<td>Creates virtual drives.</td>
</tr>
<tr>
<td>delete</td>
<td>force: Deletes the virtual drive where operating system is present.</td>
<td>Deletes a virtual drive.</td>
</tr>
<tr>
<td>set</td>
<td>See Table 16. Add RAID 0 configuration input options, and Change virtual drive properties commands.</td>
<td>Sets virtual drive properties.</td>
</tr>
<tr>
<td>show</td>
<td>all: Shows all properties of the virtual drive.</td>
<td>Shows virtual drive information.</td>
</tr>
</tbody>
</table>

### Add virtual drives commands

The PERC Command Line Tool supports the following commands to add virtual drives:

```bash
perccli /cx add vd r[0|1|5|6|10|50|60] [Size=<VD1_Sz>,<VD2_Sz>,...|all] [name=<VDNAME1>,..] [drives=e:s|x|e:s-x,y |SED] [pdcache=on|off|default] [pi] [DimmerSwitch(ds)=default|automatic(auto)] [nora|ra] [direct|cached] [CachedBadBBU|NoCachedBadBBU] [Strip=64|128|256|512|1024] [AfterVd=X] [EmulationType=0|1|2] [Spares = [e:s]|e:s-x|e:s-x,y] [force][ExclusiveAccess] [Cbsize=0|1|2 Cbmode=0|1|7] perccli /cx add vd each r0 [name=<VDNAME1>,..] [drives=e:s|x|e:s-x,y] [SED] [pdcache=on|off|default] [pi] [DimmerSwitch(ds)=default] [auto|manual] [max|maxnocache] [wt|wb|fwb] [nora|ra] [direct|cached] [CachedBadBBU|NoCachedBadBBU]
```
This command creates a RAID configuration. You can use the following options to create the RAID volume:

1. **NOTE:** * indicates default values.

The detailed description for each command follows.

```plaintext
percli /cx add vd type=raid[0|1|5|6|10|50|60][Size=<VD1_Sz>,<VD2_Sz>,..|*all] [name=<VDNAME1>,..] drives=e:s|e:s-x|e:s-x,y [PDperArray=x] [SED] [pdcache=on|off]*default [pi] [DimmerSwitch(ds)=default|automatic(auto)|*none] [Maximum(max) [MaximumWithoutCaching(maxnocache)] [wt|*wb] [nora|*ra] [*direct|cached] [CachedBadBBU|*NoCachedBadBBU] [Strip=<8|16|32|64|128|256|1024>] [AfterVd=X] [Spares = [e:]s|[e:]s-x|[e:]s-x,y] [force]
```

### Table 15. Add RAID configuration input options

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>type</strong></td>
<td>RAID [0</td>
<td>1</td>
</tr>
<tr>
<td><strong>size</strong></td>
<td>Maximum size based on the physical drives and RAID level.</td>
<td>Sets the size of each virtual drive. The default value is for the capacity of all referenced disks.</td>
</tr>
<tr>
<td><strong>name</strong></td>
<td>15 characters of length.</td>
<td>Specifies the drive name for each virtual drive.</td>
</tr>
</tbody>
</table>
| **drives**      | Valid enclosure number and valid slot numbers for the enclosure. | In e:s|e:s-x|e:s-x,y:  
<p>|                 |                                  | • e specifies the enclosure ID.                                             |
|                 |                                  | • s represents the slot in the enclosure.                                   |
|                 |                                  | • e:s-x is the range convention used to represent slots s to x in the enclosure e. |
| <strong>pdperarray</strong>  | 0 to 15.                         | Specifies the number of physical drives per array. The default value is automatically chosen. |
| <strong>sed</strong>         | —                                | Creates security-enabled drives.                                           |
| <strong>pdcache</strong>     | on|off|default.                       | Enables or disables PD cache.                                              |
| <strong>pi</strong>          | —                                | Enables protection information.                                            |
| <strong>dimmerswitch</strong>| default: Logical device uses controller default power-saving policy.       | Specifies the power-saving policy. Sets to default automatically.          |
|                 | automatic (auto): Logical device power savings are managed by firmware.    |                                                                             |
|                 | none: No power-saving policy.    |                                                                             |
|                 | maximum (max): Logical device uses maximum power savings.                  |                                                                             |
|                 | MaximumWithoutCaching (maxnocache): Logical device does not cache write to maximize power savings. |
| <strong>wt|wb</strong>         | wt: Write through.               | Enables write through. Write back is the default.                         |
| <strong>nora|ra</strong>         | ra: Read ahead.                  | Disables read ahead. Enabled is the default.                               |
| <strong>cachedbadbbu|nocachedbadbbu</strong> | cachedbadbbu: Enable bad BBU caching. | Enables caching when BBU is not functioning. Disabled is the default. |
| <strong>strip</strong>       | 8, 16, 32, 64, 128, 256, 512, 1024. | Sets the strip size for the RAID configuration.                            |</p>
<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aftervd</td>
<td>Valid virtual drive number.</td>
<td>Creates the VD in the adjacent free slot next to the specified VD.</td>
</tr>
<tr>
<td>spares</td>
<td>Number of spare physical drives present.</td>
<td>Specifies the physical drives that are to be assigned to a disk group for spares.</td>
</tr>
<tr>
<td>force</td>
<td>—</td>
<td>Forces a security-capable physical drive to be added to a drive group without security.</td>
</tr>
</tbody>
</table>

```perl
perclli /cx add vd each r0 [name=<VDNAME1>,...] [drives=e:s|e:s-x|e:s-x,y] [SED] [pdcache=on|off|default] [pi] [DimmerSwitch(ds)=default|automatic(auto)|none|maximum(max)|MaximumWithoutCaching(maxnocache)] [wt|wb] [nora|ra] [direct|cached] [CachedBadBBU|NoCachedBadBBU] [Strip=<64|128|256|512|1024>] [EmulationType=0|1|2] [ExclusiveAccess] [Cbsize=0|1|2] [Cbmode=0|1|7]
```

This command creates a RAID 0 configuration for each disk specified in the `drives` option. You can use the following options to create the RAID volume:

**Table 16. Add RAID 0 configuration input options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>RAID [0</td>
<td>1</td>
</tr>
<tr>
<td>size</td>
<td>Maximum size based on the physical drives and RAID level.</td>
<td>Sets the size of each virtual drive. The default value is for the capacity of all referenced disks.</td>
</tr>
<tr>
<td>name</td>
<td>15 characters of length.</td>
<td>Specifies the drive name for each virtual drive.</td>
</tr>
</tbody>
</table>
| drives   | Valid enclosure number and valid slot numbers for the enclosure.             | In e:s|s|x:e:s-x,y:  
  - e specifies the enclosure target.  
  - s represents the disk slot number.  
  - e:s-x is the range of disk slot numbers.  
  - e:s-x,y is the range of disk slot numbers plus the disk with a slot number out of the specified range.  
  If you replace s-x with 0-9, it will provide 10 RAID 0 virtual disks with each using one disk. |
<p>| pdperarray | 0 to 15.                                                              | Specifies the number of physical drives per array. The default value is automatically chosen. |
| sed      | —                                                                            | Creates security-enabled drives.                                           |
| pdcache  | on|off|default.                                                               | Enables or disables PD cache.                                             |
| pi       | —                                                                            | Enables protection information.                                           |
| dimmerswitch | default: Logical device uses controller default power-saving policy.       | Specifies the power-saving policy. Sets to default automatically.          |
|          | automatic (auto): Logical device power savings are managed by firmware.     |                                                                           |
|          | none: No power-saving policy.                                               |                                                                           |
|          | maximum (max): Logical device uses maximum power savings.                  |                                                                           |
|          | MaximumWithoutCaching (maxnocache): Logical device does not cache write to maximize power savings. |
| wt|wb      | wt: Write through.                                                         | Enables write through. Write back is the default.                        |
|         | wb: Write back.                                                            |                                                                           |
| nora|ra      | ra: Read ahead.                                                            | Enables read ahead. Enabled is the default.                              |</p>
<table>
<thead>
<tr>
<th>Option</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nora</td>
<td>nora: No read ahead.</td>
<td></td>
</tr>
<tr>
<td>cachedbadbbu</td>
<td>cachedbadbbu: Enable bad BBU caching.</td>
<td>Enables caching when BBU is not functioning. Disabled is the default.</td>
</tr>
<tr>
<td>nocachedbadbbu</td>
<td>nocachedbadbbu: Disable bad BBU caching.</td>
<td></td>
</tr>
<tr>
<td>strip</td>
<td>8, 16, 32, 64, 128, 256, 512, 1024.</td>
<td>Sets the strip size for the RAID configuration.</td>
</tr>
<tr>
<td>aftervd</td>
<td>Valid virtual drive number.</td>
<td>Creates the VD in the adjacent free slot next to the specified VD.</td>
</tr>
<tr>
<td>spares</td>
<td>Number of spare physical drives present.</td>
<td>Specifies the physical drives that are to be assigned to a disk group for spares.</td>
</tr>
<tr>
<td>force</td>
<td>—</td>
<td>Forces a security-capable physical drive to be added to a drive group without security.</td>
</tr>
</tbody>
</table>

Input example:
```
perccli /c0 add vd type=raid10 size=2gb,3gb,4gb names=tmp1,tmp2,tmp3 drives=252:2-3,5,7 pdperarray=2
```

## Delete virtual drives commands

The PERC Command Line Tool supports the following virtual drive delete commands:

```
perccli /cx/vx|vall del
perccli /cx/vx|vall del force
```

### NOTE: If the virtual drive has user data, you must use the force option to delete the virtual drive.

A virtual drive with a valid master boot record (MBR) and a partition table is considered to contain user data.

If you delete a virtual drive with a valid MBR without erasing the data and then create a new virtual drive using the same set of physical drives and the same RAID level as the deleted virtual drive, the old unerased MBR still exists at block0 of the new virtual drive, which makes it a virtual drive with valid user data. Therefore, you must provide the force option to delete this newly created virtual drive.

The detailed description for each command follows.

### perccli /cx/vx|vall del

This command deletes a particular virtual drive or, when the vall option is used, all the virtual drives on the controller are deleted.

Input example:
```
perccli /c0/v2 del
```

### NOTE: This command deletes virtual drives. Data located on these drives will no longer be accessible.

### NOTE: This command deletes virtual drives. Data located on these drives will no longer be accessible.

### perccli /cx/vx|vall del force

This command deletes a virtual drive only after the cache flush is completed. With the force option, the command deletes a virtual drive without waiting for the cache flush to complete.

Input example:
```
perccli /c0/v2 del force
```

### NOTE: This command deletes the virtual drive where the operating system is present. Data located on these drives and the operating system of the drive will no longer be accessible.
Delete Non-RAID disks

The PERC Command Line Tool supports the following Non-RAID disks delete commands:

- `perccli /cx[/ex]/sx del jbod [force]`
- `perccli /cx[/ex]/sall del jbod [force]`
- `perccli /cx[/ex]/sx-y del jbod [force]`

When in eHBA mode, this command deletes a particular Non-RAID disk (listed as JBOD drive) or when the `sall` option is used, all the non-RAID disks on the controller are deleted. The `x` stands for a number, list of numbers, range of numbers, or all numbers. The `force` option should be used only if the user needs to delete a Non-RAID drive with any partition.

Virtual drive show commands

The PERC Command Line Tool supports the following virtual drive show commands:

- `perccli /cx/vx show`
- `perccli /cx/vx show all`

The detailed description for each command follows.

**perccli /cx/vx show**

This command shows the summary of the virtual drive information.

Input example:

```
perccli /c0/v0 show
```

**perccli /cx/vx show all**

This command shows all virtual drive information, which includes virtual drive information, physical drives used for the virtual drives, and virtual drive properties.

Input example:

```
perccli /c0/v0 show all
```

Preserved cache commands

If a virtual drive becomes offline or is deleted because of missing physical disks, the controller preserves the dirty cache from the virtual disk. The PERC Command Line Tool supports the following commands for preserved cache:

- `perccli /cx/vx delete preservedCache [force]`
- `perccli /cx show preservedCache`

The detailed description for each command follows.

**perccli /cx/vx delete preservedcache**

This command deletes the preserved cache for a particular virtual drive on the controller in missing state. Use the `force` option to delete the preserved cache of a virtual drive in offline state.

Input example:

```
perccli /c0/v1 delete preservedcache
```

**perccli /cx show preservedCache**

This command shows the virtual drive that has preserved cache and whether the virtual drive is offline or missing.

Input example:

```
perccli /c0 show preservedCache
```
Change virtual drive properties commands

**NOTE:** In the following, /cx specifies the controller, where x is the controller index, while /vx specifies the virtual drive, where x is the virtual drive ID.

The PERC Command Line Tool supports the following commands to change virtual drive properties:

```
perccli /cx/vx set accesspolicy=RW|RO|Blocked|RmvBlkd
perccli /cx/vx set bootdrive=<on|off>
perccli /cx/vx set cbsize=0|1|2 cbmode=0|1|7
perccli /cx/vx set ds=Default|Auto|None|Max|MaxNoCache
perccli /cx/vx set iopolicy=Cached|Direct
perccli /cx/vx set name=<NameString>
perccli /cx/vx set pdcache=On|Off|Default
perccli /cx/vx set pi=Off
perccli /cx/vx set rdcache=RA|NoRA
perccli /cx/vx set wrcache=WT|WB|FWB
```

The detailed description for each command follows.

**perccli /cx/vx set accesspolicy=RW|RO|Blocked|RmvBlkd**

This command sets the access policy on a virtual drive to read write, read only, or blocked or rmvblkd (remove blocked).

Input example:

```
perccli /c0/v0 set accesspolicy=rw
```

Options:

- **RW** - Access is Read Write
- **RO** - Access is Read Only
- **Blocked** - Access is Blocked
- **RmvBlkd** - Remove Blocked Access

**perccli /cx/vx set bootdrive=<on|off>**

Sets or unsets a virtual drive as the boot drive.

**NOTE:** Set bootdrive is applicable only in legacy BIOS mode.

Input example:

```
perccli /c0/v0 set bootdrive=on
```

**perccli /cx/vx set cbsize=0|1|2 cbmode=0|1|7**

This command sets the cache bypass size and cache bypass mode on a virtual drive.

Input example:

```
perccli /c0/v0 set cbsize=0 cbmode=7
```

Options:

- **cbsize:**
  - 0 — 64k cache bypass
  - 1 — 128k cache bypass
  - 2 — 256k cache bypass

- **cbmode:**
  - 0 — Enable intelligent mode cache bypass
  - 1 — Enable standard mode cache bypass
  - 7 — Disable cache bypass
**perccli /cx/vx set ds=<Default|Auto|None|Max|MaxNoCache>**

This command changes the power-saving properties on a virtual drive.

**Input example:**

```
perccli /c0/v0 set ds=Default
```

**Options:**

- Default — Controller default power saving options are applied
- Auto — Power savings is managed by firmware
- None — Power savings is disabled
- Maximum — Maximum power savings options are applied
- MaxNoCache — Maximum power savings with no caching of writes are applied

**perccli /cx/vx set iopolicy=<cached|direct>**

This command sets the I/O policy on a virtual drive to cached I/O or direct I/O.

**Input example:**

```
perccli /c0/v0 set iopolicy=cached
```

**Options:**

- Cached — I/Os are cached
- Direct — I/Os are not cached

**perccli /cx/vx set name=<NameString>**

This command names a virtual drive. The name is restricted to 15 characters.

**Options:**

- NameString — VD name

**perccli /cx/vx set pdcache=<on|off|default>**

This command sets the current disk cache policy on a virtual drive to on, off, or default setting.

**Input example:**

```
perccli /c0/v0 set pdcache=on
```

**Options:**

- On — Enables pd caching
- Off — Disables pd caching
- Default — pd caching is set to default

**perccli /cx/vx set pi=Off**

This command disables the data protection of a virtual drive.

**Input example:**

```
perccli /cx/vx set pi=Off
```

**Options:**

- Off — Disables data protection

**perccli /cx/vx set rdcache=<ra|nora>**

This command sets the read cache policy on a virtual drive to read ahead or no read ahead.

**Input example:**

```
perccli /c0/v0 set rdcache=nora
```
Options:
RA = Read ahead
NORA = No read ahead

**perccli /cx/vx set wrcache=\<WT|WB|FWB>**

This command sets the write cache policy on a virtual drive to write back, write through, or always write back.

Input example:
perccli /c0/v0 set wrcache=wt

Options:
WT — Write through
WB — Write back
FWB — Force write back even in case of bad BBU

### Virtual drive initialization commands

The PERC Command Line Tool supports the following commands to initialize virtual drives:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx/vx show init</td>
<td>This command shows the initialization progress of a virtual drive in percentage.</td>
</tr>
<tr>
<td>perccli /cx/vx start init [full][Force]</td>
<td>This command starts the initialization of a virtual drive. The default initialization type is fast initialization. If the full option is specified, full initialization of the virtual drive starts.</td>
</tr>
<tr>
<td>perccli /cx/vx stop init</td>
<td>This command stops the initialization of a virtual drive. A stopped initialization cannot be resumed.</td>
</tr>
</tbody>
</table>

**NOTE:** If the virtual drive has user data, you must use the **force** option to initialize the virtual drive. A virtual drive with a valid MBR and partition table is considered to contain user data.

The detailed description for each command follows.

**perccli /cx/vx show init**

This command shows the initialization progress of a virtual drive in percentage.

Input example:
perccli /c0/v2 show init

**perccli /cx/vx start init [full]**

This command starts the initialization of a virtual drive. The default initialization type is fast initialization. If the full option is specified, full initialization of the virtual drive starts.

Input example:
perccli /cx/vx start init [full]

**perccli /cx/vx stop init**

This command stops the initialization of a virtual drive. A stopped initialization cannot be resumed.

Input example:
perccli /c0/v0 stop init

### Virtual drive erase commands

The PERC Command Line Tool supports the following command to erase virtual drives:

**perccli /cx/vx erase [force]**

This command erases the data on the virtual drive. You can use the **force** option as a confirmation to erase the data on the drive and the security information.
NOTE: If the virtual drive has user data, you must use the "force" option to erase the virtual drive. A virtual drive with a valid MBR and partition table is considered to contain user data.

**perccli /cx/vx show erase**

This command shows the progress of drive's security configuration and erases data in percentage.

Input example:

```
perccli /c0/v1 show erase
```

**perccli /cx/vx stop erase**

This command stops the erase operation.

Input example:

```
perccli /c0/v1 stop erase
```

**perccli /cx/vx start erase [simple|normal|thorough|standard|threepass] [patternA=<val>] [patternB=<val>]**

This command securely erases non-SED drives. The drive is written with erase patterns to ensure that the data is securely erased. You can use the following options with the start erase command:

<table>
<thead>
<tr>
<th>Options</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cx/vx</td>
<td>—</td>
<td>• /cx: specifies a controller where X is the controller index</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• /vx: specifies the virtual drive where X is the virtual drive ID</td>
</tr>
</tbody>
</table>

| erase            | simple: Single pass, single pattern write. |
|                  | • normal: Three pass, three pattern write |
|                  | • thorough: Nine pass, repeats the normal write three times |
|                  | • threepass: Three pass, pass1 random pattern write, pass 2, 3 write zero, verify |

| patternA         | 8-bit value | Erase pattern A to overwrite the data. |
| patternB         | 8-bit value | Erase pattern B to overwrite the data. |

### Virtual drive migration commands

NOTE: The virtual drive migration commands are not supported in Embedded MegaRAID.

The PERC Command Line Tool supports the following commands for virtual drive migration (reconstruction):

```
perccli /cx/vx show migrate
perccli /cx/vx start migrate type=raidx [option=add] [option=remove] drives=[e:]s| [e:]s-x| [e:]s-y [Force]
```

The detailed description for each command follows.
**perccli /cx/vx show migrate**

This command shows the progress of the virtual drive migrate operation in percentage.

Input example:

```
perccli /c0/v0 show migrate
```

**perccli /cx/vx start migrate type=raidx [option=add|remove drives=[e:]s| [e:]s-x| [e:]s-x,y] [Force]**

This command starts the reconstruction on a virtual drive to the specified RAID level by adding or removing disks from the existing virtual drive. You can use the following options with the start migrate command:

**Table 18. Virtual drive migration command options**

<table>
<thead>
<tr>
<th>Options</th>
<th>Value range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type=raidx</td>
<td>RAID [0</td>
<td>1</td>
</tr>
<tr>
<td>[option=add</td>
<td>remove drives=[e:]s</td>
<td>[e:]s-x</td>
</tr>
<tr>
<td></td>
<td>remove: Removes the specified PD from the migrated raid level.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>drives: Specifies the list of PD’s which needs to be added or removed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>force: If specified, then migration will start even if any drive in the DG is secured by FDE.</td>
<td></td>
</tr>
</tbody>
</table>

Virtual drive migration can be done between the following RAID levels:

**Table 19. Virtual drive migration table**

<table>
<thead>
<tr>
<th>Initial RAID level</th>
<th>Migrated RAID level</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAID 0</td>
<td>RAID 0</td>
</tr>
<tr>
<td>RAID 0</td>
<td>RAID 1</td>
</tr>
<tr>
<td>RAID 0</td>
<td>RAID 5</td>
</tr>
<tr>
<td>RAID 0</td>
<td>RAID 6</td>
</tr>
<tr>
<td>RAID 1</td>
<td>RAID 0</td>
</tr>
<tr>
<td>RAID 1</td>
<td>RAID 5</td>
</tr>
<tr>
<td>RAID 1</td>
<td>RAID 6</td>
</tr>
<tr>
<td>RAID 5</td>
<td>RAID 0</td>
</tr>
<tr>
<td>RAID 5</td>
<td>RAID 5</td>
</tr>
<tr>
<td>RAID 5</td>
<td>RAID 6</td>
</tr>
<tr>
<td>RAID 6</td>
<td>RAID 0</td>
</tr>
<tr>
<td>RAID 6</td>
<td>RAID 5</td>
</tr>
<tr>
<td>RAID 6</td>
<td>RAID 6</td>
</tr>
<tr>
<td>RAID 10</td>
<td>RAID 10</td>
</tr>
</tbody>
</table>

Input example:

```
perccli /c0/v3 start migrate type=r5 option=add drives=134:1-3
```
Virtual drive consistency check commands

The PERC Command Line Tool supports the following commands for virtual drive consistency checks:

- **perccli /cx/vx pause cc**
- **perccli /cx/vx resume cc**
- **perccli /cx/vx show cc**
- **perccli /cx/vx start cc [force]**
- **perccli /cx/vx stop cc**

The detailed description for each command follows.

**perccli /cx/vx pause cc**

This command pauses an ongoing consistency check process. You can resume the consistency check at a later time. You can run this command only on a virtual drive that has a consistency check operation running.

**Input example:**

```
perccli /c0/v4 pause cc
```

**perccli /cx/vx resume cc**

This command resumes a suspended consistency check operation. You can run this command on a virtual drive that has a paused consistency check operation.

**Input example:**

```
perccli /c0/v4 resume cc
```

**perccli /cx/vx show cc**

This command shows the progress of the consistency check operation in percentage.

**Input example:**

```
perccli /c0/v5 show cc
```

**perccli /cx/vx start cc force**

This command starts a consistency check operation for a virtual drive. Typically, a consistency check operation is run on an initialized virtual drive. Use the **force** option to run a consistency check on an uninitialized drive.

**Input example:**

```
perccli /c0/v4 start cc
```

**perccli /cx/vx stop cc**

This command stops a consistency check operation. You can run this command only for a virtual drive that has a consistency check operation running.

**Input example:**

```
perccli /c0/v4 stop cc
```

**NOTE:** You cannot resume a stopped consistency check process.

Background initialization commands

The PERC Command Line Tool supports the following commands for background initialization:

- **perccli /cx/vx resume bgi**
- **perccli /cx/vx set autobgi=on|off**
- **perccli /cx/vx show autobgi**
- **perccli /cx/vx show bgi**
The detailed description for each command follows.

**percli /cx/vx resume bgi**
This command resumes a suspended background initialization operation.
Input example:
percli /c0/v0 resume bgi

**percli /cx/vx set autobgi=on|off**
This command sets the auto background initialization setting for a virtual drive to on or off.
Input example:
percli /c0/v0 set autobgi=on

**percli /cx/vx show autobgi**
This command shows the background initialization setting for a virtual drive.
Input example:
percli /c0/v0 show autobgi

**percli /cx/vx show bgi**
This command shows the background initialization progress on the specified virtual drive in percentage.
Input example:
percli /c0/v0 show bgi

**percli /cx/vx stop bgi**
This command stops a background initialization operation. You can run this command only for a virtual drive that is currently initialized.
Input example:
percli /c0/v4 stop bgi

**percli /cx/vx pause bgi**
This command suspends a background initialization operation. You can run this command only for a virtual drive that is currently initialized.
Input example:
percli /c0/v4 pause bgi

### Virtual drive expansion commands

The PERC Command Line Tool supports the following commands for virtual drive expansion:

```
percli /cx/vx expand size=<value> [expandarray]
percli /cx/vx/vall show expansion
```

The detailed description for each command follows.

**percli /cx/vx expand size=<value> [expandarray]**
This command expands the virtual drive within the existing array or if you replace the drives with drives larger than the size of the existing array. The value of the expand size is in GB. If the expandarray option is specified, the existing array is expanded. If this option is not specified, the virtual drive is expanded.
perccli /cx/vx show expansion
This command shows the expansion information on the virtual drive with and without array expansion.
Input example:
perccli /c0/v0 show expansion

Foreign configurations commands
The PERC Command Line Tool supports the following commands to view, import, and delete foreign configurations:

- `perccli /cx/fx|fall del|delete [ securitykey=ssssssssssss ]`
- `perccli /cx/fx|fall import [preview][ securitykey=ssssssssssss ]`
- `perccli /cx/fx|fall show [all] [ securitykey=ssssssssssss ]`

**NOTE:** Provide the security key when importing a locked foreign configuration created in a different machine that is encrypted with a security key.

The detailed description for each command follows.

- `perccli /cx/fx|fall del|delete [ securitykey=ssssssssssss ]`
  This command deletes the foreign configuration of a controller. Input the security key if the controller is secured.
  Input example:
  `perccli /c0/fall delete`

- `perccli /cx/fx|fall import [preview] [ securitykey=ssssssssssss ]`
  This command imports the foreign configurations of a controller. The `preview` option shows a summary of the foreign configuration before importing it.
  Input example:
  `perccli /c0/fall import`

- `perccli /cx/fx|fall show [all] [ securitykey=ssssssssssss ]`
  This command shows the summary of the entire foreign configuration for a particular controller. The `all` option shows all the information of the entire foreign configuration.
  **NOTE:** The EID:Slot column is populated for the foreign PDs that are locked.
  Input example:
  `perccli /c0/fall show preview foreign`
  `perccli /c0/fall import preview`
  `perccli /c0/fall show all`

**BIOS-related commands**
The PERC Command Line Tool supports the following BIOS commands:

- `perccli /cx set bios [state=<on|off>] [Mode=<SOE|PE|IE|SME>] [abs=<on|off>] [DeviceExposure=<value>]`
- `perccli /cx show bios`

The detailed description for each command follows.
perccli /cx set bios [state=<on|off>] [Mode=<SOE|PE|IE|SME>] [abs=<on|off>] [DeviceExposure=<value>]

This command sets the BIOS controller property to on or off. The Mode sets the BIOS boot mode.

Only the following combinations are supported:

- perccli /cx set bios state=<on|off>
- perccli /cx set bios Mode=<SOE|PE|IE|SME>
- perccli /cx set bios abs=<on|off>
- perccli /cx set bios DeviceExposure=<value>

Options

SOE — Stop on errors
PE — Pause on errors
IE — Ignore errors
SME — Safe mode on errors
abs — Enables/Disables the auto boot select

DeviceExposure — Number of devices to be exposed: value range is 0–255

value 0 and 1: Expose all
value 2 — 255: Actual number of devices to be exposed

Input example:
perccli /c0 set bios=on

perccli /cx show bios

This command displays the value of the controller BIOS.

Input example:
perccli /c0 show bios

OPROM BIOS commands

The PERC Command Line Tool supports the following OPROM BIOS commands:

perccli /cx[/ex]/sx set bootdrive=on|off
perccli /cx/vx set bootdrive=on|off
perccli /cx show bootdrive

The detailed description for each command follows.

perccli /cx[/ex]/sx set bootdrive=on|off

This command sets the specified physical drive as the boot drive. During the next reboot, the BIOS looks for a boot sector in the specified physical drive. The eHBA mode supports setting a Non-RAID disk as a boot drive.

Input example:
perccli /c0[/e32]/s4 set bootdrive=on

perccli /cx/vx set bootdrive=on|off

This command sets the specified virtual drive as the boot drive. During the next reboot, the BIOS looks for a boot sector in the specified virtual drive.

Input example:
perccli /c0/v0 set bootdrive=on
**perccli/cx/vx show bootdrive**

This command shows the boot drive for the controller. The boot drive can be a physical drive or a virtual drive. The existing configured boot drives will be displayed.

Input example:

```
perccli /c0/v0 show bootdrive
```

**Drive group commands**

This section describes the drive group commands.

**Drive group show**

The PERC Command Line Tool supports the following drive group commands:

```
perccli /cx/dx show
perccli /cx/dx show all
perccli /cx/dall show mirror
perccli /cx/dall split mirror
perccli /cx/dall add mirror src=<val> [force]
perccli /cx/dx set security=on
```

**NOTE:** In the following, `/cx` specifies the controller where `x` is the controller index, while the value `/dx` specifies the disk group where `x` is the disk group index.

**perccli /cx/dx show**

This command shows the topology information of the drive group.

Input example:

```
perccli /c0/dall show
```

**perccli /cx/dall show mirror**

This command displays information about the mirror associated with drive group.

Input example:

```
perccli /c0/dall show mirror
```

**perccli /cx/dall split mirror**

This command splits apart the mirror virtual drives.

Input example:

```
perccli /c0/dall split mirror
```

**perccli /cx/dall add mirror src=<val> [force]**

This command joins the virtual drive with its mirror.

Input example:

```
perccli /c0/dall add mirror src=<2>
```

Options for `<val>`:

- **0** — Data will be copied from existing virtual drive to drives.
- **1** — Data will be copied from drives to virtual drive.
- **2** — Broken mirror is imported as a new virtual drive.
perccli /cx/dx set security=on
This command enables security on the specified drive group.
Input example:
perccli /c0/d0 set security=on

perccli /cx/dx show all
This command shows physical and virtual drive information for the disk group.
Input example:
perccli /c0/dall show all

BBU commands
The PERC Command Line Tool supports the following battery backup unit (BBU) commands:

perccli /cx/bbu set [learnDelayInterval=<val>|bbuMode=<val>|learnStartTime=[DDDHH|off]|autolearnmode=<val>|powermode=sleep|writeaccess=sealed]
perccli /cx/bbu show
perccli /cx/bbu show all
perccli /cx/bbu show learn
perccli /cx/bbu show properties
perccli /cx/bbu show status
perccli /cx/bbu start learn

In the following, /cx specifies a controller where x is the controller index, and /bbu signifies a battery backup unit.
The detailed description for each command follows:

perccli /cx/bbu set <options>
This command sets bbu properties on the controller bbu.
Options:
• learnDelayInterval=<val>: number of hours to delay a learn cycle, not greater than 7 days
• bbuMode=<val>: val range 0–255
• autolearnmode=<val>: 0 — Enabled, 1 — Disabled, 2 — WarnViaEvent
• learnStartTime=[DD HH|off]: DDD — day of week {SUN, MON, . . . SAT} HH — 0–23 hours, off: Sets learn start to OFF
• powermode=sleep
• writeaccess=sealed

perccli /cx/bbu show
This command shows the summary information for the BBU of a controller.
Input example:
perccli /c0/bbu show

perccli /cx/bbu show all
This command shows all the information of a BBU.
Input example:
perccli /c0/bbu show all
perccli /cx/bbu show learn

perccli /cx/bbu show properties
This command shows the BBU Learn properties for a controller.
Input example:
perccli /c0/bbu show properties

perccli /cx/bbu show status
This command shows summary information for the BBU of a controller.
Input example:
perccli /c0/bbu show status

perccli /cx/bbu start learn
This command starts the BBU learning cycle.
Input example:
perccli /c0/bbu start learn

Enclosure commands
The PERC Command Line Tool supports the following enclosure commands:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx/ex show</td>
</tr>
<tr>
<td>perccli /cx/ex show all</td>
</tr>
<tr>
<td>perccli /cx/ex show phyerrorcounters</td>
</tr>
<tr>
<td>perccli /cx/ex show status</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.

perccli /cx/ex show
Input example:
perccli /c1/e1 show

perccli /cx/ex show all
This command shows the status of each model in the enclosure.
Input example:
perccli /c0/e0 show all

perccli /cx /ex show phyerrorcounters
Input example:
perccli /c0 /e0 show phyerrorcounters
**perccli /cx/ex show status [extended]**

This command shows the enclosure status and the status of all the enclosure elements.

Input example:
perccli /c0/e0 show status

---

**PHY commands**

The PERC Command Line Tool supports the following PHY commands:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx/px</td>
</tr>
<tr>
<td>perccli /cx/px</td>
</tr>
<tr>
<td>perccli /cx/px</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.

**perccli /cx/px|pall set linkspeed=0(auto)|1.5|3|6|12**

This command sets the PHY link speed. You can set the speed to 1.5 Gb/s, 3 Gb/s, 6 Gb/s, or 12 Gb/s. The linkspeed is set to **auto** when you specify `linkspeed = 0`.

Input example:
perccli /c0/p0 set linkspeed=1.5

**perccli /cx/px|pall show**

This command shows the basic PHY layer information.

Input example:
perccli /c1/p0 show

**perccli /cx/px|pall show all**

This command shows all the PHY layer information.

Input example:
perccli /c1/p0 show all

---

**Logging commands**

The PERC Command Line Tool supports the following commands to generate and maintain log files:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx delete events</td>
</tr>
<tr>
<td>perccli /cx show events file=&lt;absolute path&gt;</td>
</tr>
<tr>
<td>perccli /cx show eventloginfo</td>
</tr>
</tbody>
</table>

The detailed description for each command follows.

**perccli /cx delete events**

This command deletes all records in the event log.

Input example:
perccli /c0 delete events
perccli /cx show events file=<absolute path>

This command prints the system log to a text file and saves the file in the specified location.
Input example:
perccli /c0 show events file=C:\Users\Administrator\Documents\eventreport

perccli /cx show eventloginfo

This command shows the history of log files generated.
Input example:
perccli /c0 show eventloginfo type=config

PERC CLI command examples

You can use the Dell PowerEdge RAID Controller (PERC) Command Line Interface (CLI) to manage RAID controllers, configure PERC cards, and perform a variety of controller and enclosure specific operations.

Getting a complete list of CLI commands

To view a full list of available CLI commands, use one of the following CLI commands:

perccli64.exe –help > [filename]
perccli64.exe –? > [filename]

Checking controller availability

Syntax

perccli show

Description

Displays information about the adapter and the operating system.

Result

Status Code = 0
Status = Success
Description = none

Number of Controllers = 1
Host name = WIN-RFV0S1VAILB
Operating System = Windows Server 2012

System Overview :

=================================
Ctl Model  Ports  PDs  DGs  DNOpt  VDs  VNOpt  BBU  sPR  DS  EHS  ASOs
=================================
0  Adapter  8     9    2     0     2     0    Fld   On  3    N    0

Working with the PERC Command Line Interface tool
** Viewing controllers **

** Syntax **

perccli show ctrlcount

** Description **
Displays the number of controllers detected in the server.

** Result **

Status Code = 0  
Status = Success  
Description = None  
Controller Count = 1

** Viewing free space information **

** Syntax **

perccli /c0 show freespace

** Description **
Displays the free space details of the controller.

** Result **

Status Code = 0  
Status = Success  
Description = None  
FREE SPACE DETAILS :  
---------------------  
Total Slot Count = 0  
ID-Index|DG-Drive Group|AftrVD-Identify Freespace After VD

** Viewing disk1 information **

** Syntax **

perccli /c0/d1 show

** Description **
Displays information about disk1.

** Result **

Controller = 0  
Status = Success  
Description = Show Diskgroup Succeeded
TOPOLOGY :
--------

<table>
<thead>
<tr>
<th>DG Arr</th>
<th>Row</th>
<th>EID:Slot</th>
<th>DID</th>
<th>Type</th>
<th>State</th>
<th>BT</th>
<th>Size</th>
<th>PDC</th>
<th>PI</th>
<th>SED</th>
<th>DS3</th>
<th>FSpace</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>32:2</td>
<td>DRIVE Onln</td>
<td>N</td>
<td>558.375 GB</td>
<td>dflt N</td>
<td>Y</td>
<td>Y</td>
<td>dflt</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

Viewing controller, virtual disk, and drivers information

Syntax

percli /c0 show

Description

Displays information about the adapter, virtual disks, and drivers.

Result

| Status Code | Status   | Description | Product Name | Serial Number | SAS Address | Mfg. Date | System Time | Controller Time | FW Package Build | BIOS Version | FW Version | Driver Name | Driver Version | Controller Bus Type | PCI Slot | PCI Bus Number | PCI Device Number | PCI Function Number | Drive Group |
|-------------|----------|-------------|--------------|---------------|-------------|------------|-------------|---------------|-----------------|----------------|-------------|-------------|--------------|-------------------|-------------------|-----------|----------------|-----------------|---------------------|------------|
| 0           | Success  | none        | PERC H730P Adapter | 38E005K       | 5b8ca3a0f78d9000 | 08/28/13   | 11/30/2013 05:12:51 | 11/30/2013 05:13:29 | 25.2.0.0.0014 | 6.12.00_4.12.05.00_0x06020101 | 6.220.00-2918 | PercSas3.sys | 6.600.52.00 | N/A                | N/A               | 4                    | 0                    | 0                    | 0                    | 2            |

TOPOLOGY :
--------

<table>
<thead>
<tr>
<th>DG Arr</th>
<th>Row</th>
<th>EID:Slot</th>
<th>DID</th>
<th>Type</th>
<th>State</th>
<th>BT</th>
<th>Size</th>
<th>PDC</th>
<th>PI</th>
<th>SED</th>
<th>DS3</th>
<th>FSpace</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>32:0</td>
<td>DRIVE Onln</td>
<td>N</td>
<td>558.375 GB</td>
<td>dflt N</td>
<td>V</td>
<td>dflt</td>
<td>Y</td>
<td>dflt</td>
<td></td>
</tr>
</tbody>
</table>

Virtual Drives = 2

VD LIST :
-------

<table>
<thead>
<tr>
<th>DG/VD</th>
<th>Type</th>
<th>State</th>
<th>Access</th>
<th>Consist</th>
<th>Cache</th>
<th>sCC</th>
<th>Size</th>
<th>Name</th>
</tr>
</thead>
</table>
Checking for preserved cache

Syntax

perccli /c0 show preservedcache

Description
Displays available preserved cache.

Result

Controller = 0
Status = Success
Description = None

Deleting preserved cache

Syntax

perccli /c0/v1 delete preservedcache

Description
Deletes the available preserved cache.
Result
Controller = 0
Status = Success
Description = Virtual Drive preserved Cache Data Cleared

Viewing expansion information
Syntax
perccli /c0/v0 show expansion

Description
Displays virtual drive’s expansion information with and without array expansion.

Result
Controller = 0
Status = Success
Description = None

EXPANSION INFORMATION :
VD Size OCE NoArrExp WithArrExp Status
------------------------------------------------
0 1.635 TB N - - -
------------------------------------------------
OCE - Online Capacity Expansion | WithArrExp - With Array Expansion
NoArrExp - Without Array Expansion

Working with the PERC Command Line Interface tool
51
Importing the foreign configuration

Syntax

perccli /c0/fall import

Description

Imports the foreign configurations of the selected controller.

Result

Controller = 0
Status = Success
Description = Successfully imported foreign configuration

Viewing BBU information

Syntax

perccli /c0/bbu show all

Description

Displays information related to the Battery Backup Unit (BBU) of a controller.

Result

Controller = 0
Status = Success
Description = None

BBU_Info :
----------
----------------------
Property      Value
----------------------
Type           BBU
Voltage        3 mV
Current        0 mA
Temperature    32 C
Battery State  Optimal
----------------------

BBU_Firmware_Status :
---------------------
-------------------------------------------------
Property                                   Value
-------------------------------------------------
Charging Status                            None
Voltage                                    OK
Temperature                                OK
Learn Cycle Requested                      No
Learn Cycle Active                         No
Learn Cycle Status                         OK
Learn Cycle Timeout                        No
I2C Errors Detected                        No
Battery Pack Missing                       No
Replacement required                       No
Remaining Capacity Low                     No
Periodic Learn Required                    No
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully Discharged</td>
<td>Yes</td>
</tr>
<tr>
<td>Fully Charged</td>
<td>Yes</td>
</tr>
<tr>
<td>Discharging</td>
<td>No</td>
</tr>
<tr>
<td>Initialized</td>
<td>No</td>
</tr>
<tr>
<td>Remaining Time Alarm</td>
<td>No</td>
</tr>
<tr>
<td>Remaining Capacity Alarm</td>
<td>Yes</td>
</tr>
<tr>
<td>Terminate Discharge Alarm</td>
<td>No</td>
</tr>
<tr>
<td>Over Temperature</td>
<td>No</td>
</tr>
<tr>
<td>Charging Terminated</td>
<td>No</td>
</tr>
<tr>
<td>Over Charged</td>
<td>No</td>
</tr>
<tr>
<td>Relative State of Charge</td>
<td>100%</td>
</tr>
<tr>
<td>Charger System State</td>
<td>Complete</td>
</tr>
<tr>
<td>Remaining Capacity</td>
<td>407</td>
</tr>
<tr>
<td>Full Charge Capacity</td>
<td>407</td>
</tr>
<tr>
<td>Is SOH Good</td>
<td>Yes</td>
</tr>
<tr>
<td>Battery backup charge time</td>
<td>0 hour(s)</td>
</tr>
</tbody>
</table>

**BBU_Capacity_Info**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative State of Charge</td>
<td>100%</td>
</tr>
<tr>
<td>Absolute State of charge</td>
<td>0%</td>
</tr>
<tr>
<td>Remaining Capacity</td>
<td>407 mAh</td>
</tr>
<tr>
<td>Full Charge Capacity</td>
<td>407 mAh</td>
</tr>
<tr>
<td>Run time to empty</td>
<td>Battery is not being charged</td>
</tr>
<tr>
<td>Average time to empty</td>
<td>33 min</td>
</tr>
<tr>
<td>Average Time to full</td>
<td>Battery is not being charged</td>
</tr>
<tr>
<td>Cycle Count</td>
<td>3</td>
</tr>
<tr>
<td>Max Error</td>
<td>0%</td>
</tr>
<tr>
<td>Remaining Capacity Alarm</td>
<td>0 mAh</td>
</tr>
<tr>
<td>Remaining Time Alarm</td>
<td>0 minutes(s)</td>
</tr>
</tbody>
</table>

**BBU_Design_Info**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date of Manufacture</td>
<td>18/07/2011</td>
</tr>
<tr>
<td>Design Capacity</td>
<td>90 mAh</td>
</tr>
<tr>
<td>Design Voltage</td>
<td>0 mV</td>
</tr>
<tr>
<td>Specification Info</td>
<td>0</td>
</tr>
<tr>
<td>Serial Number</td>
<td>0</td>
</tr>
<tr>
<td>Pack Stat Configuration</td>
<td>0</td>
</tr>
<tr>
<td>Manufacturer's Name</td>
<td></td>
</tr>
<tr>
<td>Device Name</td>
<td></td>
</tr>
<tr>
<td>Device Chemistry</td>
<td></td>
</tr>
<tr>
<td>Battery FRU</td>
<td>N/A</td>
</tr>
<tr>
<td>Transparent Learn</td>
<td>1</td>
</tr>
<tr>
<td>App Data</td>
<td>0</td>
</tr>
<tr>
<td>Module Version</td>
<td>0.3</td>
</tr>
</tbody>
</table>

**BBU_Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Viewing physical drive details for the specified slot in the controller

Syntax

```
perccli /c0/e32/s4 show all
```

Description

Displays information about the physical drive, including device attribute, settings, and port information for a particular slot in the controller.

Result

```
Controller = 0
Status = Success
Description = Show Drive Information Succeeded.

Drive /c0/e32/s4:

<table>
<thead>
<tr>
<th>EID: Slt</th>
<th>DID</th>
<th>State</th>
<th>DG</th>
<th>Size</th>
<th>Intf</th>
<th>Med</th>
<th>SED</th>
<th>PI</th>
<th>SeSz</th>
<th>Model</th>
<th>Sp</th>
</tr>
</thead>
<tbody>
<tr>
<td>32:4</td>
<td>4</td>
<td>Onln</td>
<td>0</td>
<td>558.375 GB</td>
<td>SAS</td>
<td>HDD</td>
<td>Y</td>
<td>Y</td>
<td>4 KB</td>
<td>ST600MP0084</td>
<td>U</td>
</tr>
</tbody>
</table>

Drive /c0/e32/s4 - Detailed Information:

Drive /c0/e32/s4 State:

- Shield Counter = 0
- Media Error Count = 0
- Other Error Count = 0
- Drive Temperature = 43c <109.40F>
- Predictive Failure Count = 0
- S.M.A.R.T alert flagged by drive = No

Drive /c0/e32/s4 Device attribute:

- SN = S2G01H5T
- WWN = 5000C5006B1A4FB8
- Firmware Revision = VB44
- Raw size = 558.911 GB [0x8bba5f6 Sectors]
- Coerced size = 558.375 GB [0x8b98000 Sectors]
- Non Coerced size = 558.411 GB [0x8b9a5f6 Sectors]
- Device Speed = 6.0Gb/s
- Link Speed = 6.0Gb/s
- Logical Sector Size = 4 KB
- Physical Sector Size = 4 KB

Drive /c0/e32/s4 Policies/Settings:

- Drive position = DriveGroup:0, Span:0, Row:3
- Enclosure Position = 0
```
Connected Port Number = 0
Sequence Number = 2
Commissioned Spare = No
Emergency Spare = No
Last Predictive Failure Event Sequence Number = 0
Successful diagnostics completion on = N/A
SED Capable = Yes
SED Enabled = Yes
Secured = Yes
Locked = No
Needs EKM Attention = No
PI Eligible = Yes
Drive is formatted for PI = Yes
PI type = 2
Number of bytes of user data in LBA = 4 KB
Certified = Yes
Wide Port Capable = No

Port Information:
-------------------------------
Port  Status  Linkspeed  SAS address
-------------------------------
0   Active  6.0Gb/s    0x5000c5006b1a4fba
1   Active  6.0Gb/s    0x0
-------------------------------

Inquiry Data =
00 00 06 12 8b 01 30 02 53 45 41 47 41 54 45 20
53 54 36 30 30 4d 50 30 30 84 20 20 20 20 20
56 42 34 34 53 32 47 30 31 48 35 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 43 6f 70 79 72 69 67 68 74 20 28 63 29 20 32
30 31 33 20 53 65 61 67 61 74 65 20 41 6c 6c 20

Viewing the boot drive for the controller

Syntax

perccli /c0 show bootdrive

Description

Displays the boot drive for the controller. The boot drive can be a physical drive or a virtual drive.

Result

Controller = 0
Status = Success
Description = None

Controller Properties:
-------------------------------
Ctrl_PROP  Value
BootDrive  VD:13
-------------------------------
Setting virtual drive as boot drive

Syntax

perccli /c0/v13 set bootdrive = on

Description

Sets the specified virtual drive as the boot drive. During the next reboot, the BIOS looks for a boot sector in the specified virtual drive.

**NOTE:** Set bootdrive is applicable only in legacy BIOS mode and is not supported in UEFI mode.

Result

<table>
<thead>
<tr>
<th>Controller</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Success</td>
<td>None</td>
</tr>
</tbody>
</table>

Detailed Status :

------------------------
VD Property   Value Status   ErrCd   ErrMsg
------------------------
13   Boot Drive On Success 0   -
------------------------

Locating a drive

Syntax

perccli /c0/e32/s0 start locate

Description

Locates a drive and activates the physical disk activity LED.

Result

<table>
<thead>
<tr>
<th>Controller</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Success</td>
<td>Start Drive Locate Succeeded</td>
</tr>
</tbody>
</table>

Stopping a locate operation

Syntax

perccli /c0/e32/s0 stop locate

Description

Stops a drive locate operation and deactivates the physical disk activity LED.
The PERC Command Line Tool supports the following commands for snapdump state, save count, collection, and deletion.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>perccli /cx show snapdump</td>
<td>This command shows the snapdump information on the controller. Input example: perccli /c0 show snapdump</td>
</tr>
<tr>
<td>perccli /cx set snapdump set=on</td>
<td>off</td>
</tr>
<tr>
<td>perccli /cx set snapdump [ savecount=&lt;value&gt;</td>
<td>delayocr=&lt;value&gt; ]</td>
</tr>
</tbody>
</table>

### Table 20. Snapdump commands

<table>
<thead>
<tr>
<th>Option</th>
<th>Value Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>savecount</td>
<td>0-255</td>
<td>Number of times snapdump persists on the controller across reboots.</td>
</tr>
<tr>
<td>delayocr</td>
<td>15-60</td>
<td>Seconds that the controller has to collect the snapdump logs before OCR.</td>
</tr>
</tbody>
</table>

**NOTE:** The default values for savecount and delayocr are 4 and 15 respectively.

**NOTE:** Snapdump is not enabled on the H345 controller.

**NOTE:** Once snapdump is disabled on the controller it will persist through the updates and reboot until it is enabled.
The `perccli /cx get snapdump` command is used to get controller snapdump data and save it to a file. The command supports the following options:

- `perccli /c0 get snapdump` [norttdump]
- `perccli /c0 get snapdump id=all` [norttdump]
- `perccli /c0 get snapdump id=<value> file=fileName` [norttdump]

Example usage:

```
perccli /c0 get snapdump id=0 file=Snapdump.zip
```

The `perccli /cx delete snapdump` command deletes controller snapdump information. The `force` option allows deleting all available snapdumps.

Example usage:

```
perccli /c0 delete snapdump force
```
Contacting Dell EMC

Dell EMC provides several online and telephone based support and service options. If you do not have an active internet connection, you can find contact information about your purchase invoice, packing slip, bill, or Dell EMC product catalog. Availability varies by country and product, and some services may not be available in your area. To contact Dell EMC for sales, technical assistance, or customer service issues:

2. Select your country from the drop-down menu on the lower right corner of the page.
3. For customized support:
   a) Enter your system Service Tag in the Enter your Service Tag field.
   b) Click Submit.
      The support page that lists the various support categories is displayed.
4. For general support:
   a) Select your product category.
   b) Select your product segment.
   c) Select your product.
      The support page that lists the various support categories is displayed.
5. For contact details of Dell EMC Global Technical Support:
   a) Click Global Technical Support.
   b) The Contact Technical Support page is displayed with details to call, chat, or e-mail the Dell EMC Global Technical Support team.

Locating your system Service Tag

Your system is identified by a unique Express Service Code and Service Tag number. The Express Service Code and Service Tag are found on the front of a physical DR Series system by pulling out the information tag. The service tag can also be found on the Support page in the GUI. This information is used to route support calls to the appropriate personnel for resolution.