



UCS Secure Data Deletion For Commission Regulation (EU) 2019/424 Users Guide

First Published: 2020-02-27

Last Modified: 2026-03-04

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



Preface

This preface contains the following topics.

- [Bias-Free Documentation](#), on page iii
- [Full Cisco Trademarks with Hardware License](#), on page iii
- [Communications, Services, and Additional Information](#), on page v

Bias-Free Documentation



Note The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Full Cisco Trademarks with Hardware License

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated

in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If the equipment causes interference to radio or television reception, which can be determined by turning the equipment off and on, users are encouraged to try to correct the interference by using one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Modifications to this product not authorized by Cisco could void the FCC approval and negate your authority to operate the product.

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

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

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

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business results you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco DevNet](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).

Cisco Bug Search Tool

[Cisco Bug Search Tool \(BST\)](#) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



CHAPTER 1

Overview

This chapter contains the following topic:

- [Overview, on page 1](#)

Overview

The Commission Regulation (EU) 2019/424 requires that data be securely disposed of. This document explains tools you can use and provides instructions about running specific tools to erase data on drives and in memory before sending your Cisco UCS servers to recyclers for proper disposal.

Secure data disposal is accomplished by using commonly available tools that erase the data from the various drives, memory, and storage in the Cisco UCS servers.

Different UCS servers have different types of devices that need to be erased. It is your responsibility to know the types of devices and data on them. It is also your responsibility to erase the data on all devices by using the appropriate tools.

Cisco is not liable for ensuring data is erased from UCS servers, erasing incorrect data, or data that is left on the UCS servers.



CHAPTER 2

Supported Servers

This chapter contains the following topic:

- [Supported Servers, on page 3](#)

Supported Servers

Secure data deletion for compliance with Commission Regulation (EU) 2019/424 is supported for the following Cisco UCS products:

- Cisco UCS X410c M8
- Cisco UCS XE9305
- Cisco UCS C845A M8
- Cisco UCS X210c M8
- Cisco UCS C240 M8
- Cisco UCS C220 M8
- Cisco UCS C245 M8
- Cisco UCS C225 M8
- Cisco UCS C885A M8
- Cisco UCS C220 M7
- Cisco UCS C240 M7
- Cisco UCS X215c M8
- Cisco UCS X210c M7
- Cisco UCS X410c M7
- Cisco UCS B200 M6
- Cisco UCS C220 M6
- Cisco UCS C225 M6
- Cisco UCS C240 M6

- Cisco UCS C245 M6
- Cisco UCS X10c
- Cisco UCS X210c M6
- Cisco UCS X9508
- Cisco UCS B200 M5
- Cisco UCS B480 M5
- Cisco UCS C125 M5
- Cisco UCS C220 M5
- Cisco UCS C240 M5
- Cisco UCS C480 M5
- Cisco UCS S3260 M5



CHAPTER 3

Supported Secure Data Deletion Tools

This chapter contains the following topics:

- [Secure Data Deletion Tools, on page 5](#)

Secure Data Deletion Tools

UCS servers have different data storage devices. You can use the following tools to delete data from your Cisco UCS servers. All the following tools are certified in Red Hat OS. Make sure that you run these tools in Linux OS.



Note Depending on your UCS servers, you might need to run multiple tools.

Make sure you are familiar with the tools. If you are not familiar with the tools, are not sure about which devices are in your UCS server, or are not sure which tools to run, do not proceed. Find someone who knows the devices in the servers and has the technical knowledge to run the tools.

sg_sanitize

The **sg_sanitize** tool is part of the `sg3_utils` package of utilities that sends SCSI commands to a device through a SCSI pass-through interface provided by the host operating system. The tool is available for general download for hard disk drives (HDDs) and solid-state drives (SSDs), including Micron SSDs, but not Micron M.2 SSDs.

To download this tool, go to: http://sg.danny.cz/sg/sg3_utils.html

To use this tool, see either of the following:

- [Running sg_sanitize on SED HDDs, on page 12](#)
- [Running sg_sanitize on SAS SSDs, on page 12](#)

sg_format

The **sg_format** tool is part of the `sg3_utils` package of utilities. You can use this tool to accomplish the following on modern SCSI disks:

- format the disk

- potentially change their block size (if permitted)
- change the block count

When you use **sg_format** with the **--format** (or **-F**) option, it will attempt to format the device you specify. The format option destroys all the data held on the device.

To use this tool, see [Running sg_format on SAS HDDs, on page 9](#).

hdparm

The **hdparm** tool supports Get and Set operations on ATA and SATA drive parameters in Linux. You can also use this tool to issue **sanitize** commands to SATA SSDs, including M.2 and U.2 form factors.

To download this tool, go to: <https://sourceforge.net/projects/hdparm/>

To use this tool, see [Running hdparm on SATA SSDs, Including M.2 and U.2 Form Factors, on page 15](#).

NVMe-cli

The **nvme-cli** is a Linux-based toolset that manages non-volatile memory devices. This tool is open source and available in the public domain from Github. You will need a Github account to get the tool.

To download this tool, go to: <https://github.com/linux-nvme/nvme-cli>

To use this tool, see [Running nvme-cli, on page 17](#).

dd Utility

The **dd** command is a Linux-based command-line utility that you can use to wipe a disk by writing all zeroes to it. This utility is useful for SD cards.

The **dd** command is integrated into the UNIX-kernel space of each server, so you don't need to download it. You must have root access to use the utility.

To use this tool, see either of the following:

- [Running dd on SATA HDDs, on page 10](#)
- [Running dd on SD Cards, on page 19](#)

ndctl

Intel provides the **ndctl** tool for secure data deletion from Intel persistent memory devices, such as Intel's Optane Persistent Memory Modules, also known as Apache Pass. The tool is available for the Linux OS at Github. You need a Github account to access **ndctl**.

To download the tool, go to <https://github.com/pmem/ndctl>.

To use this tool, see [Running ndctl on Intel Optane Persistent Memory, on page 19](#).



CHAPTER 4

Before Deleting Data

This chapter contains the following topic:

- [Before Deleting Data, on page 7](#)

Before Deleting Data

Before deleting data from the UCS servers, you must know the different types of storage devices in your UCS servers and use the appropriate tool for secure data deletion. For information about the supported tools, see [Secure Data Deletion Tools, on page 5](#).



Caution

Cisco is not liable for any damage or data loss, or any loss or impairment of functionality on the server or network that occurs from you not deleting data or incorrectly deleting data.

If you are not familiar with the types of data devices and how to delete data, do not proceed. Find someone who is familiar with the data devices and tools who can safely delete the data.



CHAPTER 5

Deleting Data

This chapter contains the following topics:

- [Running sg_format on SAS HDDs, on page 9](#)
- [Running dd on SATA HDDs, on page 10](#)
- [Running sg_sanitize on SED HDDs, on page 12](#)
- [Running sg_sanitize on SAS SSDs, on page 12](#)
- [Running sg_sanitize on Tri-Mode NVME SSDs, on page 13](#)
- [Running hdparm on SATA SSDs, Including M.2 and U.2 Form Factors, on page 15](#)
- [Running nvme-cli, on page 17](#)
- [Running dd on SD Cards, on page 19](#)
- [Running ndctl on Intel Optane Persistent Memory, on page 19](#)
- [Data Sanitization for Microchip Controllers, on page 20](#)

Running sg_format on SAS HDDs

Before you begin

- If you have not already read [Before Deleting Data, on page 7](#), read it now.
- If your UCS server does not have sg3_utils version 1.44 installed in Linux, install it now.

Procedure

Step 1 Select the appropriate option, which depends on whether the device is in a RAID or in JBOD mode.

If the device is in a RAID:

- a) Unmount the drive.
- b) Remove the virtual disk.
- c) Convert it to JBOD mode.

If the device is in JBOD mode, unmount the drive.

Step 2 Erase the data by running `sg_format --format /dev/sd*` where * is the device number.

Caution

Do not power off the system or interrupt this command before the operation is completed. Doing so may cause the drive to be in an unknown state and can possibly lead to a dead drive (bricking the drive).

If the command completes without an error or busy feedback, the format operation is successful.

Running dd on SATA HDDs

Before you begin

If you have not already read [Before Deleting Data, on page 7](#), read it now.

Procedure

Step 1 Select the appropriate option, which depends on whether the device is in a RAID, in JBOD mode, or is in an embedded RAID controller.

If the device is in a RAID:

- a) Unmount the drive.
- b) Remove the virtual disk.
- c) Convert it to JBOD mode.
- d) Go to Step 6.

If the device is in JBOD mode, unmount the drive, then go to Step 6.

If the device is in an embedded software RAID controller, perform Step 2 through Step 5.

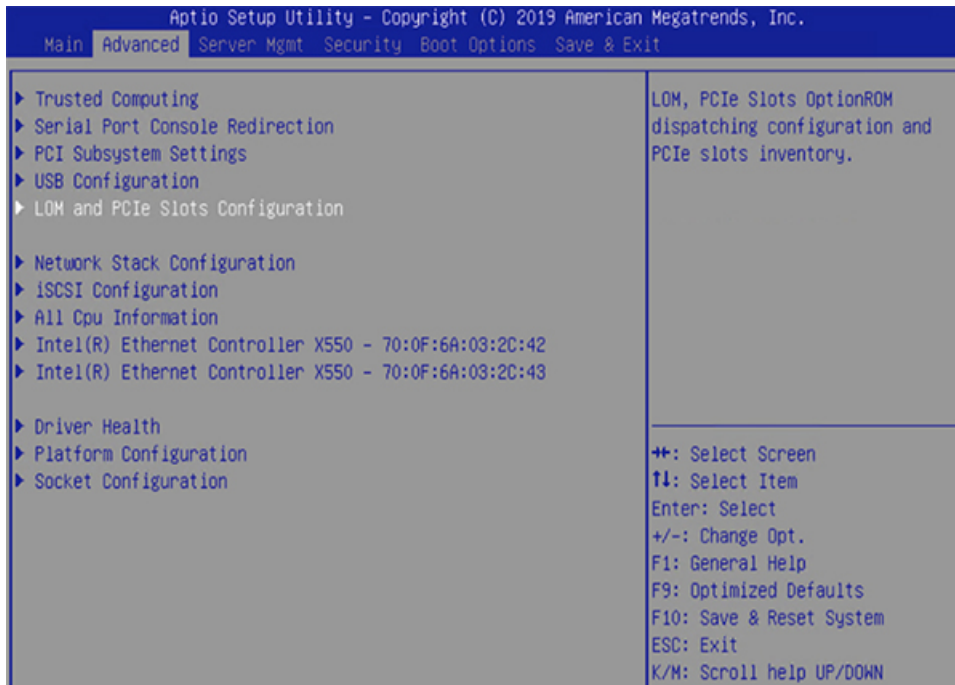
Caution

Converting a device in an embedded software RAID controller to AHCI mode is applied *globally to the entire software RAID controller*. You cannot target individual drives. Be aware that data may be lost on drives you don't specify, in other words, the rest of the virtual drives under the same controller.

Step 2 To set the device in the embedded software RAID controller to AHCI mode:

- a) Boot into BIOS mode.
- b) Select Advanced>LOM and PCIe Slots Configuration

Figure 1: BIOS Menu, Advanced Tab



c) Select the pSATA SATA OpROM option to change the embedded controller (pSATA) to AHCI mode.

This step displays the pSATA SATA OpROM popup dialog.

d) Select **AHCI**.

Note

Remember that the device you target *plus all other virtual devices under the same controller* are set AHCI mode.

Step 3 Save the change to AHCI.

Step 4 Exit the BIOS menu.

Step 5 Choose the correct option based on whether the AHCI driver is blacklisted:

- If the native Linux AHCI driver was added into the blacklist (disabled) in your Linux OS, please reinstall your Linux OS with AHCI mode. Otherwise, the OS can't discover the devices that have been converted to AHCI mode. Go to Step 6.
- If the AHCI driver was not disabled, then go to Step 6.

Step 6 Erase the data by running `dd if=/dev/zero of=/dev/sd* bs=1M` where * is the device number.

If the command completes without an error or busy feedback, the `dd` operation is successful.

Running `sg_sanitize` on SED HDDs

Before you begin

- If you have not already read "[Before Erasing Data](#)," read it now.
- If your UCS server does not have `sg3_utils` version 1.44 installed in Linux, install it now.

Procedure

Step 1 Select the appropriate option, which depends on whether the device is in a RAID or in JBOD mode:

If the device is in a RAID:

- a) Unmount the drive.
- b) Remove the virtual disk.
- c) Convert it to JBOD mode.

If the device is in JBOD mode, unmount the device.

Step 2 Run `psid revert` on each SED HDD even if drive security is disabled. If you skip this step, the `sg_sanitize` command fails.

Step 3 Erase the data by running `sg_sanitize --crypto /dev/sd*` where * is the device number.

If the command completes without an error or busy feedback, the sanitize operation is successful.

Running `sg_sanitize` on SAS SSDs

- If you have not already read "[Before Erasing Data](#)," read it now.
- If your UCS server does not have `sg3_utils` version 1.44 installed in Linux, install it now.

Use this task to run `sg_sanitize` to securely delete data from UCS server SAS SSDs. You can run the command with either of the following options:

- `block erase`: Erases all the drive's blocks, including the over-provision space. This option completes fast.
- `crypto erase`: Applies to SED SAS SSDs. This option deletes the drive's internal key so that data becomes unusable. This option completes very fast.

Procedure

Step 1 Select the appropriate option, which depends on whether the device is in a RAID or in JBOD mode:

If the device is in a RAID:

- a) Unmount the drive.
- b) Remove the virtual disk.
- c) Convert it to JBOD mode.

If the device is in JBOD mode, unmount the device.

Step 2 Choose the correct option:

- a) If the device is a SED SAS SSD, run **psid revert** on each SED SAS SSD even if drive security is disabled. If you skip this step, the `sg_sanitize` command fails. Proceed to the next step.
- b) If the device is not a SED SAS SSD, proceed to the next step.

Step 3 Run `sg_sanitize` based on the type of drive:

- For SED SAS SSDs: Run the command in crypto erase mode. For example, `sg_sanitize --crypto /dev/sd*` where * is the device number
- For non-SED SAS SSDs: Run the command in block erase mode. For example, `sg_sanitize --block /dev/sd*`

Example:

```
[root@localhost ~]# sg_sanitize --block /dev/sdc
WDC          WUSTR1548ASS200  A925  peripheral_type: disk [0x0]
  << supports protection information>>
  Unit serial number:          WTX006GA
  LU name: 5000cca0a870bf28

A SANITIZE will commence in 15 seconds
  ALL data on /dev/sdc will be DESTROYED
  Press control-C to abort

A SANITIZE will commence in 10 seconds
  ALL data on /dev/sdc will be DESTROYED
  Press control-C to abort

A SANITIZE will commence in 5 seconds
  ALL data on /dev/sdc will be DESTROYED
  Press control-C to abort
[root@localhost ~]#
```

If the command completes without an error or busy feedback, the sanitize operation is successful.

Running `sg_sanitize` on Tri-Mode NVME SSDs

- If you have not already read "[Before Erasing Data](#)," read it now.
- If your UCS server does not have `sg3_utils` version 1.44 installed in Linux, install it now.

Use this task to run `sg_sanitize` to securely delete data from UCS server NVMe SSDs that are connected to the Cisco UCSC 24G Tri-Mode RAID Controller with 4GB FBWC (UCSC-RAID-HP). Tri-Mode RAID is supported on the following products:

- Cisco UCS C220 M7 and C240 M7 rack server with either:

- Cisco UCSC 24G Tri-Mode RAID Controller with 4GB Flash-Backed Write Cache (FBWC) (UCSC-RAID-HP-D)
- Cisco 12-Gbps modular RAID controller (PCIe 4.0) with 4-GB Flash-Backed Write Cache (FBWC) (UCSC-RAID-HP).
- Cisco UCS X210c M6 compute Node and the Cisco UCS X210c M7 and X410c M7 compute nodes with the UCS X10c Compute RAID Controller with LSI 3900 (UCSX-X10C-RAIDF)

You can run the `sg_sanitize` command on NVMe drives that are running in Tri-Mode.

Procedure

Step 1 When the device is in RAID mode:

- Unmount the drive.
- Remove the virtual disk.
- Convert it to JBOD mode.

Step 2 Run `sg_sanitize` for NVME drives connected to the Tri-Mode controller:

Run the command in block erase mode. For example, `sg_sanitize --block /dev/sd*`

Example:

```
[root@localhost ~]# sg_sanitize --block /dev/sd*
NVME      Micron_7450_MTFD  S002  peripheral_type: disk [0x0]
Unit serial number: 22213ACA1755
LU name: 00a075013aca1755

A SANITIZE will commence in 15 seconds
  ALL data on /dev/sd* will be DESTROYED
  Press control-C to abort

A SANITIZE will commence in 10 seconds
  ALL data on /dev/sd* will be DESTROYED
  Press control-C to abort

A SANITIZE will commence in 5 seconds
  ALL data on /dev/sd* will be DESTROYED
  Press control-C to abort
```

If the command completes without an error or busy feedback, the sanitize operation is successful.

Running hdparm on SATA SSDs, Including M.2 and U.2 Form Factors

Before you begin

- If you have not already read [Before Deleting Data, on page 7](#), read it now.
- If your UCS server does not have **hdparm** version 9.54 installed in Linux, install it now.

Use this task to run **hdparm** to securely delete data from UCS server SATA SSDs, including M.2 and U.2 SATA SSDs. You can run the command with either of the following options:

- `block erase`: Erases all the drive's blocks, including the over-provision space. This option completes fast.
- `crypto erase`: Applies to SED SATA SSD. This option deletes the drive's internal key so that data becomes unusable. This option completes very fast.

Procedure

Step 1 Select the appropriate option, which depends on whether the device is in a RAID, in JBOD mode, or is in an embedded RAID controller.

If the device is in a RAID:

- a) Unmount the drive.
- b) Remove the virtual disk.
- c) Convert it to JBOD mode.
- d) Go to Step 6.

If the device is in JBOD mode, unmount the drive, then go to Step 6.

If the device is in an embedded software RAID controller, perform Step 2 through Step 5.

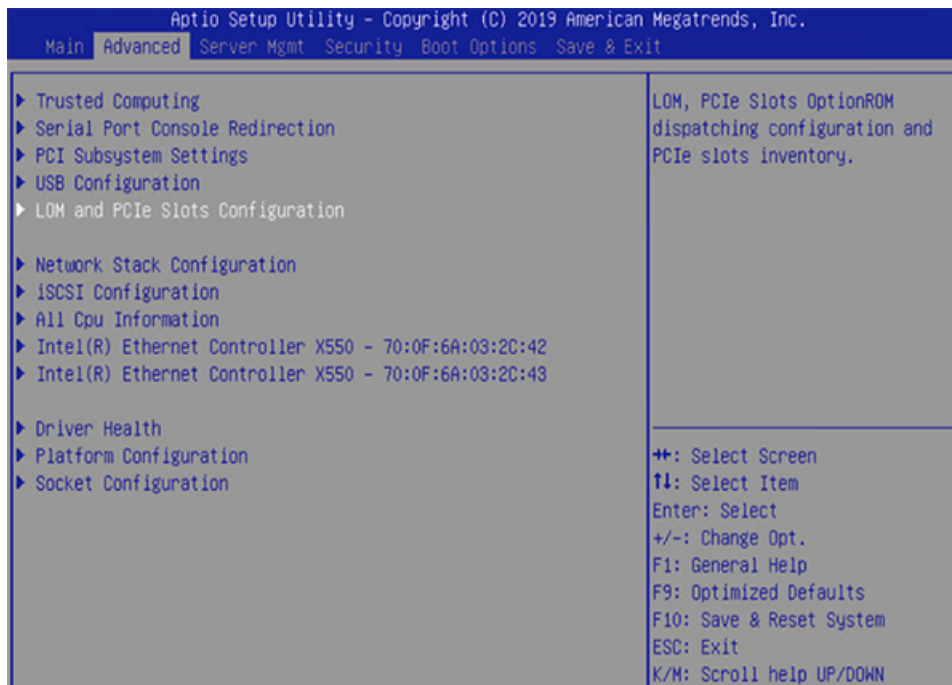
Caution

Converting a device in an embedded software RAID controller to AHCI mode is applied *globally to the entire software RAID controller*. You cannot target individual drives. Be aware that data may be lost on drives you don't specify, in other words, the rest of the virtual drives under the same controller.

Step 2 To set the device in the embedded software RAID controller to AHCI mode:

- a) Boot into BIOS mode.
- b) Select Advanced>LOM and PCIe Slots Configuration

Figure 2: BIOS Menu, Advanced Tab



c) Select the pSATA SATA OpROM option to change the embedded controller (pSATA) to AHCI mode.

This step displays the pSATA SATA OpROM popup dialog.

d) Select **AHCI**.

Note

Remember that the device you target *plus all other virtual devices under the same controller* are set AHCI mode.

Step 3 Save the change to AHCI.

Step 4 Exit the BIOS menu.

Step 5 Choose the correct option based on whether the AHCI driver is blacklisted:

- If the native Linux AHCI driver was added into the blacklist (disabled) in your Linux OS, please re-install your Linux OS with AHCI mode. Otherwise, the OS can't discover the devices that have been converted to AHCI mode. Go to Step 6.
- If the AHCI driver was not disabled, then go to Step 6.

Step 6 Choose the correct option:

- If the device is a SED SATA SSD, run **psid revert** on each SED SATA SSD even if drive security is disabled. If you skip this step, the **hdparm** command fails. Proceed to the next step.
- If the device is not a SED SATA SSD, proceed to the next step.

Step 7 Run **hdparm** based on the type of drive:

- For SED SATA SSDs: Run the command in crypto erase mode. For example, **hdparm --yes-i-know-what-i-am-doing --sanitize-crypto-scramble /dev/sd*** where * is the device number.

Use this command to check the sanitize process, **hdparm --sanitize-status /dev/sd***

- For non-SED SATA SSDs and M.2 SATA SSDs: Run the command in block erase mode. For example, **hdparm --yes-i-know-what-i-am-doing --sanitize-block-erase /dev/sd*** where * is the device number.

Use this command to check the sanitize process, **hdparm --sanitize-status /dev/sd***

Example:

```
[root@localhost ~]# hdparm --yes-i-know-what-i-am-doing --sanitize-block-erase /dev/sdb

/dev/sdb:
Issuing SANITIZE_BLOCK_ERASE command
Operation started in background
You may use `--sanitize-status` to check progress
[root@localhost ~]# hdparm --sanitize-status /dev/sdb

/dev/sdb:
Issuing SANITIZE_STATUS command
Sanitize status:
  State:    SD2 Sanitize operation In Process
  Progress: 0x0 (0%)
[root@localhost ~]# hdparm --sanitize-status /dev/sdb

/dev/sdb:
Issuing SANITIZE_STATUS command
Sanitize status:
  State:    SD2 Sanitize operation In Process
  Progress: 0x805b (50%)
[root@localhost ~]# hdparm --sanitize-status /dev/sdb

/dev/sdb:
Issuing SANITIZE_STATUS command
Sanitize status:
  State:    SD0 Sanitize Idle
  Last Sanitize Operation Completed Without Error
[root@localhost ~]#
```

Running nvme-cli

Use the **nvme format** command to erase data from non-volatile memory devices in your UCS servers, including NVMe drives connected directly to a CPU. This command is part of the open source **nvme-cli** package. The **nvme format** command has a secure erase option (**--ses**) which you can run with the following qualifiers:

- `no secure erase`, value 0 (zero). This option does not erase data.
- `user data erase`, value 1. This option erases all user data.
- `cryptographic erase`, value 2. This option erases user data cryptographically by deleting the encryption key.

Cisco recommends using user data erase (**--ses=1**).

Before you begin

If you have not already read [Before Deleting Data, on page 7](#), read it now.

Procedure

Step 1 Unmount the non-volatile drive.

Note

You can issue **nvme list** to find the namespace value and node name of the NVE device.

Example:

```
[root@localhost ~]# nvme list
Node                               SN                               Model                               Namespace  Usage
Format                             FW Rev
-----
/dev/nvme0n1                       SDM1234567A                     UCSC-NVME-H76801                   1          7.6 TB
/ 7.68 TB
512 B +
/dev/nvme1n1                       PHLF736301522POHGN             INTEL SSDPE2KX020T7K              1          2.00 TB
/ 2.00 TB
512 B +
/dev/nvme2n1                       PHLF7363011Y2POHGN             INTEL SSDPE2KX020T7K              1          2.00 TB
/ 2.00 TB
512 B +
/dev/nvme3n1                       PHIF7363015X2POHGN             INTEL SSDPE2KX020T7K              1          2.00 TB
/ 2.00 TB
512 B +
/dev/nvme4n1                       PHIF736301442POHGN             INTEL SSDPE2KX020T7K              1          2.00 TB
/ 2.00 TB
512 B +
/dev/nvme5n1                       SDM0000021EC                   UCSC-NVMtHW-HBGO                   1          800.17 GB
/ 800.17 TB
512 B +
[root@localhost ~]#
```

Step 2 Run the **nvme format** command, with the user data erase option:

```
nvme format --ses=value --namespace-id=namespace-value /dev/device-id --timeout=timeout-value
```

where:

namespace-value is the number in the Namespace column of **nvme list**

device-id is the unique name of the NVME device in the Node column of **nvme list**, for example `nvme0n1`

--timeout-value= is 1800000. Use this value to allow enough time for the command to complete.

Example:

```
[root@localhost ~]# nvme format --ses=1 --namespace-id=1 /dev/nvme0n1 --timeout=1800000
Success formatting namespace:1
[root@localhost ~]#
```

Running dd on SD Cards

The UCS server Linux OS has an integrated **dd** utility. Use this utility to overwrite a UCS server's SD card with zeroes (zero-fill) which erases all data on the SD card.

Before you begin

If you have not read [Before Deleting Data, on page 7](#), read it now.

Procedure

Step 1 Unmount the SD card through the server OS.

Step 2 Run the **dd** command.

Example:

```
[root@localhost ~]# dd if=/dev/zero of=/dev/sd*
[root@localhost ~]#
```

Running ndctl on Intel Optane Persistent Memory

Intel provides the **ndctl** utility to erase the Intel Optane Persistent Memory Modules (PMem). Use **ndctl** utility to erase the DCPMMs in your UCS servers.

Before you begin

- If you have not already read [Before Deleting Data, on page 7](#), read it now.
- Your UCS server must be running Linux kernel version 5.1 or higher to use the **ndctl** tool.

Procedure

Step 1 Unmount the device through the server OS.

Step 2 Run **uname -srm** to check the Linux kernel version.

Example:

```
# uname -srm
# Linux 3.10.0-1062.el7.x86_64 x86_64
```

If the Linux kernel version is less than 5.1, deleting content through **ndctl** will fail.

Step 3 Install **ndctl**.

Go to: <https://github.com/pmem/ndctl>

Step 4 Run **ndctl list -D** to discover the DCPMM inventory.

This command lists all the DCPMM handles.

Example:

```
[root@localhost ~]# ndctl list -D
[
  {
    "dev": "nmem1",
    "id": "8089-a2-1843-000006de",
    "handle": 289,
    "phys_id": 59,
    "security": "disabled"
  },
  {
    "dev": "nmem3",
    "id": "8089-a2-1843-00001652",
    "handle": 4385,
    "phys_id": 71,
    "security": "disabled"
  },
  {
    "dev": "nmem0",
    "id": "8089-a2-1839-0000118f",
    "handle": 33,
    "phys_id": 53,
    "security": "disabled"
  },
  {
    "dev": "nmem2",
    "id": "8089-a2-1843-00000ae8",
    "handle": 4129,
    "phys_id": 65,
    "security": "disabled"
  }
],
]
```

Step 5 Run `ndctl disable-region all` to disable all memory regions.

Step 6 Run `ndctl sanitize-dimm -c -o all -z` to sanitize all installed DCPMMs.

Example:

```
[root@localhost ~]# ndctl sanitize-dimm -c -o all -z
sanitized 4 nmems.
```

Data Sanitization for Microchip Controllers

Introduction

Use the Microchip Command Line Utility ‘`arconf`’ to sanitize data on SAS/SATA HDDs, SAS/SATA SSDs and NVMe SSDs drives that are connected to the following Microchip controllers.

RAID Controllers:

- UCSC-RAID-M1L16
- UCSC-RAID-MP1L32

HBA Controller:

- UCSC-HBA-M1L16

Command syntax

The following command syntax is used to sanitize drives described in these instructions:

arconf TASK START <Controller#>**DEVICE** <Channel# ID#> **secureerase PATTERN**<erasePattern>

The sanitize <erasePattern> values are:

- **4--Crypto Scramble Sanitize Method.** HDDs and SSDs. This option changes the encryption keys on the physical device to prevent correct decryption of previously stored information, which may cause protection information, if any, to be indeterminate.
- **5--Block Erase Sanitize Method.** SSDs only. Erase voltage is applied to all NAND cells.
- **6--Overwrite Sanitize Method.** HDDs only. Initializes blocks using complex multi-byte data pattern.

Before you begin

Download and install the Microchip storage management tools

Before you begin

1. Download Microchip storage management tools **arconf** and **maxView**. [Download link](#)
2. In Linux OS version, extract ‘Utilities’ from the ISO image.
3. Copy it into the DUT server Linux OS.
4. Go to the folder:
Utilities/arconf/1183a438bdccd0491ae3c285e6f20bab/ASM-ARCCONF-LINUX64-ADAP/cdrom1/linux_x64/rpm
5. Enter the following command to install the arconf rpm:

rpm -ivh xxx.rpm

```
[root@localhost rpm]# rpm -ivh Arconf-4.17-26540.x86_64.rpm
Verifying... ##### [100%]
Preparing... ##### [100%]

Arconf
Version 4.17
Updating / installing...
 1:Arconf-4.17-26540 ##### [100%]

Arconf is located at /usr/Arconf
Installation completed successfully
```

476291

6. Go to the following folder:
Utilities/maxview/d4776e396fbc1811920894090a1e24/ASM-MANAGER-LINUX64-ADAP/cdrom1/linux_x64/manager

7. Provide 'execute' privilege to the installment package file.
8. Execute the package file.

```
[root@localhost manager]# ./StorMan-4.17-26540.x86_64.bin
Verifying... ##### [100%]
Preparing... ##### [100%]

maxView Storage Manager
Version 4.17

Updating / installing...
 1:StorMan-4.17-26540 ##### [100%]

Do you want to install maxView as Desktop Web Application: [default: no]:no
Do you want to install maxView in Standalone Mode: [default: no]:no
Starting maxView Storage Manager Redfish Server [ OK ]
Starting maxView Storage Manager WebServer [ OK ]

Installation completed successfully.
```

476292

Prepare the drives: CLI

To prepare the drives, delete the logical drive, check the physical drive, and get the physical drive controller, channel and device ID numbers.

You can use the following commands, or the Microchip maxView GUI tool.

1. Unmount the disks in the OS.
2. Delete the logical drive on the physical drive for RAID controllers.

Note: Make sure it's in 'Ready' state on Microchip RAID controllers.

arconf DELETE <Controller#> **LOGICALDRIVE** <LogicalDrive#>

Example: **arconf DELETE 1 LOGICALDRIVE 0**

```
[root@localhost ~]# arconf DELETE 1 LOGICALDRIVE 0
Controllers found: 1

WARNING: Deleting this logical device will automatically delete array 0 because it is the only logical device present on that
array.
All data in logical device 0 will be lost.
Delete the logical device?
Press y, then ENTER to continue or press ENTER to abort: y

Deleting: logical device 0 ("LogicalDrv 0")

Command completed successfully.
```

476293

3. Check the physical drive state:

arconf GETCONFIG <Controller#> **PD** <Channel# ID#>

Example: **arconf GETCONFIG 1 PD 0 5**

```
[root@localhost ~]# arcconf GETCONFIG 1 PD 0 5
Controllers found: 1
-----
Physical Device information
-----
Device #0
  Device is a Hard drive
  State : Ready
  Drive has stale RIS data : False
  Disk Name : /dev/sdw (Disk0) (Bus: 0, Target: 22, Lun: 0)
  Block Size : 512 Bytes
  Physical Block Size : 512 Bytes
  Transfer Speed : SAS 12.0 Gb/s
  Reported Channel,Device(T:L) : 0,5(5:0)
  Reported Location : Backplane 0, Slot 6(Connector 0:CN2)
  Vendor : TOSHIBA
  Model : AL15SEB060N
  Firmware : 5704
  Serial number : 1860A016FJPF
  World-wide name : 5000039868091E16
  Reserved Size : 32768 KB
  Used Size : 0 MB
  Unused Size : 572293 MB
  Total Size : 572325 MB
```

476294

Prepare the drives: GUI

Use the Microchip tool maxView to delete the logical drive, check the physical drive, and get the physical drive controller, channel and device ID numbers.

The screenshot shows the maxView GUI interface. The top navigation bar includes tabs for System, Controller, Array, Logical Device, Physical Device, and maxCache. The main content area is divided into three sections: Logical Device Info, Settings and Status, and Progress Task. The Logical Device Info section shows details for Logical Drive 0, including RAID Level (0), Device Type (Data), Data Space (5.822 TB), and Member Dev... (512 Bytes). The Settings and Status section shows Status (Optimal), Name (LogicalDrv 0), and Disk Name (/dev/sdd (Dis...)). The Progress Task section shows Task Name (Not Applicable) and Task Percent... (Not Applicable). A red arrow points to the 'Logical Device' icon in the top navigation bar, and another red arrow points to the 'LogicalDrv 0' entry in the sidebar. A red text annotation '2. Delete this logical drive' is visible above the 'Settings and Status' tab.

476295

Sanitize SED drives (HDD and SSD)

Complete the instructions to sanitize a SED drive.

1. Complete PSID revert on SED drives before the data sanitize action using the following command:

```
arconf SEDENCRYPTION <Controller#> REVERTTOOFS device <Channel# ID#> <Device#ID#>
PSID <PSID string>
```

Note: The <PSID string> is on the physical SED drive label.

Example: **arconf SEDENCRYPTION 1 REVERTTOOFS device 0 1 PSID 76619977D0D4928965F0ABC2D2C774E1**

```
[root@localhost ~]# arconf SEDENCRYPTION 1 REVERTTOOFS device 0 1 PSID 76619977D0D4928965F0ABC2D2C774E1
Controllers found: 1
Reverting the Self Encrypting Drive (SED) to its original factory state will destroy all user data stored on the device
Are you sure you want to continue?
Press y, then ENTER to continue or press ENTER to abort: y

Command completed successfully.
```

476296

2. Sanitize SED drives:

```
arconf TASK START <Controller#> DEVICE <Channel# ID#> secureerase PATTERN 4
```

Example: **arconf TASK START 1 DEVICE 0 1 secureerase PATTERN 4**

```
[root@localhost ~]# arconf TASK START 1 DEVICE 0 1 secureerase PATTERN 4
Controllers found: 1
WARNING: Use of SED device(s) may lead to data loss if ownership is taken on the specified device(s).
Secure erase of a Hard drive is a long process.

Are you sure you want to continue?
Press y, then ENTER to continue or press ENTER to abort: y

Secure Erasing Channel 0, Device 1.

Command completed successfully.
```

476297

Sanitize Non-SED HDDs

Sanitize HDDs using the following command. This may take several hours depending on the drive capacity.

1. Sanitize the drive:

```
arconf TASK START <Controller#> DEVICE <Channel# ID#> secureerase PATTERN 6
```

Example: **arconf TASK START 1 DEVICE 0 6 secureerase PATTERN 6**

```
[root@localhost Desktop]# arconf TASK START 1 DEVICE 0 6 secureerase PATTERN 6
Controllers found: 1
Secure erase of a Hard drive is a long process.

Are you sure you want to continue?
Press y, then ENTER to continue or press ENTER to abort: y

Secure Erasing Channel 0, Device 6.

Command completed successfully.
```

476298

Sanitize Non-SED SSDs

Sanitize SSDs, including SAS/SATA SSD and NVMe SSD, using the following command.

This may take several minutes to complete, depending on the drive capacity.

1. Sanitize the drive:

```
arconf TASK START <Controller#> DEVICE <Channel# ID#> secureerase PATTERN 5
```

Example: `arconf TASK START 1 DEVICE 0 14 secureerase PATTERN 5`

```
[root@localhost Desktop]# arconf TASK START 1 DEVICE 0 14 secureerase PATTERN 5
Controllers found: 1
Secure erase of a Hard drive is a long process.

Are you sure you want to continue?
Press y, then ENTER to continue or press ENTER to abort: y

Secure Erasing Channel 0, Device 14.

Command completed successfully.
```

476299

2. After the data erase is complete, initialize the drive using the **Microchip maxView** or **arconf** tool.

arconf command:

```
arconf SETSTATE <Controller#> DEVICE <Channel# ID#>
```

476300



INDEX

D

dd command [19](#)
DIMMs, Optane [19](#)

I

Intel Optane DIMMs [19](#)

N

ndctl [19](#)

non-volatile drives [17](#)
nvme-cli [17](#)

S

sg_sanitize, SSD [12–13](#)
solid state drive [12](#)

T

tri-mode NVMe SSD [13](#)

