Release Notes

ARC Software/Firmware

Released

May 2020

Microsemi

a Microchip company

Revision History

Revision	Revision Date	Details of Change
6	May 2020	Updated for 2020.1 release
5	December 2019	Updated for 2019.2.1 patch release.
4	August 2019	Updated for 2019.2 release.
3	January 2019	Updated for 2019.1 release.
2	April 2018	Updated for 2018.1 release.
1	October 2017	Document created.

Software License Agreement

PLEASE READ CAREFULLY: THE USE OF THIS SOFTWARE IS SUBJECT TO THE SOFTWARE LICENSE TERMS OF MICROSEMI, INC. AND OTHER LICENSORS WHOSE SOFTWARE MAY BE BUNDLED WITH THIS PRODUCT.

BY YOUR USE OF THE SOFTWARE INCLUDED WITH THIS PRODUCT YOU AGREE TO THE LICENSE TERMS REQUIRED BY THE LICENSOR OF THAT SOFTWARE, AS SET FORTH DURING THE INSTALLATION PROCESS. IF YOU DO NOT AGREE TO THE LICENSE TERMS APPLICABLE TO THE SOFTWARE, YOU MAY RETURN THE ENTIRE UNUSED PRODUCT FOR A FULL REFUND.

In return for acquiring a license to use the Microsemi software, which may include software from third party licensors and patches made available by Microsemi ("Software"), and the related documentation, you agree to the following terms and conditions:

1. License. This Agreement grants you, the Licensee, a license to:
   a. Use the Software on a single computer system, which is not intended for use by more than five (5) users; and:
   b. Make one copy of the Software in machine readable form solely for back-up purposes, provided you reproduce Microsemi's copyright proprietary legends. Notwithstanding the foregoing, the Software may be used on the home, laptop or other secondary computer of the principal user of the Software, and an additional copy of the Software may be made to support such use. As used in this license, the Software is "in use" when it is either loaded into RAM or installed on a hard disk or other permanent memory device. The Software may be "in use" on only one computer at any given time. (Different license terms and fees are applicable for networked or multiple user applications.) As a specific condition of this license, you agree to use the Software in compliance with all applicable laws, including copyright laws, and that you will not copy, transmit, perform or distribute any audio or other content using the Software without obtaining all necessary licenses or permissions from the owner of the content.
2. Restrictions. You may not distribute copies of the Software to others or electronically transfer the Software from one computer to another over a network. You may not post or otherwise make available the Software, or any portion thereof, in any form, on the Internet. You may not use the Software in a computer service business, including in time sharing applications. The Software contains trade secrets and, in order to protect them, you may not decompile, reverse engineer, disassemble, or otherwise reduce the Software to a human-perceivable form. YOU MAY NOT MODIFY, ADAPT, TRANSLATE, RENT, LEASE, LOAN, RESELL FOR PROFIT, DISTRIBUTE, NETWORK OR CREATE DERIVATIVE WORKS BASED UPON THE SOFTWARE OR ANY PART THEREOF.
3. Ownership of Software. As Licensee, you own the media upon which the software is recorded or fixed, but Microsemi and its licensors retain title and ownership of the Software recorded on the original media and all subsequent copies of the Software, regardless of the form or media in which or on which the original and other copies may exist. This license is not a sale of the Software or any copy.
4. Confidentiality. You agree to maintain the Software in confidence and that you will not disclose the Software to any third party without the express written consent of Microsemi. You further agree to take all reasonable precautions to preclude access of unauthorized persons to the Software.
5. Term. This license is effective until January 1, 2045, unless terminated earlier. You may terminate the license at any time by destroying the Software (including the related documentation) together with all copies or modifications in any form. Microsemi will have the right to terminate our license immediately if you fail to comply with any term or condition of this Agreement. Upon any termination, including termination by you, you must destroy the Software (including the related documentation), together with all copies or modifications in any form.
6. Special Terms Applicable to Databases. Where a database is included with the Software, you acknowledge that it is licensed only in connection with the use of the Software to perform disc creation, and that the

database and all data derived therefrom must be maintained in confidence in accordance with the provisions of Section 4. This license does not grant you any rights to distribute or disclose such database or data.

7. Limited Warranty. Microsemi and its Licensor warrant only that the media upon which the Software is furnished will be free from defects in material or workmanship under normal use and service for a period of thirty (30) days from the date of delivery to you. MICROSEMI AND ITS LICENSORS DO NOT AND CANNOT WARRANT THE PERFORMANCE OR RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE OR DOCUMENTATION. THE FOREGOING STATES THE SOLE AND EXCLUSIVE REMEDIES MICROSEMI AND ITS LICENSORS WILL PROVIDE FOR BREACH OF WARRANTY. EXCEPT FOR THE FOREGOING LIMITED WARRANTY, MICROSEMI AND ITS LICENSORS MAKE NO WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED, AS TO NON-INFRINGEMENT OF THIRD PARTY RIGHTS, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow the exclusion of implied warranties or limitations on how long an implied warranty may last, so the above limitations may not apply to you. This warranty gives you specific legal rights and you may also have other rights which vary from state to state.
8. The entire liability of Microsemi and its licensors, and your exclusive remedy for a breach of this warranty, shall be:
   a. The replacement of any media not meeting the above limited warranty which is returned to Microsemi; or:
   b. if Microsemi or its distributor is unable to deliver replacement media which is free from defects in materials or workmanship, you may terminate this Agreement by returning the Software and your money will be refunded.
9. Limitation of Liability. IN NO EVENT WILL MICROSEMI OR ITS LICENSORS BE LIABLE TO YOU FOR ANY INCIDENTAL, CONSEQUENTIAL OR INDIRECT DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS, OR LOSS OF DATA, EVEN IF MICROSEMI OR A LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY. Some states do not allow the exclusion or limitation of special, incidental, or consequential damages, so the above limitation or exclusion may not apply to you.
10. Export. You acknowledge that the laws and regulations of the United States and other countries may restrict the export and re-export of the Software. You agree that you will not export or re-export the Software or documentation in any form in violation of applicable United States and foreign law.
11. Government Restricted Rights. The Software is subject to restricted rights as follows. If the Software is acquired under the terms of a GSA contract: use, reproduction or disclosure is subject to the restrictions set forth in the applicable ADP Schedule contract. If the Software is acquired under the terms of a DoD or civilian agency contract, use, duplication or disclosure by the Government is subject to the restrictions of this Agreement in accordance with 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors and 49 C.F.R. 227.7202-1 of the DoD FAR Supplement and its successors.
12. General. You acknowledge that you have read this Agreement, understand it, and that by using the Software you agree to be bound by its terms and conditions. You further agree that it is the complete and exclusive statement of the agreement between Microsemi and you, and supersedes any proposal or prior agreement, oral or written, and any other communication between Microsemi and you relating to the subject matter of this Agreement. No additional or any different terms will be enforceable against Microsemi unless Microsemi gives its express consent, including an express waiver of the terms of this Agreement, in writing signed by an officer of Microsemi. You assume full responsibility for the use of the Software and agree to use the Software legally and responsibly. This Agreement shall be governed by California law, except as to copyright matters, which are covered by Federal law. This Agreement is deemed entered into at Sunnyvale, California by both parties. Should any provision of this Agreement be declared unenforceable in any jurisdiction, then such provision shall be deemed severable from this Agreement and shall not affect the remainder hereof. All rights in the Software not specifically granted in this Agreement are reserved by Microsemi.

Should you have any questions concerning this license, contact:

Microsemi Corporation
Corporate Headquarters
One Enterprise
Aliso Viejo, CA 92656 USA

Contents

1 About This Release

This document describes a maintenance release package of Microsemi's Adaptec® RAID controller firmware, OS drivers, tools, and host management software.

1.1 Release Identification

The firmware, software, and driver versions for this maintenance release are shown in the following table.

Table 1 • Release Summary

Package Release Date	May 22, 2020
Firmware Version	7.16.0 build 33456 (Basecode B833)
UEFI Version	2.0.100.33456
Legacy BIOS Version	2.0.100.33456
Microsemi Adaptec RAID Controller Configuration Utilities (ARCCONF Command Line Interface, maxView Storage Manager, maxView vSphere Plugin, Boot USB, EventMonitor)	3.04.00.23699
Drivers Package Version	B59002 [Windows/Linux/VMware] (2019.2 web released build)

1.2 Supported Controllers

The following controllers are supported:

- Microsemi Adaptec RAID 8405
- Microsemi Adaptec RAID 8405E
- Microsemi Adaptec RAID 8805
- Microsemi Adaptec RAID 8805E
- Microsemi Adaptec RAID 8885
- Microsemi Adaptec RAID 8885Q
- Microsemi Adaptec RAID 81605Z
- Microsemi Adaptec RAID 81605ZQ

2 What is New?

2.1 Fixes and Enhancements

This section details the fixes and enhancements in this release.

2.1.1 Firmware Fixes

2.1.1.1 Fixes and Enhancements for Firmware Release Build 33456

This release provides the following fixes and enhancements:

- Fixed a SATA drive drop issue when drive times out an IO followed by TUR command failure upon drive reset.
  - *Root cause:* When SATA drives times out an IO due to link errors and other conditions, firmware error handling logic resets the drive followed by Test Ready Unit (TUR) command before resubmitting the command. If drive fails TUR command and subsequent TURs retries, drive is marked offline and dropped out of configuration.
  - *Fix:* Error handling logic is modified to include additional drive resets on TUR failure to get the drive back online.
  - *Risk:* Low
- Fixed an issue where host memory can get corrupted when SATA drive is stuck in NCQ error state during IO timeout error handling.
  - *Root cause:* During SATA drive IO timeout error handling, after a drive reset, to get the drive out of any possible NCQ error state condition a PIO READ command is performed on LBA 0. The data transfer direction flag for this command is set incorrectly, resulting in LBA 0 contents transferred to host memory instead of controller memory.
  - *Fix:* The data transfer direction flags for PIO READ read command is set correctly and data is now transferred to the intended controller memory.
  - *Risk:* Low
- Fixed a controller assert when rebooting server with RAID 1E logical volume in rebuilding state.
  - *Root cause*: When rebooting server with rebuilding RAID 1E logical volume, rebuild task would run into a NULL pointer memory access, causing a controller assert.
  - *Fix*: Appropriate NULL memory pointer sanity check is added to RAID 1E rebuild path.
  - *Risk*: Low
- Fixed an issue where any background task that would get initiated on R10, R50, or R60 volumes would always run on high priority.
  - *Root cause*: End user settings on background task priority (low, medium, high) were not correctly propagated to dual RAID level volumes.
  - *Fix*: End user priority settings are now correctly propagating to dual RAID level volumes.
  - *Risk*: Low
- Fixed an occasional issue where RAW drives were not exposed to OS after a drive swap during reboot.
  - *Root cause*: During drive discovery on reboot after a drive swap condition is detected, a minor race condition window between two threads could clear the RAW drive type flag for RAW drives, causing them to be masked from the OS.
  - *Fix*: After drive swap condition is detected, new flags are added to serialize the threads to avoid clearing the RAW drive type for RAW drives.
  - *Risk*: Low

2.1.1.2 Fixes and Enhancements for Firmware Release Build 33401

This release provides the following fixes and enhancements:

- Enabled parallel rebuilds on overlapping logical drive configurations.
- Fixed an issue where the controller would assert during logical drive deletion.
  - *Root cause*: During logical drive deletion, the host command to get logical drive information gets processed with an invalid logical drive number causing assert.
  - *Fix*: Command to get logical drive information returns failure if logical drive number is invalid.
  - *Risk*: Low
- Fixed an issue where the drive slot/bay number was not listed correctly on few sepless expander configuration.
  - *Root cause*: Incorrect code flow caused the drive/bay number to be out of order.
  - *Fix*: Changed code flow so the enclosure initialization logic can list the drive/bay number in order.
  - *Risk*: Low
- Fixed an issue where RAID-configured disk will be exposed to OS after exchanging drive slots with raw state disk.
  - *Root cause*: Under certain conditions, during drive discovery, drive type value gets reset to default value 'RAW drive', which makes the RAID drive exposed to OS.
  - *Fix*: Drive type value is updated correctly at the end of drive discovery phase.
  - *Risk*: Low
- Fixed an issue in PBSI interface where a drive part of a failed array is reported as 'Ready' state drive.
  - *Root cause*: Drive part of failed array is incorrectly reported as 'Not part of an array' instead of 'data drive' as failed partition information is not processed by PBSI module.
  - *Fix*: Failed partition information is processed now to export the drive as 'data drive'.
  - *Risk*: Low
- Fixed a NVSRAM content loss issue during a minor dirty shutdown window.
  - *Root cause*: NVSRAM data protected by CRC, could get corrupted during a small dirty shutdown window, where CRC is calculated using a write/read/write operation.
  - *Fix*: NVSRAM's CRC update is now calculated using a single write operation than a write/read/write operation.
  - *Risk*: Low
- Fixed an issue where product revision of SATA drives are not reported correctly.
  - *Root cause*: When a drives product revision is more than 5 bytes or if it has a NULL character, FW swaps the characters incorrectly leading to a NULL or incorrect product revision string to be displayed in host.
  - *Fix*: Adjustment in product revision byte swap and NULL character handling is added to export product revision in correct order.
  - *Risk*: Low
- Fixed an occasional exception when copyback task terminates.
  - *Root cause*: When copyback task terminates, one of locks may not get freed up properly leading to an exception.
  - *Fix*: In copyback task, lock is now released properly in its task termination path.
  - *Risk*: Low
- Fixed an issue where PBSI interface reports an incorrect GB status as READY when there are NAND-related errors.
  - *Root cause*: PBSI interface processes doesnt process flash NAND-related errors.

- *Fix*: On top of supercap related errors, Flash NAND-related errors are processed now in PBSI module.
- *Risk*: Low

2.1.1.3 Fixes and Enhancements for Firmware Release Build 33303

This release provides the following fixes and enhancements:

- Fixed an issue where RAID 5 Copyback could take up to 15 times longer than Rebuild to complete.
  - *Root cause*: Small block transfer size caused increased copy iterations.
  - *Fix*: Increased block transfer size reducing iterations.
  - *Risk*: Low
- Fixed an issue where a command timeout would lead to an IOP reset.
  - *Root cause*: Large FIB writes (commands) such as 256K could be broken into smaller serialized commands, increasing the time to complete a single command.
  - *Fix*: Check for large FIBS and do not break up, sending larger commands completes faster.
  - *Risk*: Low
- Fixed an issue where a single bad drive would cause drive discovery failure.
  - *Root cause*: Bad drive caused expander PHY connected to controller to be stuck in "reset in progress" state which caused controller discovery failure.
  - *Fix*: Detect this special state and allow controller to boot, detecting all other devices.
  - *Risk*: Low
- Fixed an issue where the UEFI GUI drive speed field would display "NA" instead of proper value (i.e., 12.0G).
  - *Root cause*: Character data structure too small.
  - *Fix*: Increased structure size.
  - *Risk*: Low
- Fixed an issue where the REBUILD LED did not properly display in IBPI configuration.
  - *Root cause*: LEDs not managed per IBPI spec.
  - *Fix*: Set both "locate"/"fault" LEDs for identify device/failed drive activities per IBPI spec. Change rebuild behavior from "blinking fault LED at 1Hz and leave other LEDs as is" back to "set locate and fault LEDs to logic 1.
  - *Risk*: Low
- Fixed an issue where ARC firmware history logs could be lost if firmware assert and power loss.
  - *Root cause*: History log information not preserved if firmware assert during initialization, followed by power loss.
  - *Fix*: Enable offline log preserving.
  - *Risk*: Low
- Fixed an issue where subtractive-to-subtractive inter-expander was misinterpreted.
  - *Root cause*: Subtractive-to-subtractive inter-expander connection was treated as a loop.
  - *Fix*: Add proper detection of subtractive-to-subtractive connection.
  - *Risk*: Low
- Fixed an issue where ARCCONF setstatsdatacollection returned the wrong status, but command is actually successful.
  - *Root cause*: Status not properly returned.
  - *Fix*: Qualified proper status.
  - *Risk*: Low

- Fixed an issue where the UEFI Boot Option Priorities field displayed bad characters.
  - *Root cause*: Incorrect drive structure usage.
  - *Fix*: Resolved drive structure conflict.
  - *Risk*: Low

2.1.1.4 Fixes and Enhancements for Firmware Release Build 33263

This release provides the following fixes and enhancements:

- Added support for the LED to light on the target device during COPYBACK task.
- Fixed an issue where the device was not detected after a hot-add.
- Fixed an issue where a drive command timeout could cause the adapter to reset.
- Fixed an issue where a redundant logical drive could fail after single drive failure.
- Fixed an issue where the UEFI drive would be incorrectly identified.
- Fixed an issue where the logical drive size was inconsistent between BIOS and PBSI.
- Fixed an issue to properly display the Micron 5100 SSD firmware version in PBSI.
- Fixed an issue with a slow shutdown when logical drive was failed or degraded.
- Fixed an issue where the firmware would crash after a rebuild with maxCache enabled.
- Fixed an issue where the OS adapter would reset during multiple-controller Kdump activity.
- Fixed an issue where the logical drive would fail when a metadata write failed during rebuild.
- Fixed an issue to honor PBSI frequency settings.

2.1.1.5 Fixes and Enhancements for Firmware Release Build 33204

This release provides the following fixes and enhancements:

- Removed support for auto-detecting the hot-plugged drives feature from ExtScsiPassThru layer without needing to reload the controller UEFI driver.
  - *Previous behavior*: Drives that are added after the UEFI driver has loaded could be seen by the UEFI driver by making a second call to the function EfiScsiPassThruGetNextTargetLun. This behavior has now been removed.
  - *Current behavior*: To see hot-plugged drives added after the UEFI driver has loaded, you will need to remove and reload the controller UEFI driver and then you will see the new drives. You will no longer see hot-plugged drives by calling the EfiScsiPassThruGetNextTargetLun function.
  - *Risk*: Low
- Added support for physical drive failure events through PBSI event logging.
- Added support for Green Backup events through PBSI event logging.
- Added support to allow RAW drives to remain in a spun-down state when the user spins down the drive.
  - *Exposure*: Applies to RAW drives only.
- Fixed an issue where the console showed a controller cache warning due to a backup unit fault after flashing new firmware in UEFI mode.
- Fixed an issue where the box display order could be inconsistent during POST.
- Fixed an issue with setting the Spare Drive Flag=0 for PBSI event logging.
- Removed bad character from "Physical drive failure" field in PBSI.
- Fixed a timestamp issue in PBSI.
- Fixed an issue where RAW drives were lost when spun down and followed by an ARCCONF RESCAN.
- Fixed an issue with a media change error message in the UEFI script, causing a script failure.

- Fixed an issue where entering the system BIOS, changing and saving a setting, then exiting caused the system to hang (introduce CPLD 12).

  Note:

  CPLD is not activated automatically with a standard firmware flash. To update CPLD, run the following command in ARCCONF: `ARCCONF CPLD 1 FLASHUPDATE`

  See the *Microsemi Adaptec RAID Controller Command Line Utility User's Guide* (ESC-2160659) for more information.
- Fixed an issue where the drive would go offline during an expander reset operation.
- Fixed expander mapping issues.
- Fixed an issue where the Read Buffer would fail.
- Fixed an issue where the controller would hang when the drive member was removed from a RAID 5 array.
- Fixed an issue where the UEFI would freeze while saving the support archive.
- Added support for driver NMI feature to controllers.
- Fixed an issue where the INQUIRY to VPD page 0xB0 WRITE SAME length was leading to IO timeouts.
  - *Root cause*: The INQUIRY to VPD page 0xB0 has a field to specify the maximum WRITE SAME length that the drive can support. SAT-4 had this field as unspecified and therefore SATL was setting it. However, this is the field that the latest Linux version uses to determine the proper WRITE SAME size to send and treats "0" as the sending maximum. This led to IO timeouts.
  - *Exposure*: Setting correct fields in INQUIRY to VPD page 0xB0.
  - *Fix*: Set the WRITE SAME length field to the maximum amount a 512-buffer can accept.
  - *Risk*: None
- Set the SES, STE, SVPD to a higher priority.
  - *Root cause*: The firmware sends an RCV_DIAGNOSTICS command to the VSEP every 10 seconds and will time out this command if it does not receive a completion within 30 seconds. In highly loaded systems, the SVPD thread may not wake up within 30 seconds, causing the firmware to time out the command.
  - *Exposure*: ARC programs
  - *Fix*: Increase the priority of the SES, STE, and SVPD threads. These threads are low-activity threads and should be serviced immediately by the firmware. The high-activity threads, like SAS_FAST, will have a lower priority relative to the SES, STE, and SVPD threads.
  - *Risk*: Low

2.1.2 UEFI/BIOS Fixes

2.1.2.1 Fixes and Enhancements for UEFI and Legacy BIOS Build 33456

There are no fixes or enhancements for this release.

2.1.2.2 Fixes and Enhancements for UEFI and Legacy BIOS Build 33401

This release provides the following fixes and enhancements:

- Fixed an issue where offline drives are displayed as raw drives in HII.
  - *Root cause*: No check for the drives which are set as offline in HII menu where drives are listed.
  - *Fix*: Display text added for case where the drives are set as offline in all drive listing menu.
  - *Risk*: Low
- Fixed an issue where a drive's write cache is not properly configured per global cache policy setting.

- *Root cause*: When global drive cache policy is changed from UEFI, the new setting is not always sent to all the drives.
- *Fix*: Logic to send new global cache policy setting value to all connected drive is added in UEFI.
- *Risk*: Low

- Fixed an issue where the Max and Transfer speed value is not consistent across UEFI and BIOS
  - *Root cause*: UEFI and BIOS displayed different parameters and were therefore inconsistent.
  - *Fix*: Displayed both Programmed max speed and Transfer Speed in HII to make the fields consistent.
  - *Risk*: Low

2.1.3 Driver Fixes

2.1.3.1 Fixes and Enhancements for Windows, Linux, and VMware Drivers Build 59002

This release includes the following Linux driver fixes and enhancements:

- Improved TMF commands for abort and device reset of native raw devices
  - *Root cause*: Incorrect TMF commands were formed for the type HBA_IU_TYPE_SCSI_TM_REQ.
  - *Fix*: TMF requests sent down to the firmware - HBA_TMF_ABORT_TASK and HBA_TMF_LUN_RESET – replace the HBA command request for HBA_IU_TYPE_SCSI_TM_REQ.
  - *Risk*: Low
- Issuing a test lockup during heavy IO will generate crashdump and controller soft reset is observed
  - *Root cause*: Missing controller offline feature during the lockup feature in driver.
  - *Fix*: Introduced new field adapter_panic in aac_dev to handle the offline feature during the Controller lockup in aac_flush_ios, allows controller to reset
  - *Risk*: None

There are no VMware fixes from the previous Build 58012 or Windows fixes from the previous Build 58015.

2.1.3.2 Fixes and Enhancements for Windows Driver Build 58015 and Linux and VMware Drivers Build 58012

This release includes the following Windows driver fixes and enhancements:

- Fixed an issue where logical device command timeouts occurred and caused a controller reset.
  - *Root cause*: When the host stopped sending sequential commands to a logical device, the driver command coalescing logic flag, which was based per controller, was not managed properly and caused a failure of sending coalesced commands to the logical device.
  - *Fix*: The coalescing logic flag is now based per logical device, instead of per controller.
  - *Risk*: Low

There are no Linux or VMware fixes from the previous Build 58012.

2.1.3.3 Fixes and Enhancements for Windows, Linux, and VMware Drivers Build 58012

This release provides the following fixes and enhancements:

- Fixed an issue where reading/writing data beyond LBA will trigger kernel panic.
  - *Root cause*: Inappropriate sense data being set would return the wrong return value resulting in kernel panic.
  - *Fix*: Proper sense data copied with the right return values fixes the issue.
  - *Risk*: Low
- Fixed an issue where the sysfs entries were not consistent between AACRAID and smartpqi drivers.

- *Root cause*: Both drivers developed separately had only some fields in common and few additional fields which were not of primary interest.
- *Fix*: Ensure consistency between the two drivers, and retain important values so that a script could be used to parse them.
- *Risk*: Low

2.1.3.4 Fixes and Enhancements for Windows Driver Build 57011 and Linux and VMware Drivers Build 57013

This release includes the following Windows driver fixes and enhancements:

- Added support for Windows 2019 driver.

This release includes the following Linux driver fixes and enhancements:

- Added retpoline support to correct potential spectre/meltdown issues for OEL 6.9 with both UEK4 and UEK3.
- Use command lsblk to show drives behind the controller are removable.
  - *Root cause*: The SCSI device removable bit was set to 1 in the driver code.
  - *Fix*: Added module parameter aac_removable (to pass while loading a module) which indicates whether the drive is removable or not. (aac_removable = 0: Not removable, aac_removable = 1: Removable)
- Fixed an issue where the build packaging changes and version numbers in modinfo and the rpm file did not match.
  - *Root cause*: To add a subversion (-1) after the version in UEK binaries. "-1" indicates a bump in driver version and the subsequent script changes will be "-2" and so on, when the driver code does not change.
  - *Fix*: Changed the Oracle Linux UEK spec file and the aacraid_build script to add an extra literal.
- Changed timeout in ARC to 60 seconds.
  - *Root cause*: IOP_RESET occurs more frequently.
  - *Fix*: Changed "AAC_ARC_TIMEOUT = 45" to "AAC_ARC_TIMEOUT = 60".
- Fixed an issue where the driver was sending IOP reset and causing the firmware to assert.
  - *Root cause*: The driver was sending IOP reset and causing the firmware to assert.
  - *Fix*: Issue drop I/O before changing interrupt mode to prevent firmware from asserting when sending IOP reset.
- Fixed an issue where using kickstart to install the OS leads to AACRAID driver install failure during pre-install steps.
  - *Root cause*: Pre-install would fail if the "/lib/modules/`uname -r`/updates" directory did not exist.
  - *Fix*: Removed the %pre section and the subsequent builds to be handled.
- Fixed an issue to prevent resets during Extended Error Handling scenario.
  - *Root cause*: Extended Error Handling mechanism gets triggered during error handling reset.
  - *Fix*: Check the bit that indicates if Extended Error Handling is enabled and required before any PCI BAR register access is done. If enabled, then disallow the regular error handling path.
- Added support for consistent naming convention for RPMs in UEK R3 and R4.
- Fixed an issue where the OEL/UEK matrix was broken.
  - *Root cause*: OEL/UEK matrix was broken.
  - *Fix*: Added UEK R4 support for OEL 7.2 and UEK R3 support for OEL 6.8.
- Fixed an issue where the controller could not detect the disk after doing a hot plug test.
  - *Root cause*: The OS was not exposing a drive when host_target_num=256.

- *Fix*: Properly initialized a device element in the driver to allow a device rescan after hot-adding the drive

- Fixed an issue where ARCCONF stopped working after savesupportarchive was executed.
  - *Root cause*: All management commands are utilized at some point while executing the "arcconf savesupportarchive" command. So the driver was not able to perform the requested command.
  - *Fix*: Use aac_fib_free() api to free the management commands instead of aac_fib_free_tag() in aac_intr_normal().
- Fixed an issue where the Device Identification VPD page(0x83) was not supported for logical drive in SUSE11 SP3.
  - *Root cause*: VPD page(0x83) is not supported for SUSE kernel.
  - *Fix*: Added VPD page(0x83) for SUSE kernel.

2.1.3.5 Fixes and Enhancements forWindows and Linux Drivers Build 56008 and VMware Drivers Build 56009

This release includes the following Windows driver fixes and enhancements:

- Fixed an issue where the copyright and provider information were incorrect.
  - *Root cause*: The copyright and provider were incorrect.
  - *Fix*: Updated the copyright and provider information.
  - *Risk*: Low

This release includes the following Linux driver fixes and enhancements:

- Fixed an issue where reset functions were broken up into their individual functions.
  - *Root cause*: The driver only had two reset functions, which were broken up into their individual functions for device, target, bus, and host.
  - *Fix*: This was an upstream patch, which was propagated into the out-of-box driver.
  - *Risk*: Low
- Backported inbox changes from 50740 to an out-of-box driver.
  - *Root cause*: In order to harmonize in-box/out-of-box drivers, sync changes between the two drivers.
  - *Fix*:
  - *Risk*: Low
- Fixed an issue with the static code analysis.
  - *Root cause*: Correct static code analysis issues.
  - *Fix*:
  - *Risk*: Low
- Fixed an issue where some devices may be offline after a reset.
  - *Root cause*: Devices which caused errors may be offline after reset.
  - *Fix*: Reset offline devices to be available after reset.
  - *Risk*: Low
- Reworked the `IOP_Reset` functionality.
  - *Root cause*: In-box driver 50834 included changes to the `IOP_Reset` functionality.
  - *Fix*: Backported these changes to the inbox driver.
  - *Risk*: Low
- Fixed an issue where the ARCCONF reset command could hang.
  - *Root cause*: Need to ensure that we are the only user of ioctl commands.

- *Fix*: Protect access using the ioctl mutex.
- *Risk*: Low

- Added support for kdump on 3 or more adapters.
  - *Root cause*: When using reset devices, an IOP Reset for each controller will take 45 seconds, while the timeout for bringing devices online is 120 seconds. A method was required to perform a soft reset that will not take as long as an IOP Reset.
  - *Fix*: Incorporated the soft-reset functionality, and dropped IO in firmware to allow the kdump to be able to handle more than two controllers.
  - *Risk*: Medium
- Fixed an issue where the ARCCONF does not respond to commands after lockup on second controller.
  - *Root cause*: Check for controller failure during ioctl processing.
  - *Fix*: Included a check for controller health prior to executing ioctl.
  - *Risk*: Low
- Fixed an issue where a controller would unexpectedly reset due to a lack of resources.
  - *Root cause*: Some fibs would not be returned to the management free pool.
  - *Fix*: Included a check to determine which pool a fib belongs to and return appropriately.
  - *Risk*: Low
- Fixed an issue where kdump would not function on RHEL 7.1.
  - *Root cause*: Kdump on RHEL 7.1 would only allow 30 seconds prior to issuing a SIGKILL, which caused the driver to report an error.
  - *Fix*: Included a workaround for RHEL 7.1, which would not be interruptible during kdump.
  - *Risk*: Low
- Backported a timespec64 change from the Linux kernel.
  - *Root cause*: The current time methods would rollover, due to 32-bit values.
  - *Fix*: Use the 64-bit time methods.
  - *Risk*: Low
- Hardened support for AER errors.
  - *Root cause*: Include support to handle all soft/hard AER error.
  - *Fix*:
  - *Risk*: Low

This release includes the following VMware driver fixes and enhancements:

- Fixed a VMware issue that was causing the driver to fail.
  - *Root cause*: Some Linux functions are not available on VMware.
  - *Fix*: Implement compatibility functions for VMware.
  - *Risk*: Low

2.1.3.6 Fixes and Enhancements for Drivers Build 55022

This release includes the following Windows driver fixes and enhancements:

- Fixed an issue where the IO would be lost during heavy IO, resulting in an IOP reset.
  - *Root cause*: A race condition exists in MSI-X mode which would cause the Fib pointer to be cleaned up for the wrong handle.
  - *Fix*: Use the MSIlock() for locking when using MSI-X.
  - *Exposure*: Affects Windows driver for Series-6, 7, 8, and SmartRaid.
  - *Risk*: Low

This release includes the following Linux and VMware driver fixes and enhancements:

- Fixed an issue where a driver could crash during heavy IO runs with 64 RAID volumes.
  - *Root cause*: A call trace was seen when running heavy IO on 64 containers, which included driver re-entry after completing a command.
  - *Fix*: Use atomic memory allocation for the buffer, as it is in interrupt mode.
  - *Exposure*: All Linux versions.
  - *Risk*: Low

2.1.4 Management Software Fixes

2.1.4.1 Fixes and Enhancements for maxView Storage Manager/ARCCONF Version 3.04.00 Build 23699

This release includes the following fixes and enhancements.

- Fixed an issue where logical device status is shown as “Degraded Rebuilding” on two logical devices instead on one rebuilding logical device.
  - Root Cause: Status of the logical device is set to "Degraded Rebuilding” based on the ‘Rebuilding’ state of the physical device.
  - Fix: Added changes to consider the logical device chunk information along with the physical device state to set the logical device state.
  - Exposure: Build 23488 and further releases
  - Risk: Low
- Fixed an issue in ESXi where messages in /var/log/NRMFSADebug.log caused an issue of "var ramdisk is full".
  - Root Cause: Invalid attributes read from NRMConfig file for Series-8 controller to cause overwriting into log file.
  - Fix: Added check for ESXi to not read the unsupported attributes from NRMConfig file.
  - Exposure: 23600 and further releases
  - Risk: Low
- Fixed an issue where expansion of logical device is not working with Series 8 in arcconf.
  - Root Cause: A unsupported condition for Smart Array controller was blocking expansion of logical device if new hard drive is added to expansion list.
  - Fix: Added a valid condition to allow expansion of logical device to allow adding new hard drives for logical device expansion
  - Exposure: Build 23600 and further releases
  - Risk: Low

2.1.4.2 Fixes and Enhancements for maxView Storage Manager/ARCCONF Version 3.02.00 Build 23600

This release provides the following fixes and enhancements:

- Fixed an issue where ARCCONF was unresponsive after device firmware update with 256K chunk size.
  - *Root cause*: Providing unsupported chunk size as input to IMAGEUPDATE command resulted in unresponsive controller.
  - *Fix*: Added changes to validate the input chunk size and provide a valid response when user provides with invalid chunk size.
  - *Risk*: Low
- Fixed an issue where SMART data is not displayed for physical devices in standalone mode of maxView.
  - *Root cause*: The response buffer was corrupt due to invalid memory access resulting in failure to display the information in maxView with Standalone mode.

- *Fix*: The invalid memory access is fixed to send the proper response buffer to be displayed in maxView for SMART data.
- *Risk*: Low

2.1.4.3 Fixes and Enhancements for maxView Storage Manager/ARCCONF Version 2.06.00 Build 23488

This release provides the following fixes and enhancements:

- Fixed an issue where a warning message was missing when a customer created multiple logical devices on the same physical devices.
  - *Root cause*: Need to add a warning if multiple logical devices are created on a device that is online.
  - *Fix*: Added a warning if multiple logical devices are created on a device that is online.
- Fixed an issue where the default size was not accepted for the expansion of RAID1 created with SSDs in maxView.
  - *Root cause*: While expanding a logical device from maxView, max size and user input size were not compared properly and the default value was not accepted.
  - *Fix*: Made the changes to compare the max size with the user input size to accept default size.
- Fixed an issue where RAID level migration by specifying the size as MAX was failing in ARCCONF.
  - *Root cause*: Validation of stripe size, size, RAID level, and legs parameters were missing in ARCCONF while blocking the modify command.
  - *Fix*: Added validation based on stripe size, size, RAID level, and legs parameters for RAID level migration.
- Fixed an issue where lane information was missing in maxView.
  - *Root cause*: New implementation to display the lane information of connector maxView.
  - *Fix*: Added changes to display the lane information Connector tab in maxView.
- Fixed an issue where creating a logical device failed in German Windows server.
  - *Root cause*: In maxView, an exception while formatting the size in German locale failed to create a logical device.
  - *Fix*: Made changes to format the size according to locale while populating the size tool-tip in maxView.
- Fixed an issue where maxView email notification was not working.
  - *Root cause*: Instead of TLS, the mail server was using deprecated SSL, which failed to deliver the email notifications.
  - *Fix*: Added changes to use SSL as a fallback mechanism for secure communication for the mail server when TLS connection fails.
- Fixed an issue where firmware upgrade for Micron 5100 SATA device failed.
  - *Root cause*: Mode 3 support is failing for SATA device firmware upgrade due to wrong parameters for ATA passthrough command.
  - *Fix*: Made changes to provide valid input parameter for modes 3 and 7 for SATA firmware upgrade through ATA passthrough command.

2.1.4.4 Fixes and Enhancements for maxView Storage Manager/ARCCONF Version 2.06.00 Build 23164

This release provides the following fixes and enhancements:

- Reinstated a check for ESXi memory allocation.
- Added a condition for controlling out-of-bound memory access.
- Added code to hold `ldname` in temporary variable and set `ldname` after `memset`.

2.1.4.5 Fixes and Enhancements for maxView Storage Manager/ARCCONF Version 2.05.00 Build 22932

This release provides the following fixes and enhancements:

- Added support for executing UART commands through ARCCONF.
- Added a new command in ARCCONF to enable/disable SMART poll on the controller for RAW drives.
- Fixed an issue where exceeding the backup unit temperature caused a permanent warning sign to display on the controller.
  - *Root cause*: There was no check to not render the warning icon when the CAP temperature in the backup unit went back to normal.
  - *Fix*: When the CAP temperature is back to normal, the rendering of the orange warning icon for CAP temperature is blocked.
  - *Risk*: Low
- Fixed an issue where every user was being assigned an administrator role regardless of their original role.
  - *Root cause*: Using the domain name, every user was getting the Admin role irrespective of his original role.
  - *Fix*: Correct user roles are identified using domain-level authentication as well.
  - *Risk*: Low
- Miscellaneous UI changes/enhancements.

2.2 Limitations

This section details the limitations in this release.

2.2.1 Firmware Limitations

2.2.1.1 Limitations for Firmware Build 33401

There are no known limitations for this release.

2.2.1.2 Limitations for Firmware Build 33303

There are no known limitations for this release.

2.2.1.3 Limitations for Firmware Build 33263

There are no known limitations for this release.

2.2.1.4 Limitations for Firmware Build 33204

There are no known limitations for this release.

2.2.2 UEFI/BIOS Limitations

2.2.2.1 Limitations for UEFI and Legacy BIOS Build 33456

There are no known limitations for this release.

2.2.2.2 Limitations for UEFI and Legacy BIOS Build 33401

There are no known limitations for this release.

2.2.3 Driver Limitations

2.2.3.1 Limitations for Windows, Linux, and VMware Drivers Build 58012

There are no known limitations for this release.

2.2.3.2 Limitations for Windows Driver Build 57011 and Linux and VMware Drivers Build 57013

There are no known limitations for this release.

2.2.3.3 Limitations forWindows and Linux Drivers Build 56008 and VMware Drivers Build 56009

There are no known limitations for this release.

2.2.3.4 Limitations for Drivers Build 55022

There are no known limitations for this release.

2.2.4 Management Software Limitations

2.2.4.1 Limitations for maxView Storage Manager/ARCCONF Version 3.04.00 Build 23699

There are no known limitations for this release.

2.2.4.2 Limitations for maxView Storage Manage/ARCCONF Version 3.02.00 Build 23699

This release includes the following limitation:

- Drive firmware update will not be supported for Intel SSD 3510 model.

2.2.4.3 Limitations for maxView Storage Manage/ARCCONF Version 3.02.00 Build 23600

This release includes the following limitation:

- Drive firmware update will not be supported for Intel SSD 3510 model.

2.2.4.4 Limitations for maxView Storage Manager/ARCCONF Version 2.06.00 Build 23488

There are no known limitations for this release.

2.2.4.5 Limitations for maxView Storage Manager/ARCCONF Version 2.06.00 Build 23164

There are no known limitations for this release.

2.2.4.6 Limitations for maxView Storage Manager/ARCCONF Version 2.05.00 Build 22932

This release includes the following limitation:

- The locate logical device operation in GUI displays a blank dialog box.

  WORKAROUND: Use the physical device locate operation to locate the logical device member drives.

Microsemi
2355 W. Chandler Blvd.
Chandler, AZ 85224 USA

Within the USA: +1 (480) 792-7200
Fax: +1 (480) 792-7277

Microsemi Corporation, a subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), and its corporate affiliates are leading providers of smart, connected and secure embedded control solutions. Their easy-to-use development tools and comprehensive product portfolio enable customers to create optimal designs which reduce risk while lowering total system cost and time to market. These solutions serve more than 120,000 customers across the industrial, automotive, consumer, aerospace and defense, communications and computing markets. Headquartered in Chandler, Arizona, the company offers outstanding technical support along with dependable delivery and quality. Learn more at **www.microsemi.com**.

ESC-2171642