SCS Package Release: SAS3 Web Posting Package

Generate PDF

ReleaseOrderPkg ID: SCGCQ01167961 Open In CQWeb Download
Headline: SAS3 Phase 13 - External Web Posting for invader/Fury Boards
Package_Type: SAS3 Web Posting Package
Tier Account: LSI Generic
Package_Version: SAS3 1.0
Date Generated: Aug 25, 2016

- SCGCQ01141661 - GCA Release: IRCU_MPT_GEN3_Phase13.0 - 14.00.00.00
- SCGCQ01141636 - GCA Release: FLASH_MPT_GEN3_Phase13.0 - 14.00.00.00
- SCGCQ01141489 - GCA: BIOS_MPT_GEN3_Phase13.0 - 8.31.00.00 BIOS
- SCGCQ01141463 - GCA: SAS3 IT/IR UEFI Boot Services Driver and HII utility v15.00.00.00
- SCGCQ01140808 - GCA Release: SAS3FW_Phase13.0 - 13.00.00.00 Firmware IT IR Release
- SCGCQ01137086 - Linux Phase 13 GCA release.
- SCGCQ01135878 - FreeBSD SAS3 IT/IR: mpt driver GCA release with v14.00.00.00
- SCGCQ01108847 - GCA: LSI_SAS3 v2.51.15.00 Phase 13 Windows Driver
- SCGCQ00179278 - SAS2 NVDATA Parser Tool Version 09.00.00.00

GCA Release (SCGCQ01141661) - SAS3IRCU - 14.00.00.00 ^

ReleaseOrderID: SCGCQ01141661 Open In CQWeb Download
Headline: GCA Release: IRCU_MPT_GEN3_Phase13.0 - 14.00.00.00
Release Version: 14.00.00.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Applications
UCM Project: IRCU_MPT_GEN3_Phase13.0
Release Type: GCA
UCM Stream: IRCU_MPT_GEN3_Phase13.0_Rel
Owner: buildsvc
Release Baseline: IRCU_MPT_GEN3_Phase13.0-2016-07-21-14.00.00.00_REL_1469100659@\SAS2
Release Date / Time: 25-JUL-16
Release Type: GCA

GCA Release (SCGCQ01141636) - SAS3FLASH - 14.00.00.00 ^

ReleaseOrderID: SCGCQ01141636 Open In CQWeb Download
Headline: GCA Release: FLASH_MPT_GEN3_Phase13.0 - 14.00.00.00
Release Version: 14.00.00.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Applications
UCM Project: FLASH_MPT_GEN3_Phase13.0
Release Type: GCA
UCM Stream: FLASH_MPT_GEN3_Phase13.0_Rel
Owner: buildsvc
Release Baseline: FLASH_MPT_GEN3_Phase13.0-2016-07-21-14.00.00.00_REL_1469099330@\SAS2
Release Date / Time: 22-JUL-16
Release Type: GCA

GCA Release (SCGCQ01141489) - BIOS - 8.31.00.00 ^

ReleaseOrderID: SCGCQ01141489 Open In CQWeb Download
Headline: GCA: BIOS_MPT_GEN3_Phase13.0 - 8.31.00.00 BIOS
Release Version: 8.31.00.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_BIOS
UCM Project: BIOS_MPT_GEN3_Phase13.0
Release Type: GCA
UCM Stream: BIOS_MPT_GEN3_Phase13.0_Main_Dev
Owner: shivanis
Release Baseline: BIOS_MPT_GEN3_Phase13.0_Main_Dev_2016-07-21@\SAS2
Release Date / Time: 29-JUL-16
Release Type: GCA

GCA Release (SCGCQ01141463) - UEFI BSD - 15.00.00.00 ^

ReleaseOrderID: SCGCQ01141463 Open In CQWeb Download
Headline: GCA: SAS3 IT/IR UEFI Boot Services Driver and HII utility v15.00.00.00
Release Version: 15.00.00.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_BIOS
UCM Project: UEFIBSDHII_MPT_GEN3_Phase13.0
Release Type: GCA
UCM Stream: UEFIBSDHII_MPT_GEN3_Phase13.0_Rel
Owner: shivanis
Release Baseline: UEFIBSDHII_MPT_GEN3_Phase13.0_Rel_2016-07-21@\SAS2
Release Date / Time: 29-JUL-16
Release Type: GCA

GCA Release (SCGCQ01140808) - Firmware - 13.00.00.00 ^

ReleaseOrderID: SCGCQ01140808 Open In CQWeb Download
Headline: GCA Release: SAS3FW_Phase13.0 - 13.00.00.00 Firmware IT IR Release
Release Version: 13.00.00.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Controller_FW
UCM Project: SAS3FW_Phase13.0
Release Type: GCA

UCM Stream: SAS3FW_Phase13.0_Rel
Owner: buildsvc
Release Baseline: SAS3FW_Phase13.0-2016-07-20-13.00.00.00_REL_1469005660@\SAS_CTRL_FW
Release Date / Time: 29-JUL-16
Release Type: GCA

GCA Release (SCGCQ01137086) - Linux Driver - RHEL5-6_SLES10-11 - 14.00.00.00-1

ReleaseOrderID: SCGCQ01137086 Open In CQWeb Download
Headline: Linux Phase 13 GCA release.
Release Version: 14.00.00.00-1
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Drivers
UCM Project: LINUX_RH_SL_OEL_CTX_MPT_GEN3_PHASE13.0
Release Type: GCA
UCM Stream: LINUX_RH_SL_OEL_CTX_MPT_GEN3_PHASE13.0_Dev
Owner: vt017885
Release Baseline: LINUX_RH_SL_OEL_CTX_MPT_GEN3_PHASE13.0_Dev_2016-08-11@\SAS2
Release Date / Time: 16-AUG-16
Release Type: GCA

GCA Release (SCGCQ01135878) - Free BSD Driver - 14.00.00.00-1

ReleaseOrderID: SCGCQ01135878 Open In CQWeb Download
Headline: FreeBSD SAS3 IT/IR: mpt driver GCA release with v14.00.00.00
Release Version: 14.00.00.00-1
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Drivers
UCM Project: FREEBSD_MPT_GEN3_PHASE13.0
Release Type: GCA
UCM Stream: FREEBSD_MPT_GEN3_PHASE13.0_Rel
Owner: srreddy
Release Baseline: FREEBSD_MPT_GEN3_PHASE13.0_Rel_2016-07-13_v14.00.00.00@\SAS2
Release Date / Time: 22-JUL-16
Release Type: GCA

GCA Release (SCGCQ01108847) - Windows Driver - 2.51.15.00

ReleaseOrderID: SCGCQ01108847 Open In CQWeb Download
Headline: GCA: LSI_SAS3 v2.51.15.00 Phase 13 Windows Driver
Release Version: 2.51.15.00
Product Org: SCS
Product Gen: SAS3
Product Family: SAS3_Drivers
UCM Project: WINDOWS_MPT_GEN3_PHASE13.0
Release Type: GCA
UCM Stream: WINDOWS_MPT_GEN3_PHASE13.0_MAIN_DEV
Owner: ssamant
Release Baseline: WINDOWS_MPT_GEN3_PHASE13.0_SDV_issue_Int_2016-08-17@\SAS2
Release Date / Time: 19-AUG-16
Release Type: GCA

Windows Driver Defects Fixed (11):

Defect ID: SCGCQ01092212
Headline: SAS3 Windows Driver:- Drive enumeration or Drive discovery failed from Intruder/Cutlass card on Windows_8.1_x86 OS
Description Of Change: The inconsistency in INF file is fixed. In INF [SourceDisksFiles.amd64] section renamed to just [SourceDisksFiles] so it matched up with [SourceDisksNames].
Issue Description: Config details:-

OS:- Windows 8.1 x86 client
HBA:- Intruder/Cutlass
Driver :- 2.51.13.00

HBA --> Expander--> Enclosure (SAS drives)
    --> Jbod (SATA & SAS drives)

1. Installed Intruder/Cutlass card in the system and flashed latest phase13 FW on the card.
2. System booted to Windows 8.1 x86 OS, and installed latest Phase-13 windows 8.1 x86 driver. (win8.1_x86 driver)
3. driver installed successfully and HBA discovered from the OS.
4. Connected enclosure and a JBOD as shown above.
5. Drive discovery failed, unable to list the drives from device manager.
6. Drivers were listed successfully from FW (uart:- pl status output)
7. While rebooting the system, From BIOS post Drivers discovered attached to HBA.

Same is working with Fury/invader card.

Steps To Reproduce:
1. Installed Intruder/Cutlass card in the system and flashed latest phase13 FW on the card.
2. System booted to Windows 8.1 x86 OS, and installed latest Phase-13 windows 8.1 x86 driver. (win8.1_x86 driver)
3. driver installed successfully and HBA discovered from the OS.
4. Connected enclosure and a JBOD as shown above.
5. Drive discovery failed, unable to list the drives from device manager.
6. Drivers were listed successfully from FW (uart:- pl status output)
7. While rebooting the system, From BIOS post Drivers discovered attached to HBA.

Defect ID: SCGCQ01095525
Headline: Remove the logging of Null SRB context when the SMID is staticMsg1SMID.
Description Of Change: Static Message1 is not part of tracking array so any command sent using static message 1 should not have any SRB to check from tracking array.
Issue Description: TUR ready command as part of Lun Reset is sent by static msg1 , in max cases this commands times out resulting unnecessary logging when we get the address reply for it when LUn reset sent to FW.
Steps To Reproduce: When the TUR times out , you can see in the rttrace RT_EVID_ERROR_COMPLETE_NULL_SRB is being logged and in windows system event MPI_LOG_NULL_SRB_BAD_CONTEXT is always logged.

Defect ID: SCGCQ01107003
Headline: SAS3 Windows Driver : Windows OS crashes while running IOs and TMs
Description Of Change: when we got through the Complete IO routine because of LUN TUR timer call back , the code was not returning from completeIO therefore accessing the NULL exSRB.
Issue Description: Config details

Server :- DELL T7600
OS:- Windows 2016 server
Driver :- 2.51.14.00
HBA:- Cutlass

\>>Installed latest Driver and successfully discovered all the drives attached to the HBA.
>> Started IOs on the bare drives, kept running one hour.
\>>Then triggered TMs in the background.
\>>after two iterations of TMs , system crashed with "Driver_IRQL_not_less_or_equal(LSISAS3.sys)"

Steps To Reproduce: >>Installed latest Driver and successfully discovered all the drives attached to the HBA.
>> Started IOs on the bare drives, kept running one hour.
>>Then triggered TMs in the background.
>>after two iterations of TMs , system crashed with "Driver_IRQL_not_less_or_equal(LSISAS3.sys)"

Defect ID: SCGCQ01162603

Headline: To fix SDV issue

Description Of Change: all reported errors addressed.

Issue Description: To fix SDV issue

Steps To Reproduce: Please run the SDV to get the reported error .

Defect ID: SCGCQ01086126

Headline: Correct day-one error in LUN Reset Handling Logic

Description Of Change: Modified the loop as follows.

```
        // Scan tracking array looking for a stuck IO on the reset domain (Target for Target Reset, LUN for LUN Reset)
        for (index = 1; index <= DeviceExtension->MaxNumOfRequests; index++)
        {
            // Get pointer to current entry in tracking array
            MFABuffer = &DeviceExtension->SMFAMessageArray[index];

            // Only examine active tracking array entries
            if (MFABuffer->Saved_SRB)
            {
                // Does the TM match the scope of this entry in the tracking array
                if (MFABuffer->AbortParams.params.devHandle == devHandle)
                {
                    // If this is a LUN reset but on another LUN, then just skip to next entry in tracking array
                    if (taskType == MPI2_SCSITASKMGMT_TASKTYPE_LOGICAL_UNIT_RESET &&
                        MFABuffer->AbortParams.params.lun != lun)
                    {
                        continue;
                    }

                    // We have found a pending IO on the tracking array for the domain being reset
                    foundPendingIo = TRUE;

                    if (--DeviceExtension->LunTargetResetPendingIoRetryCount)
                    {
                        // Found stuck IO, but count has not expired so start timer and retry later
                        *pState = LUN_TARGET_RESET_STATE_START_TIMER;
                    }
                    else
                    {
                        // Count has expired, so no retries remain
                        // There is an active outstanding I/O for this scope (STUCK!)
                        // Flag the error to be handled in FINISH state below, and generate a log event
                        LunTargetResetFailed = TRUE;
                        *pState = LUN_TARGET_RESET_STATE_FINISH;
#if _WIN32_WINNT >= WINDOWS_8
                        RTT_LOG_EVENT(RT_EVID_TM_RESET_ACTIVE_IO_AFTER_RESET, &(SrbExt->bus), 3);
#else
                        RTT_LOG_EVENT(RT_EVID_TM_RESET_ACTIVE_IO_AFTER_RESET, &(Srb->PathId), 3);
#endif
                    }
                    break;
                }
            }
        }
```

Issue Description: This issue will need to be CSET back to LSI_SAS2 and LSI_SAS3.

This issue could cause a LUN reset to be completed with a successful status even though it failed to abort all pending IOs on that LUN. This would only occur if there were a LUN reset in-process on another LUN at the same time, which is likely when generating lots of resets as during testing.

The below loop as implemented will stop advancing the MFABuffer pointer once it finds a LUN reset on another LUN. The loop will still run to the end, but MFABuffer will not be advanced beyond that entry so no following entries will be checked and IOs could be pending on that LUN and be missed as a result.

This code is in reset.c module ProcessLunTargetReset (ca_util.c function ProcessTmResetReply in older drivers).

```
        // go through tracking array looking for outstanding I/O's of this scope
        MFABuffer = &DeviceExtension->SMFAMessageArray[1];

        // Scan tracking array looking for a stuck IO on the reset domain (Target for Target Reset, LUN for LUN Reset)
        for (index = 1; index <= DeviceExtension->MaxNumOfRequests; index++)
        {
            // Only examine active tracking array entries
            if (MFABuffer->Saved_SRB)
            {
                // Does the TM match the scope of this entry in the tracking array
                if (MFABuffer->AbortParams.params.devHandle == devHandle)
                {
                    // If this is a LUN reset but on another LUN, then just skip to next entry in tracking array
                    if (taskType == MPI2_SCSITASKMGMT_TASKTYPE_LOGICAL_UNIT_RESET &&
                        MFABuffer->AbortParams.params.lun != lun)
                    {
                        continue;
                    }

                    // We have found a pending IO on the tracking array for the domain being reset
                    foundPendingIo = TRUE;

                    if (--DeviceExtension->LunTargetResetPendingIoRetryCount)
                    {
                        // Found stuck IO, but count has not expired so start timer and retry later
                        *pState = LUN_TARGET_RESET_STATE_START_TIMER;
                    }
                    else
                    {
                        // Count has expired, so no retries remain
                        // There is an active outstanding I/O for this scope (STUCK!)
                        // Flag the error to be handled in FINISH state below, and generate a log event
                        LunTargetResetFailed = TRUE;
                        *pState = LUN_TARGET_RESET_STATE_FINISH;
#if _WIN32_WINNT >= WINDOWS_8
                        RTT_LOG_EVENT(RT_EVID_TM_RESET_ACTIVE_IO_AFTER_RESET, &(SrbExt->bus), 3);
#else
                        RTT_LOG_EVENT(RT_EVID_TM_RESET_ACTIVE_IO_AFTER_RESET, &(Srb->PathId), 3);
#endif
                    }
                    break;
                }
            }
            MFABuffer++;
        }
```

Steps To Reproduce: N/A

Defect ID: SCGCQ01091762

Headline: Add event throttling to selected RTTRACE events to prevent log overrun when ResetActive is set

Description Of Change: Added throttling to specific RTTRACE events to prevent them from overrunning the log buffer during some failure scenarios involving ResetActive being set.

Issue Description: A number of RTTRACE events spew out constantly when ResetActive is set, which overruns the log file and keeps needed log history from being available. These logs need to be throttled to prevent this from occurring.

Steps To Reproduce: Use a Windows kernel debugger to set DeviceExtension->ResetActive to 1.
Run an IO load on the driver.
Collect the RTTRACE logs from the driver to see that it has been overrun with lots of these events.

Defect ID: SCGCQ01094588

Headline: Fix day-one bug in handling of tracking array

Description Of Change: Moved setting of Saved_SRB to after the pointers have been changed.

Issue Description: A day-one bug was discovered in handling of the tracking array which is surfacing now due to the additional activity caused by activating multi-threaded ISRs. The Saved_SRB field in the tracking array, which is used to determine when an entry is free and available for use, is being cleared before the pointers to the message FIFO queue are updated. The entry should only be marked as free once all pointers and other fields in the tracking array entry have been completely updated.

Steps To Reproduce: N/A

Defect ID: SCGCQ01095072
Headline: Srb and SrbExt should not be accessed after completing Srb
Description Of Change: Moved capture of data for RTTRACE event to occur before the SRB is completed back to StorPort.
Issue Description: In ca_int.c, in the section that is completing Success IOs, there are RTTRACE logs being created which access the Srb and SrbExt pointers after the Srb has been completed. We have learned from MR that if processing is interrupted at just the right time, Storport can reuse that Srb (modifying many elements) and those accesses will be getting data for the new Srb and not the Srb that is being completed. The RTTRACE code should be executed before the call to StorPortNotification(RequestComplete).

Defect noticed during code inspection.

Note that these RTTRACE logs only execute when tracing for the main IO path is enabled, which it is disabled in all production drivers for performance.

Steps To Reproduce: N/A

Defect ID: SCGCQ01115185
Headline: Second UAR received on LUN reset TUR while UAR already in-progress aborts causing lost IO SRB
Description Of Change: Added a check when kicking-of the UAR state machine to handle a new UAR to verify its current state. If the state machine is already running, the UAR is rejected and not handled. This allows the current UAR to finish completion and avoids losing the associated SRB, which allows the system to recover cleanly.
Issue Description: When a UAR is detected on an IO reply from a LUN the driver starts the UAR state machine to handle the UAR. If StorPort sends a LUN reset to the same LUN while the UAR state machine is running, and the LUN reset TUR also comes back with a UAR, the driver will restart the UAR state machine from the beginning abruptly. This results in the original IO SRB getting lost and never completed. This in-turn causes StorPort to enter into an infinite LUN reset loop on that LUN, and all IOS will be blocked.
Steps To Reproduce: Run cable breaker every 5 seconds while also running a heavy IO load across topology with lots of disks (55+).
May require an hour or little longer to reproduce.

Defect ID: SCGCQ01058006
Headline: Ventura Windows Driver: System crashes while running IOs on storage pools along with TMs
Description Of Change: Replaced all polling of the ISR when running under ISR context with a delay to allow each ISR thread time to process its own IOs.
Issue Description: Configuration details:-

OS:- Windows server 2016 TP4
HBA:- Ventura
FW:- 00.250.65.00
Driver:- 2.60.16.00

Server -> HBA (ventura) --> JBOD attached NVME drive
--> JBOD attached NVME drive
--> cobra expander --> Super micro 12G Enclosure
--> Dataon 12G Enclosure

>>Ventura HBA was flashed with latest FW (00.250.65.00) and installed in the system.
>>Installed supported Driver in the OS (2.60.16.00).
>>Discovered all the drives connected to the HBA in the OS.
>>created Storage pools from Windows server manager.
>>Created four storage pools (storage pool1 --> sas drives (Virtual disk1- 6TB), Storage pool2 --> sas drives (virtual disk2 - 3TB), Storage pool3 --> sas drives (virtual disk3 - 5TB), Storage pool4 --> NVME drives - (virtual disk4 - 750GB))
>>Created NTFS File systems on all the four Virtual disks from windows Disk management.
>> started IOs on all the File systems using IOMETER tool. (Random IOs reads & writes)
>>In the background started triggering TMs (target resets, lun resets, & diag resets) Using a SAL script.
>>After 10 mins of the Test, system crashed with "DPC_Timeout" and rebooted.

Steps To Reproduce: Configuration details:-

OS:- Windows server 2016 TP4
HBA:- Ventura
FW:- 00.250.65.00
Driver:- 2.60.16.00

Server -> HBA (ventura) --> Enclosure attached NVME drive
--> Enclosure attached NVME drive
--> cobra expander --> Super micro 12G Enclosure
--> Dataon 12G enclosure

>>Ventura HBA was flashed with latest FW (00.250.65.00) and installed in the system.
>>Installed supported Driver in the OS (2.60.16.00).
>>Discovered all the drives connected to the HBA in the OS.
>>created Storage pools from Windows server manager.
>>Created four storage pools (storage pool1 --> sas drives (Virtual disk1- 6TB), Storage pool2 --> sas drives (virtual disk2 - 3TB), Storage pool3 --> sas drives (virtual disk3 - 5TB), Storage pool4 --> NVME drives - (virtual disk4 - 750GB))
>>Created File systems on all the four Virtual disks from windows Disk management.
>> started IOs on all the File systems using IOMETER tool.
>>In the background started triggering TMs (target resets, lun resets, & diag resets)
>>After 10 mins of the Test, system crashed with "DPC_Timeout" and rebooted.

Defect ID: SCGCQ01145845
Headline: Device Driver INF Verification Test (Certification) fails with error on all OSs Win10 client/server HLK
Description Of Change: Both Production and Inbox INF files are modified .
Issue Description: Windows INF test Device Driver INF Verification Test (Certification) fails with error on all OSs Win10 client/server HLK
UserText=The inf C:\Windows\Inf\oem0.inf had an error at line 357.
This line is missing the double quote (&quot;) at the end of the line. Adding one to continue processing.CA="8651479" LA=8651604

Error gets filtered on earlier OS's below win 8.1 but expires in October.
Error does not occurs with phase 9 driver used for Cutlass installs.

Steps To Reproduce: Run the INF support test in an HLK environment on win 10, win server 2016

Windows Driver Enhancements Implemented (1):

EnhancementRequest ID: SCGCQ01108849
Headline: Add PCI IDs for NGM HBA330 Controllers
Description Of Change: Both Production and Inbox INF files have been updated.

Alpha Release (SCGCQ00179278) - SAS2PARSER - 09.00.00.00

^

ReleaseOrderID: **SCGCQ00179278** **Open In CQWeb** **Download**
Headline: **SAS2 NVDATA Parser Tool Version 09.00.00.00**
Release Version: **09.00.00.00**
Product Org: **SCS**
Product Gen: **SAS2**
Product Family: **SAS2_Applications**
UCM Project: **SAS2PARSER**
Release Type: **Alpha**
UCM Stream: **SAS2PARSER_Rel**
Owner: **sbakthav**
Release Baseline: **SAS2PARSER_2_18_2011_v9.00.00.00_REL@\SAS2**
Release Date / Time: **18-FEB-11**
Release Type: **Alpha**