VSI OpenVMS x86-64 V9.2-2 Update V2

Publication Date: May 2024

Operating System: VSI OpenVMS x86-64 V9.2-2

Kit Name: VMS922X_UPDATE-V0200

VMS Software, Inc. (VSI)
Boston, Massachusetts, USA

Table of Contents

1. Kit Name 7
2. Kit Description 7
2.1. Installation Rating 7
2.2. Reboot Requirement 7
2.3. Version(s) of VSI OpenVMS to Which This Kit May Be Applied 8
2.4. Target Disk Requirements 8
3. Kits Superseded by This Kit 8
4. Kit Dependencies 9
5. Problems Addressed in This Kit 9
5.1. SCSI Disk May Enter Mount Verification When Copying Very Large Files 9
5.1.1. Problem Description 9
5.1.2. Images and/or Files Affected 9
5.1.3. VSI Case Identifier 9
5.1.4. Release Version of VSI OpenVMS That Will Contain This Change 9
5.2. SATA Disk May Enter Mount Verification When Copying Very Large Files 10
5.2.1. Problem Description 10
5.2.2. Images and/or Files Affected 10
5.2.3. VSI Case Identifier 10
5.2.4. Release Version of VSI OpenVMS That Will Contain This Change 10
5.3. Increase the MPW_WAITLIMIT System Parameter Value From AUTOGEN 10
5.3.1. Problem Description 10
5.3.2. Images and/or Files Affected 10
5.3.3. VSI Case Identifier 10
5.3.4. Release Version of VSI OpenVMS That Will Contain This Change 10
5.4. Determine if System Parameters Will Exhaust 32-bit Address Space 11
5.4.1. Problem Description 11
5.4.2. Images and/or Files Affected 11
5.4.3. VSI Case Identifier 11
5.4.4. Release Version of VSI OpenVMS That Will Contain This Change 11
5.4.5. Workaround 11
5.5. Change How AUTOGEN Computes the SYSMWCNT System Parameter 12
5.5.1. Problem Description 12
5.5.2. Images and/or Files Affected 12
5.5.3. VSI Case Identifier 12
5.5.4. Release Version of VSI OpenVMS That Will Contain This Change 12
5.5.5. Workaround 12
5.6. Avoid Boot Failures Due to the Error Log Buffer System Parameters 12
5.6.1. Problem Description 12
5.6.2. Images and/or Files Affected 12
5.6.3. VSI Case Identifier 13
5.6.4. Release Version of VSI OpenVMS That Will Contain This Change 13
5.6.5. Workaround 13
5.7. AUTHORIZE Returns a Warning Instead of a Success Status 13
5.7.1. Problem Description 13
5.7.2. Images and/or Files Affected 14
5.7.3. VSI Case Identifier 14
5.7.4. Release Version of VSI OpenVMS That Will Contain This Change 14
5.8. Exception Handling Could Produce Access Violation or Hang 14
5.8.1. Problem Description 14
5.8.2. Images and/or Files Affected 14

5.8.3. VSI Case Identifier 14
5.8.4. Release Version of VSI OpenVMS That Will Contain This Change 14
5.9. XFC SDA Extension May Not Display Header for SHOW FILE 15
5.9.1. Problem Description 15
5.9.2. Images and/or Files Affected 15
5.9.3. VSI Case Identifier 15
5.9.4. Release Version of VSI OpenVMS That Will Contain This Change 15
5.10. UNXSIGNAL System Crash When Using Magtape 15
5.10.1. Problem Description 15
5.10.2. Images and/or Files Affected 15
5.10.3. VSI Case Identifier 15
5.10.4. Release Version of VSI OpenVMS That Will Contain This Change 15
5.11. Importing an Exported VMware VM May Result in Missing LAN Devices 16
5.11.1. Problem Description 16
5.11.2. Images and/or Files Affected 16
5.11.3. VSI Case Identifier 16
5.11.4. Release Version of VSI OpenVMS That Will Contain This Change 16
5.11.5. Workaround 16
5.12. DIRECTORY With File Size Selection May Not Display Files Over 1TB 16
5.12.1. Problem Description 16
5.12.2. Images and/or Files Affected 17
5.12.3. VSI Case Identifier 17
5.12.4. Release Version of VSI OpenVMS That Will Contain This Change 17
5.13. SET SHADOW or SHOW SHADOW Command May Cause System Crash 17
5.13.1. Problem Description 17
5.13.2. Images and/or Files Affected 17
5.13.3. VSI Case Identifier 17
5.13.4. Release Version of VSI OpenVMS That Will Contain This Change 17
5.14. Adding Duplicate Network Proxy Entity Causes Security Server Failure 17
5.14.1. Problem Description 17
5.14.2. Images and/or Files Affected 18
5.14.3. VSI Case Identifier 18
5.14.4. Release Version of VSI OpenVMS That Will Contain This Change 18
5.15. LIB$X86_ALLOC_BPV Allocates Insufficient Memory 18
5.15.1. Problem Description 18
5.15.2. Images and/or Files Affected 18
5.15.3. VSI Case Identifier 18
5.15.4. Release Version of VSI OpenVMS That Will Contain This Change 18
5.16. Errors in VAX Format Complex Floating Point Math Functions 18
5.16.1. Problem Description 18
5.16.2. Images and/or Files Affected 19
5.16.3. VSI Case Identifier 19
5.16.4. Release Version of VSI OpenVMS That Will Contain This Change 19
5.17. Floating Point Condition Handler Correction for COBOL 19
5.17.1. Problem Description 19
5.17.2. Images and/or Files Affected 19
5.17.3. VSI Case Identifier 19
5.17.4. Release Version of VSI OpenVMS That Will Contain This Change 19
5.18. Return the Correct VAX-Style Condition Status for BASIC and COBOL 19
5.18.1. Problem Description 19
5.18.2. Images and/or Files Affected 19
5.18.3. VSI Case Identifier 20

5.18.4. Release Version of VSI OpenVMS That Will Contain This Change 20
5.19. VAX/IEEE Conversion Routines May Corrupt a Register 20
5.19.1. Problem Description 20
5.19.2. Images and/or Files Affected 20
5.19.3. VSI Case Identifier 20
5.19.4. Release Version of VSI OpenVMS That Will Contain This Change 20
5.20. Fix Memory Leak in SYS$GOTO_UNWIND_64 20
5.20.1. Problem Description 20
5.20.2. Images and/or Files Affected 20
5.20.3. VSI Case Identifier 20
5.20.4. Release Version of VSI OpenVMS That Will Contain This Change 21
5.21. Correct Issues With SYS$UNWIND on Error or Abort 21
5.21.1. Problem Description 21
5.21.2. Images and/or Files Affected 21
5.21.3. VSI Case Identifier 21
5.21.4. Release Version of VSI OpenVMS That Will Contain This Change 21
6. Problems Addressed from Previous Kits 21
6.1. System May Crash With DELCONPFN Bugcheck 21
6.1.1. Problem Description 21
6.1.2. Images and/or Files Affected 21
6.1.3. VSI Case Identifier 22
6.1.4. Release Version of VSI OpenVMS That Will Contain This Change 22
6.2. SYS$IO_PERFORM May Return Incorrect Data or Corrupt Files 22
6.2.1. Problem Description 22
6.2.2. Images and/or Files Affected 22
6.2.3. VSI Case Identifier 22
6.2.4. Release Version of VSI OpenVMS That Will Contain This Change 22
6.3. Disallow Shadow Sets as Dump Off System Disk (DOSD) Devices 22
6.3.1. Problem Description 22
6.3.2. Images and/or Files Affected 22
6.3.3. VSI Case Identifier 23
6.3.4. Release Version of VSI OpenVMS That Will Contain This Change 23
6.3.5. Workaround 23
6.4. CLUE$STARTUP Exhausts System Disk Space if no DOSD Devices Are Mounted 23
6.4.1. Problem Description 23
6.4.2. Images and/or Files Affected 23
6.4.3. VSI Case Identifier 23
6.4.4. Release Version of VSI OpenVMS That Will Contain This Change 23
6.4.5. Workaround 23
6.5. Protected Shareable Image Issue Could Cause Process or System Crash 23
6.5.1. Problem Description 23
6.5.2. Images and/or Files Affected 24
6.5.3. VSI Case Identifier 24
6.5.4. Release Version of VSI OpenVMS That Will Contain This Change 24
6.5.5. Workaround 24
6.6. RUNOFF Command May Display XPORT Errors or Corrupted Filenames 24
6.6.1. Problem Description 24
6.6.2. Images and/or Files Affected 25
6.6.3. VSI Case Identifier 25
6.6.4. Release Version of VSI OpenVMS That Will Contain This Change 25
6.7. Various Corrections to OpenVMS Debugger Behavior 25
6.7.1. Problem Description 25

6.7.2. Images and/or Files Affected 25
6.7.3. VSI Case Identifier 25
6.7.4. Release Version of VSI OpenVMS That Will Contain This Change 25
6.8. ACMS Failures From Image Activation Issues 26
6.8.1. Problem Description 26
6.8.2. Images and/or Files Affected 26
6.8.3. VSI Case Identifier 26
6.8.4. Release Version of VSI OpenVMS That Will Contain This Change 26
6.9. Misconfigured LAN Failover Devices or VLAN Devices for Cluster Boot 26
6.9.1. Problem Description 26
6.9.2. Images and/or Files Affected 26
6.9.3. VSI Case Identifier 26
6.9.4. Release Version of VSI OpenVMS That Will Contain This Change 27
6.9.5. Workaround 27
6.10. XA Gateway Failures in XaOpen() and IVP 27
6.10.1. Problem Description 27
6.10.2. Images and/or Files Affected 27
6.10.3. VSI Case Identifier 27
6.10.4. Release Version of VSI OpenVMS That Will Contain This Change 27
6.11. Add Support for Packed Decimal Routines Needed for BASIC 27
6.11.1. Problem Description 27
6.11.2. Images and/or Files Affected 28
6.11.3. VSI Case Identifier 28
6.11.4. Release Version of VSI OpenVMS That Will Contain This Change 28
6.12. Availability Manager (DECamds) Could Induce INCONSTATE Bugcheck 28
6.12.1. Problem Description 28
6.12.2. Images and/or Files Affected 28
6.12.3. VSI Case Identifier 28
6.12.4. Release Version of VSI OpenVMS That Will Contain This Change 28
6.13. Access Violation in DTI$SHARE May Cause Process Deletion 29
6.13.1. Problem Description 29
6.13.2. Images and/or Files Affected 29
6.13.3. VSI Case Identifier 29
6.13.4. Release Version of VSI OpenVMS That Will Contain This Change 29
6.13.5. Workaround 29
6.14. Correct Spurious Exception Error Messages During Traceback Processing 29
6.14.1. Problem Description 29
6.14.2. Images and/or Files Affected 29
6.14.3. VSI Case Identifier 29
6.14.4. Release Version of VSI OpenVMS That Will Contain This Change 30
6.15. Add Support for x86 to BASIC Run-Time Library 30
6.15.1. Problem Description 30
6.15.2. Images and/or Files Affected 30
6.15.3. VSI Case Identifier 30
6.15.4. Release Version of VSI OpenVMS That Will Contain This Change 30
6.16. DCL Procedure Error DOUBLEDEALO Bugcheck and Process Deletion 30
6.16.1. Problem Description 30
6.16.2. Images and/or Files Affected 30
6.16.3. VSI Case Identifier 30
6.16.4. Release Version of VSI OpenVMS That Will Contain This Change 30
6.17. Executive Mode Bugcheck May Cause KRNLSTAKNV System Crash 31
6.17.1. Problem Description 31

6.17.2. Images and/or Files Affected 31
6.17.3. VSI Case Identifier 31
6.17.4. Release Version of VSI OpenVMS That Will Contain This Change 31
6.18. Remove Spurious Linker Warnings 31
6.18.1. Problem Description 31
6.18.2. Images and/or Files Affected 31
6.18.3. VSI Case Identifier 32
6.18.4. Release Version of VSI OpenVMS That Will Contain This Change 32
6.19. Improve SECURITY_SERVER Behavior for Systems Under Persistent Attack 32
6.19.1. Problem Description 32
6.19.2. Images and/or Files Affected 32
6.19.3. VSI Case Identifier 32
6.19.4. Release Version of VSI OpenVMS That Will Contain This Change 32
6.20. Various Fixes to SECURITY_SERVER Behavior 32
6.20.1. Problem Description 32
6.20.2. Images and/or Files Affected 33
6.20.3. VSI Case Identifier 33
6.20.4. Release Version of VSI OpenVMS That Will Contain This Change 33
6.21. Some Exceptions May Mistakenly Crash the System 33
6.21.1. Problem Description 33
6.21.2. Images and/or Files Affected 33
6.21.3. VSI Case Identifier 33
6.21.4. Release Version of VSI OpenVMS That Will Contain This Change 33
7. Images or Files Replaced 33
8. Installation Instructions 48
8.1. Compressed File 48
8.2. Installation Command 48
8.3. Special Installation Instructions 48
9. Copyright 49
10. Disclaimer of Warranty and Limitation of Liability 49
11. Patch ID 50
Appendix A. User-Selectable Control Options and Scripting Considerations 50
A.1. Controlling Kit Behavior for Introductory Questions 50
A.2. Standard Behavior for YES/NO Questions Asked During Kit Installation 51
A.3. Installing a Kit From a Batch Job 51
A.4. Deprecated Logical Names From HPE Patch Kits 52

Read This Before Installing the Kit

There is an important issue with installing the VSI OpenVMS x86-64 V9.2-2 Update V2.0 patch kit, that was discovered after releasing this kit.

If SYS$SHARE:SDA$SHARE.EXE is installed as a known image on your system (possibly with privileges to allow access to ANALYZE/SYSTEM or ANALYZE/CRASH for non-privileged users) when the kit has finished updating the system disk and is running SYS$MD before rebooting, the following errors will occur, and the disk will become unusable:

```
%SYSTEM-W-NOSUCHFILE, no such file
 \VMSSYS$DKA0:[SYS0.SYSCOMMON.SYSLIB]SDA$SHARE.EXE\
Unable to copy file SDA$SHARE.EXE and/or modify its ownership
An unexpected error has occurred. Please contact VSI Support for assistance.
 Cleaning up...
```

To prevent this from happening, you must remove SYS$SHARE:SDA$SHARE.EXE from the list of installed known images *before* installing this patch kit.

To determine whether SYS$SHARE:SDA$SHARE.EXE is installed as a known image on your system, use the following command:

```
$ INSTALL LIST SYS$SHARE:SDA$SHARE.EXE
```

If the image is displayed after entering the command above, use the following command:

```
$ INSTALL REMOVE SYS$SHARE:SDA$SHARE.EXE
```

If you are unsure whether you should perform the procedure described above, VMS Software recommends that you perform it anyway, because even if SYS$SHARE:SDA$SHARE.EXE is not a known installed image on your system, using the **`INSTALL REMOVE`** command on it will be completely harmless and will just display a corresponding warning message.

1. Kit Name

`VMS922X_UPDATE-V0200`

2. Kit Description

2.1. Installation Rating

INSTALL_1: To be installed by all customers.

This installation rating serves as a guide to which customers should apply this remedial kit.

Reference the attached Disclaimer of Warranty and Limitation of Liability Statement.

2.2. Reboot Requirement

A reboot is mandated as part of installing this kit, performed automatically following the kit installation.

VSI OpenVMS for the x86 architecture uses a memory disk image, stored on the system disk, when booting the system. The content of the memory disk must remain consistent with the system disk content.

This kit updates the memory disk image and invokes a system reboot sequence (shutdown with reboot) directly as part of the kit installation. You must be prepared to allow the system reboot when installing the kit. After all other kit actions are complete, the system will automatically shutdown and reboot.

If you allow the reboot, you will have the choice of whether to invoke the site-specific shutdown procedure, SYS$MANAGER:SYSHUTDWN.COM during the shutdown portion of the reboot.

By default, after installation completes, the minutes until shutdown is zero. If you wish to leave additional time before the shutdown begins, define the system logical name SHUTDOWN $MINIMUM_MINUTES as the integer value of the wait time in minutes. For example:

```
DEFINE/SYSTEM SHUTDOWN$MINIMUM_MINUTES 10
```

No other options for the shutdown may be specified.

The shutdown will commence directly after the memory disk update as the final portion of the kit installation.

2.3. Version(s) of VSI OpenVMS to Which This Kit May Be Applied

VSI OpenVMS x86-64 V9.2-2

2.4. Target Disk Requirements

This kit will create a new memory disk image file during **PRODUCT INSTALL** or **PRODUCT UNDO PATCH** operations. This file is required for x86 systems bootstrap. If this file cannot be correctly created, the target disk will not be bootable and you will need to restore from backup.

During **PRODUCT INSTALL**, the kit will check for sufficient space on the target disk. The minimum free space is 600,000 blocks to ensure that the memory disk image file and kit files can safely fit on the disk.

Additionally, a check is made to determine if the disk is too fragmented to correctly create the memory disk file. If either check fails, the installation will be aborted before making any changes. After you take corrective actions the **PRODUCT INSTALL** can be re-attempted.

These same checks are not automatically handled by the kit before a **PRODUCT UNDO PATCH** operation. This is not a frequent operation for customer systems. Should you need to remove this kit, you should ensure sufficient disk space before you start. The same 600,000 block minimum applies. To check fragmentation requirements, you can use the following commands:

```
$ ANALYZE/DISK/EXTENTS/REQUIRED=200000/NOOUTPUT disk
$ SHOW SYMBOL ANALYZE$REQUIRED_EXTENTS
```

If the symbol value is 25 or more, the disk is too fragmented and you should defragment it before using **PRODUCT UNDO PATCH**. You can accomplish this using a defragmentation tool or by restoring an image backup.

3. Kits Superseded by This Kit

VMS922X_UPDATE-V0100	

VMS922X_HOTFIX-V0300	This HOTFIX kit contained the corrected images for the problems described in Sections 6.1 and 6.2 below. It was only distributed to the customer reporting the issues.
VMS922X_HOTFIX-V0400	This HOTFIX kit contained the corrected image for the problem described in Section 5.1 below. It was only distributed to the customer reporting the issue.

4. Kit Dependencies

VMS922X_PCSI-V0100

Note

The VMS922X_PCSI-V0100 kit must be installed prior to installing this kit, using a separate **`PRODUCT INSTALL`** command. The kits may not be jointly installed with a single PRODUCT INSTALL operation.

5. Problems Addressed in This Kit

5.1. SCSI Disk May Enter Mount Verification When Copying Very Large Files

5.1.1. Problem Description

Copying very large disk files (4GB or larger) on VSI OpenVMS x86-64 may lead to the disk entering mount verification. Depending on the circumstances, the verification may never complete and the disk may suffer a mount verification timeout. Such a case would necessitate a **`DISMOUNT/ABORT`** or a reboot to recover access to the disk.

The disk controller requires that buffers used for transfers do not cross any 4GB boundary. When creating the scatter-gather map to describe the memory buffers to use, the disk driver incorrectly calculated the byte counts for the pieces which were split over such a boundary. The resulting error returned from the disk would trigger the mount verification event.

5.1.2. Images and/or Files Affected

[SYS$LDR]SYS$PKMDRIVER.EXE

5.1.3. VSI Case Identifier

Jiras SPS-1427, DRIV-404

5.1.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.2. SATA Disk May Enter Mount Verification When Copying Very Large Files

5.2.1. Problem Description

Copying very large disk files (4GB or larger) on VSI OpenVMS x86-64 may lead to the disk entering mount verification. Depending on the circumstances, the verification may never complete and the disk may suffer a mount verification timeout. Such a case would necessitate a **DISMOUNT/ABORT** or a reboot to recover access to the disk.

This problem is very similar to that described above in Section 5.1. In this case for SATA disks, the driver correctly handled transfers which crossed a 4GB boundary. However, it incorrectly created an additional scatter-gather list entry for a segment of a buffer that was immediately before a 4GB boundary but did not cross the 4GB boundary. The resulting error returned from the disk would trigger the mount verification event.

5.2.2. Images and/or Files Affected

[SYS$LDR]SYS$PKDDRIVER.EXE

5.2.3. VSI Case Identifier

Jira DRIV-371

5.2.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.3. Increase the MPW_WAITLIMIT System Parameter Value From AUTOGEN

5.3.1. Problem Description

The AUTOGEN computation of the MPW_WAITLIMIT system parameter now calculates an increased value. A larger value will reduce the possibility of processes being placed in a resource wait state (RWMPB) when the modified page list must be flushed to the page file.

5.3.2. Images and/or Files Affected

[SYSUPD]AUTOGEN.COM

5.3.3. VSI Case Identifier

Jira BO-1417

5.3.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.4. Determine if System Parameters Will Exhaust 32-bit Address Space

5.4.1. Problem Description

There are a number of system parameters that control allocation of system structures which must reside in 32-bit S0/S1 address space. The total of S0/S1 address space cannot exceed 2GB.

If the total of these S0/S1 parameters should exceed 2GB of address space, that would exhaust the available 32-bit memory and the system would fail to boot.

AUTOGEN will now recognize and report an error if the relevant system parameters will exhaust the available 32-bit space. The computation is not exact as there are some aspects of memory usage that AUTOGEN cannot determine.

If AUTOGEN calculates that the system 32-bit address space will be exhausted using the current parameter values, an error message and additional informational messages are generated to provide guidance for the system manager to adjust them using MODPARAMS.DAT.

AUTOGEN will provide a list of the parameters and their sizes when reporting the message. An example of the error message and presented data is:

```
%AUTOGEN-E-EXH_S0S1 - The generated parameters will exhaust S0S1 space.

The generated parameters will result in a system that will not boot.
The following parameters consume 2399.85MB of S0S1 space of which there
is only 2GB (2048MB) available.  You should review records pertaining to
the following parameters in MODPARAMS.DAT:

      PAGEDYN            579724288 Bytes    552.87 MB
      NPAGEVIR          1879048192 Bytes   1792.00 MB
      GH_EXEC_DATA            2048 Pages     16.00 MB
      GH_RES_CODE             3072 Pages     24.00 MB
      GH_RES_DATA              512 Pages      4.00 MB
      GH_RO_EXEC_S0           1024 Pages      8.00 MB
      GH_RO_RES_S0            1024 Pages      8.00 MB
```

5.4.2. Images and/or Files Affected

[SYSUPD]AUTOGEN.COM

5.4.3. VSI Case Identifier

Jira UT-220

5.4.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.4.5. Workaround

Update MODPARAMS.DAT to reduce the size of the parameters which contribute to the 32-bit address space.

5.5. Change How AUTOGEN Computes the SYSMWCNT System Parameter

5.5.1. Problem Description

The SYSMWCNT system parameter specifies the size of the pageable portion of the system working set. On X86, none of system space is pageable.

If a large value for this parameter is computed by AUTOGEN and used during system operation, it could impact the ability of the system to generate a valid system crash dump file during a system crash.

AUTOGEN will no longer calculate a large value for the SYSMWCNT parameter.

The next release of VSI OpenVMS x86-64 will remove references to SYSMWCNT from the system and make the SYSMWCNT system parameter obsolete.

5.5.2. Images and/or Files Affected

[SYSUPD]AUTOGEN.COM

5.5.3. VSI Case Identifier

Jira BO-1527

5.5.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.5.5. Workaround

Limit the size of the SYSMWCNT system parameter by adding this line to MODPARAMS.DAT:

```
MAX_SYSMWCNT=8192
```

5.6. Avoid Boot Failures Due to the Error Log Buffer System Parameters

5.6.1. Problem Description

The system will fail to boot if the error log system parameters (ERRORLOGBUFFERS and ERLBUFFERPAGES) are larger than the S2 error log system parameters (ERRORLOGBUFF_S2 and ERLBUFFERPAG_S2).

AUTOGEN has been updated to insure the generated ERRORLOGBUFF_S2 and ERLBUFFERPAG_S2 parameters are at least as large as the ERRORLOGBUFFERS and ERLBUFFERPAGES parameters.

The next release of VSI OpenVMS x86-64 will correct the issue and allow the system to successfully boot regardless of any differences in these parameters.

5.6.2. Images and/or Files Affected

[SYSUPD]AUTOGEN.COM

5.6.3. VSI Case Identifier

Jiras SPS-1419, BO-1542

5.6.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.6.5. Workaround

If the X86 boot fails with the following error:

```
%SYSBOOT-F-GHREGIONFULL, An allocation was attempted that exceeded
                         the space in the EXEC S2 region.
This is a fatal error in source code.  Please contact VSI support.
Error: Slice allocation S2 Nonpaged pool status: 00000124 for 0x0 bytes
 halting ...
```

1. Boot conversationally to stop in SYSBOOT and do a **SHOW ER*** command.

```
SYSBOOT>  SHOW ER*
Parameter Name            Current    Default     Min.       Max.   Unit
 Dynamic
--------------            -------    -------   -------    -------  ----
 -------
ERRORLOGBUFFERS                20          4         2         64 Buffers
ERLBUFFERPAGES                 16          8         2        128 Pagelets
ERLBUFFERPAG_S2                 8          8         2         -1 Pagelets
ERRORLOGBUFF_S2                 6          4         2         -1 Buffers
```

2. Then set the ERLBUFFERPAG_S2 to match the value for ERLBUFFERPAGES and set ERRORLOGBUFF_S2 to match the value for ERRORLOGBUFFERS:

```
SYSBOOT> SET ERLBUFFERPAG_S2 16
SYSBOOT> SET ERRORLOGBUFF_S2 20
```

3. After the system boots, add the following lines to your MODPARAMS.DAT file to insure AUTOGEN computes the larger S2 error log parameters:

```
MIN_ERLBUFFERPAG_S2=16
MIN_ERRORLOGBUFF_S2=32
```

5.7. AUTHORIZE Returns a Warning Instead of a Success Status

5.7.1. Problem Description

When invoked as a foreign command to execute a single operation, the LIST command in AUTHORIZE produces the desired output file but returns an incorrect status code to DCL in the $STATUS symbol.

For example:

```
$ mcr authorize list/brief SYSTEM
$ show symbol $status
$STATUS == "%X10002EE8"
```

```
$ write sys$output f$message(%X0002EE8)
%SYSTEM-W-INVKTLIM, invalid kernel thread limit specified
```

This patch kit corrects the behavior so that the normal success status code is returned as expected.

5.7.2. Images and/or Files Affected

[SYSEXE]AUTHORIZE.EXE

5.7.3. VSI Case Identifier

Jiras SPS-1436, UT-269

5.7.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.8. Exception Handling Could Produce Access Violation or Hang

5.8.1. Problem Description

Exception handling could not always find the target context for a SYS$GOTO_UNWIND_64 operation. This could result in calling the Pascal RTL condition handler an additional time, which eventually caused either an access violation (ACCVIO) or program hang.

This symptom occurred in a Pascal program which used a GOTO statement.

Other exception handler usages within OpenVMS which could experience the problem have also been corrected.

5.8.2. Images and/or Files Affected

[SYS$LDR]EXCEPTION.EXE
[SYS$LDR]EXCEPTION.STB
[SYS$LDR]EXCEPTION_MON.EXE
[SYS$LDR]EXCEPTION_MON.STB
[SYSEXE]LANACP_BFS.EXE
[SYSEXE]DBGHK$PRCDUMP_KERNEL.EXE
[SYSLIB]IMGDMP.EXE
[SYSLIB]LIBRTL.EXE
[SYSLIB]LIBRTL.STB
[SYSLIB]SDA$SHARE.EXE

5.8.3. VSI Case Identifier

Jiras SPS-1347, RTLS-425

5.8.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.9. XFC SDA Extension May Not Display Header for SHOW FILE

5.9.1. Problem Description

The XFC extension for SDA may not display the header for some files displayed by a SHOW FILE command, such as:

```
SDA> XFC SHOW FILE /BRIEF /OPEN
```

The routine used to display the file information did not always detect switching to a different disk volume and skipped the header information.

This has been corrected with this patch kit.

5.9.2. Images and/or Files Affected

[SYSLIB]XFC$SDA.EXE

5.9.3. VSI Case Identifier

Jira BO-1538
Netsuite NS6238

5.9.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.10. UNXSIGNAL System Crash When Using Magtape

5.10.1. Problem Description

When a magtape is mounted in ANSI mode (not foreign), the magtape ACP (MTAAACP) controls file access on the tape.

The layout of structures used in MTAAACP is different on X86 due to compiler changes from previous architectures. In some cases this caused a timing issue where some transient data was not accessible. This produced an access violation which resulted in an UNXSIGNAL bugcheck and system crash.

The problem is corrected with this patch kit.

5.10.2. Images and/or Files Affected

[SYSEXE]MTAAACP.EXE

5.10.3. VSI Case Identifier

Jira BO-1598

5.10.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.11. Importing an Exported VMware VM May Result in Missing LAN Devices

5.11.1. Problem Description

An export/import of a VMware VM may change the LAN device adapter type from PCI to PCIE. The device configuration file, SYS$SYSTEM:SYS$CONFIG.DAT, allowed only PCI devices.

This has been corrected by duplicating the VMware LAN device entries and marking one of them PCI and the other PCIE. This configures the LAN devices regardless of the adapter type.

A side effect of this update is that if the e1000e network type is selected when configuring a network device, rather than no device being configured when the VM is booted, the EI device will be configured and an error message is displayed:

```
Unsupported network interface card, device set offline
```

The system manager can then edit the VM configuration and select the e1000. Note that the e1000e device will be supported in the next release of VSI OpenVMS x86-64.

5.11.2. Images and/or Files Affected

[SYSEXE]SYS$CONFIG.DAT

5.11.3. VSI Case Identifier

Jiras SPS-1447, BO-1569

5.11.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.11.5. Workaround

Edit SYS$SYSTEM:SYS$CONFIG.DAT to duplicate the VMware LAN device entries (search for VMware in SYS$CONFIG.DAT) and change the adapter type such that one entry is PCI and the other is PCIE.

5.12. DIRECTORY With File Size Selection May Not Display Files Over 1TB

5.12.1. Problem Description

A DIRECTORY command with a file size selection qualifier may not display files larger than 1TB. For example, the command:

```
$ DIRECTORY/SELECT=SIZE=MINIMUM=1
```

would not display any files in the directory larger than 1TB.

This problem was due to the selection routine incorrectly interpreting the file size as a signed integer value.

The problem is corrected with this patch kit.

5.12.2. Images and/or Files Affected

[SYSEXE]DIRECTORY.EXE

5.12.3. VSI Case Identifier

Jira BO-1562

5.12.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.13. SET SHADOW or SHOW SHADOW Command May Cause System Crash

5.13.1. Problem Description

The SETSHOSHADOW image provides the interface for the **SET SHADOW** and **SHOW SHADOW** commands to modify behavior or display information from the shadowing device driver.

Timing issues may arise when these commands are issued before the relevant driver structures are completely initialized while a shadow set is mounted. This would typically result in an access violation and subsequent system crash.

A synchronization mechanism has been added to prevent data being misinterpreted before the mount is complete.

5.13.2. Images and/or Files Affected

[SYSEXE]SETSHOSHADOW.EXE

5.13.3. VSI Case Identifier

Jiras BO-1575, DRIV-188

5.13.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.14. Adding Duplicate Network Proxy Entity Causes Security Server Failure

5.14.1. Problem Description

Adding a duplicate network proxy (for example, for having one with **/DEFAULT** and one without) results in the security server failing and issuing timeout warnings. The only recovery would be to restart the security server.

This problem arose from incorrectly accounting for the new local user in the proxy record.

The problem is corrected with this patch kit.

5.14.2. Images and/or Files Affected

[SYSEXE]SECURITY_SERVER.EXE

5.14.3. VSI Case Identifier

Jiras BO-1531, BO-1584

5.14.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.15. LIB$X86_ALLOC_BPV Allocates Insufficient Memory

5.15.1. Problem Description

The library routine LIB$X86_ALLOC_BPV is used to allocate a bound procedure value memory area. The conversion to pagelets did not correctly account for the pagesize on X86, resulting in much less memory allocated than was requested. Subsequent access to the memory would either fail or generate an access violation.

This problem is corrected with this patch kit.

5.15.2. Images and/or Files Affected

[SYSLIB]LIBRTL.EXE
[SYSLIB]LIBRTL.STB

5.15.3. VSI Case Identifier

Jira RTLS-429

5.15.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.16. Errors in VAX Format Complex Floating Point Math Functions

5.16.1. Problem Description

The VAX format complex floating point math functions return an incorrect or invalid value for the real portion of the result.

The CVT_IEEE_COMPLEX_TO_VAX macro to convert to VAX format contained a typo for the real portion description.

This problem is corrected with this patch kit.

5.16.2. Images and/or Files Affected

[SYSLIB]DPML$SHR.EXE

5.16.3. VSI Case Identifier

Jira RTLS-431

5.16.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.17. Floating Point Condition Handler Correction for COBOL

5.17.1. Problem Description

Due to corrections included in V9.2-2, the COBOL RTL needs to readjust code that deals with floating divide-by-zero and invalid floating traps. Without the change, programs that expect to see such errors may get incorrect output or tracebacks.

Correctly working programs are not impacted.

The problem is corrected with this patch kit.

5.17.2. Images and/or Files Affected

[SYSLIB]DEC$COBRTL.EXE

5.17.3. VSI Case Identifier

Jira DEV-2034

5.17.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.18. Return the Correct VAX-Style Condition Status for BASIC and COBOL

5.18.1. Problem Description

The routines OTS$ADDP, OTS$ASHP, and OTS$CMPP were rewritten for the V9.2-2 Update V1.0 patch kit to resolve BASIC and COBOL issues.

Further improvements to these routines now provide the expected VAX-style condition status for the functions being performed.

5.18.2. Images and/or Files Affected

[SYSLIB]LIBOTS2.EXE

5.18.3. VSI Case Identifier

Jira RTLS-426

5.18.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.19. VAX/IEEE Conversion Routines May Corrupt a Register

5.19.1. Problem Description

The VAX/IEEE conversion routines in LIBOTS.EXE incorrectly modified a preserved register without saving and restoring it. This could lead to incorrect results and unknown effect to the calling routine.

The code has been modified to use a non-preserved temporary register for the affected operations.

5.19.2. Images and/or Files Affected

[SYSLIB]LIBOTS.EXE
[SYSLIB]LIBOTS.STB

5.19.3. VSI Case Identifier

Jira DEV-2043

5.19.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.20. Fix Memory Leak in SYS$GOTO_UNWIND_64

5.20.1. Problem Description

When a condition handler performed a goto unwind operation, some context from the original condition was not deallocated before execution continued at the target of the goto unwind.

This problem is corrected with this patch kit.

5.20.2. Images and/or Files Affected

[SYS$LDR]EXCEPTION.EXE
[SYS$LDR]EXCEPTION.STB
[SYS$LDR]EXCEPTION_MON.EXE
[SYS$LDR]EXCEPTION_MON.STB

5.20.3. VSI Case Identifier

Jiras BO-1570, DEV-1739

5.20.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

5.21. Correct Issues With SYS$UNWIND on Error or Abort

5.21.1. Problem Description

In a scenario where an $UNWIND operation is initiated intending to reach the base frame, the $UNWIND could fail to reach the base frame and the operation would be aborted. But, since the original mechanism depth was not restored on abort, control would return to the incorrect call frame. This could result in a hang, continuously retrying and aborting the same $UNWIND.

The mechanism depth is now saved during the $UNWIND so control can return to the correct call frame on error or abort.

5.21.2. Images and/or Files Affected

[SYS$LDR]EXCEPTION.EXE
[SYS$LDR]EXCEPTION.STB
[SYS$LDR]EXCEPTION_MON.EXE
[SYS$LDR]EXCEPTION_MON.STB

5.21.3. VSI Case Identifier

Jiras SPS-1469, BO-1600

5.21.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6. Problems Addressed from Previous Kits

6.1. System May Crash With DELCONPFN Bugcheck

6.1.1. Problem Description

When mapping large process-based PFN-mapped sections, the process working set may be left in an inconsistent state. When the process logs out, the system could then crash with a DELCONPFN bugcheck.

This issue can often occur when installing large images with the **`/SHARE=ADDRESS/RESIDENT`** qualifiers.

The problem is corrected with this patch kit.

6.1.2. Images and/or Files Affected

[SYS$LDR]SYS$VM.EXE

[SYS$LDR]SYS$VM.STB
[SYS$LDR]SYS$VM_MON.EXE
[SYS$LDR]SYS$VM_MON.STB

6.1.3. VSI Case Identifier

Jiras SPS-1273, SPS-1285, BO-1473

6.1.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.2. SYS$IO_PERFORM May Return Incorrect Data or Corrupt Files

6.2.1. Problem Description

Under some circumstances, the SYS$IO_PERFORM[W] system services may incorrectly compute the data buffer address for the I/O operation.

This can result in file corruption or application issues due to reading the wrong data.

The problem is corrected with this patch kit.

6.2.2. Images and/or Files Affected

[SYS$LDR]IO_ROUTINES.EXE
[SYS$LDR]IO_ROUTINES.STB
[SYS$LDR]IO_ROUTINES_MON.EXE
[SYS$LDR]IO_ROUTINES_MON.STB

6.2.3. VSI Case Identifier

Jiras SPS-1256, BO-1485

6.2.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.3. Disallow Shadow Sets as Dump Off System Disk (DOSD) Devices

6.3.1. Problem Description

The **SET DUMP/DEVICE** command allows the specification of shadow sets as Dump Off System Disk (DOSD) devices and **SHOW DUMP** will display a dump file found on the shadow set. However, the dump kernel cannot locate the dump file on a shadowed DOSD device.

6.3.2. Images and/or Files Affected

[SYSEXE]SETSHOWDUMP.EXE

6.3.3. VSI Case Identifier

Jira KT-191

6.3.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.3.5. Workaround

Do not specify shadow sets to **`SET DUMP/DEVICE`**.

6.4. CLUE$STARTUP Exhausts System Disk Space if no DOSD Devices Are Mounted

6.4.1. Problem Description

An error in CLUE$STARTUP failed to end the search for a dump disk if none of the devices specified in the SET DUMP command were mounted at the time CLUE$STARTUP runs during boot.

This could result in an infinite loop within the procedure, and the resulting log file would eventually consume all the space on the system device.

This problem is corrected with this patch kit.

6.4.2. Images and/or Files Affected

[SYS$STARTUP]CLUE$STARTUP.COM

6.4.3. VSI Case Identifier

Jiras KT-188, KT-195

6.4.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.4.5. Workaround

Make sure any DOSD devices are mounted before CLUE$STARTUP runs. Mounting the devices in SYS $STARTUP:SYPAGSWPFILES.COM fulfills this condition.

6.5. Protected Shareable Image Issue Could Cause Process or System Crash

6.5.1. Problem Description

For a merge-activation of a protected shareable image, the transfer code to relocate the image was incorrectly allocated in P0 space. This can result in an access violation in an inner mode if the P0 address space is deleted or overwritten, causing a process or system crash.

With this patch kit, the transfer code is correctly allocated in a protected area in P1 space, eliminating the risk and potential crash.

6.5.2. Images and/or Files Affected

[SYS$LDR]SYSTEM_PRIMITIVES_0.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_0.STB
[SYS$LDR]SYSTEM_PRIMITIVES_0_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_0_MIN.STB
[SYS$LDR]SYSTEM_PRIMITIVES_2.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_2.STB
[SYS$LDR]SYSTEM_PRIMITIVES_2_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_2_MIN.STB
[SYS$LDR]SYSTEM_PRIMITIVES_3.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_3.STB
[SYS$LDR]SYSTEM_PRIMITIVES_3_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_3_MIN.STB
[SYS$LDR]SYSTEM_PRIMITIVES_4.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_4.STB
[SYS$LDR]SYSTEM_PRIMITIVES_4_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_4_MIN.STB
[SYS$LDR]SYSTEM_PRIMITIVES_6.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_6.STB
[SYS$LDR]SYSTEM_PRIMITIVES_6_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_6_MIN.STB
[SYS$LDR]SYSTEM_PRIMITIVES_7.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_7.STB
[SYS$LDR]SYSTEM_PRIMITIVES_7_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_7_MIN.STB

6.5.3. VSI Case Identifier

Jira BO-1537

6.5.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.5.5. Workaround

Install the protected shareable image with shared address data.

6.6. RUNOFF Command May Display XPORT Errors or Corrupted Filenames

6.6.1. Problem Description

The RUNOFF code uses some special RMS internals to adjust RMS data structures for the files it is processing, for example the NAM fields using the FAB$V_NAM flag.

However, it incorrectly misses a few fields, especially the NAM$L_DEV field, which causes internal structure corruption when run on x86. This would result in XPORT errors and possibly display corrupted file names.

This problem is corrected with this patch kit.

6.6.2. Images and/or Files Affected

[SYSEXE]DSRINDEX.EXE
[SYSEXE]DSRTOC.EXE
[SYSEXE]RUNOFF.EXE

6.6.3. VSI Case Identifier

Jiras BO-1504, SPS-926, SPS-991

6.6.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.7. Various Corrections to OpenVMS Debugger Behavior

6.7.1. Problem Description

The following issues in the Debugger are corrected with this patch kit:

- Debugger is unable to examine Pascal strings.
- Debugger does not handle reading the DWARF namespace tag.
- Debugger cannot load a C++ program, failing with the following messages:

```
%DEBUG-E-ERROR, debugger error or session corruption
%DEBUG-I-INITIAL, Language: C++, Module: !AS
```

6.7.2. Images and/or Files Affected

[SYSLIB]DEBUG.EXE
[SYSLIB]DEBUGSHR.EXE

6.7.3. VSI Case Identifier

Jiras DEV-1683, DEV-1941, DEV-1967

6.7.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.8. ACMS Failures From Image Activation Issues

6.8.1. Problem Description

ACMS on x86 would encounter various failures, usually resulting in the server process crashing. Even the IVP would fail during product installation.

ACMS uses merge-activation of a protected shareable image. One portion of the structures involved in this mechanism was not correctly made permanent in P1 space. If the space was reused ACMS would encounter spurious failures.

The problem is corrected with this patch kit.

6.8.2. Images and/or Files Affected

[SYS$LDR]IMAGE_MANAGEMENT.EXE
SYS$LDR]IMAGE_MANAGEMENT.STB

6.8.3. VSI Case Identifier

Jira BO-1536

6.8.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.9. Misconfigured LAN Failover Devices or VLAN Devices for Cluster Boot

6.9.1. Problem Description

When a system configured as a cluster member boots, the cluster port driver, PEDRIVER, configures on all available LAN devices, including LAN Failover (LL) devices and VLAN (VL) devices.

For x86, a standalone version of LANACP.EXE is run to read the LAN device database and configure LAN Failover and VLAN devices.

An error in the code would only allow configuration of LLA and VLA devices, regardless of the actual device names. If the hardware included other devices, they would not be configured. If the hardware didn't include LLA or VLA, LANACP would attempt to configure the devices as LLA and VLA, resulting in an incorrect and unusable device configuration.

This problem is corrected with this patch kit.

6.9.2. Images and/or Files Affected

[SYSEXE]LANACP.EXE
[SYSEXE]LANACP_BFS.EXE

6.9.3. VSI Case Identifier

Jira SPS-1352

6.9.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.9.5. Workaround

This workaround requires redundant cluster communications paths so that cluster communications are not lost during these manual steps:

1. Remove the LAN Failover and VLAN definitions from the LAN permanent device database.
2. Reboot the system.
3. Do SCACP STOP LAN devname for the LAN Failover devices.
4. Do LANCP commands to set up the LAN Failover sets and VLAN devices.
5. Finally, do SCACP START LAN devname to start NISCA on the newly setup devices.

6.10. XA Gateway Failures in XaOpen() and IVP

6.10.1. Problem Description

XA Gateway XaOpen() function failed due to a latent coding error which did not affect other architectures for OpenVMS pre-x86.

Typical errors were like this one displayed by the IVP during kit installation:

```
IVP fail: failed to call gateway XaOpen function (-3)
```

This issue is corrected with this patch kit.

6.10.2. Images and/or Files Affected

[SYSEXE]DDTM$XG_SERVER.EXE

6.10.3. VSI Case Identifier

Jira SPS-1271

6.10.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.11. Add Support for Packed Decimal Routines Needed for BASIC

6.11.1. Problem Description

The following routines were added to the Object Time System (OTS) to enable packed decimal use required by the BASIC compiler.

- OTS$ADDP
- OTS$CMPP
- OTS$ASHP

These emulate the VAX instructions ADDPx, CMPPx and ASHP.

6.11.2. Images and/or Files Affected

[SYSLIB]LIBOTS2.EXE

6.11.3. VSI Case Identifier

Jiras RTLS-42, RTLS-424, DEV-2015

6.11.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.12. Availability Manager (DECamds) Could Induce INCONSTATE Bugcheck

6.12.1. Problem Description

The system will crash with an INCONSTATE bugcheck if the Availability Manager (DECamds) connects to an OpenVMS x86 system and displays single process information.

The routine used on x86 to collect process memory information for Availability Manager had an issue which could result in the INCONSTATE system crash.

This has been corrected with this patch kit.

6.12.2. Images and/or Files Affected

[SYS$LDR]SYS$VM.EXE
[SYS$LDR]SYS$VM.STB
[SYS$LDR]SYS$VM_MON.EXE
[SYS$LDR]SYS$VM_MON.STB

6.12.3. VSI Case Identifier

Jira AM-221

6.12.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.13. Access Violation in DTI$SHARE May Cause Process Deletion

6.13.1. Problem Description

When the DTI$SHARE image was installed with shared address data, a bug in the generated code caused an executive mode access violation. This would usually result in an unexpected process termination.

The issue is corrected with this patch kit.

6.13.2. Images and/or Files Affected

[SYSLIB]DTI$SHARE.EXE

6.13.3. VSI Case Identifier

Jiras SPS-1271, BO-1535

6.13.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.13.5. Workaround

Install the DTI$SHARE image without shared address data. On a running system which (by default) already has this image installed, use the command:

```
$ INSTALL REPLACE SYS$SHARE:DTI$SHARE - /OPEN/HEADER/SHARED=NOADDR/PROTECTED
```

6.14. Correct Spurious Exception Error Messages During Traceback Processing

6.14.1. Problem Description

Under some circumstances, traceback processing could encounter an exception, generating the message:

```
TRACEBACK - Exception occurred during traceback processing
```

An example of this occurred when the LSE editor encountered an error and attempted to display a traceback detailing the call stack.

This problem is corrected with this patch kit.

6.14.2. Images and/or Files Affected

[SYSLIB]TRACE.EXE

6.14.3. VSI Case Identifier

Jiras SPS-1339, DEV-2013

6.14.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.15. Add Support for x86 to BASIC Run-Time Library

6.15.1. Problem Description

Various modules in the BASIC RTL have been enhanced to enable x86-64 support. This includes support to:

- Extend some arguments from 32-bit integers to 64-bit integers
- Change conditional compilation symbols
- Update packed-decimal floating point passing mechanisms.

6.15.2. Images and/or Files Affected

[SYSLIB]DEC$BASRTL.EXE

6.15.3. VSI Case Identifier

Jira RTLS-423

6.15.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.16. DCL Procedure Error DOUBLEDEALO Bugcheck and Process Deletion

6.16.1. Problem Description

A DCL command procedure with certain syntax errors may result in a supervisor mode DOUBLEDEALO bugcheck, resulting in the deletion of the executing process.

This issue is corrected with this patch kit.

6.16.2. Images and/or Files Affected

[SYSEXE]DCL.EXE

6.16.3. VSI Case Identifier

Jiras SPS-1428, SPS-1429, BO-1541

6.16.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.17. Executive Mode Bugcheck May Cause KRNLSTAKNV System Crash

6.17.1. Problem Description

An executive mode bugcheck may be declared as fatal or non-fatal. A fatal executive mode bugcheck will crash the entire system. A non-fatal executive mode bugcheck will terminate the offending process.

If the BUGCHECKFATAL system parameter is set to 1, any executive mode bugcheck will cause a system crash (typically, this is used to get a crash dump snapshot of the underlying problem).

An error in the return mechanism from executive mode exception handling could result in a KRNLSTAKNV system crash, even if the BUGCHECKFATAL system parameter was set to 0 and the triggering executive mode bugcheck was declared as non-fatal.

This problem has been corrected with this patch kit.

6.17.2. Images and/or Files Affected

[SYS$LDR]EXCEPTION.EXE
[SYS$LDR]EXCEPTION.STB
[SYS$LDR]EXCEPTION_MON.EXE
[SYS$LDR]EXCEPTION_MON.STB

6.17.3. VSI Case Identifier

Jiras SPS-1375, SPS-1343

6.17.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.18. Remove Spurious Linker Warnings

6.18.1. Problem Description

The linker sometimes reports conflicting attributes for .text sections.

Code sections in object modules should have the SHR attribute. The x86 compilers did not always set this attribute. With some .text sections being SHR and some NOSHR, the linker reports this with a warning. The warnings at link time are propagated into the image.

With this change, the linker silently changes all PSECTS with a name prefix of .text and the EXE attribute to SHR. This way no link time warnings are reported or propagated into the image file due to this condition.

6.18.2. Images and/or Files Affected

[SYSEXE]LINK.EXE

6.18.3. VSI Case Identifier

Jiras DEV-1944, DEV-2019

6.18.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.19. Improve SECURITY_SERVER Behavior for Systems Under Persistent Attack

6.19.1. Problem Description

Systems under persistent attack could fail to run the CIA database maintenance routine indefinitely. This could result in accumulating expired records over time, with the possibility of introducing transaction timeouts and denial of service. Should the database grow large enough, it could exhaust required resources and halt transaction processing entirely.

To alleviate this condition, a new server thread has been added which runs the CIA database periodic maintenance routine at a selectable interval regardless of attack volume. This guarantees that expired records are removed from the database and avoids the potential issues.

A new system logical name, SYS$CIA_MAINTENANCE_INTERVAL, now allows system managers to set the interval, in minutes, between maintenance runs. The interval can be set based on the attack volume for a system or cluster. The larger the attack volume, the smaller the interval should be to help keep the database and transaction times as small as possible. Intervals can be set from 1 to 60 minutes, with 10 minutes being the default.

For example, to set an interval of 5 minutes, use the command:

```
$ DEFINE/EXEC/SYSTEM SYS$CIA_MAINTENANCE_INTERVAL 5
```

6.19.2. Images and/or Files Affected

[SYSEXE]SECURITY_SERVER.EXE

6.19.3. VSI Case Identifier

Jira BO-1506

6.19.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.20. Various Fixes to SECURITY_SERVER Behavior

6.20.1. Problem Description

The following issues have been corrected in the security server with this patch kit:

- The server was adding errant equivalent entries on multi-node clusters where login attempts from the same remote source fail on multiple nodes.
- Under some circumstances, attempting to send a message to OPCOM could errantly return SS $_INVARG status. This prevented certain messages being sent to the console. It could also trigger an unnecessary restart of the server.
- Redefining a default proxy to a standard proxy could result in a request timeout.
- An ill-formed audit (INSFARG) is triggered when converting a default proxy to a standard proxy.

6.20.2. Images and/or Files Affected

[SYSEXE]SECURITY_SERVER.EXE

6.20.3. VSI Case Identifier

Jiras BO-1506, BO-1531

6.20.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

6.21. Some Exceptions May Mistakenly Crash the System

6.21.1. Problem Description

Some exceptions which should have only deleted the current process mistakenly crashed the system.

This problem is corrected with this patch kit.

6.21.2. Images and/or Files Affected

[SYS$LDR]EXCEPTION.EXE
[SYS$LDR]EXCEPTION.STB
[SYS$LDR]EXCEPTION_MON.EXE
[SYS$LDR]EXCEPTION_MON.STB

6.21.3. VSI Case Identifier

Jiras BO-1540, SPS-1428

6.21.4. Release Version of VSI OpenVMS That Will Contain This Change

Next VSI OpenVMS x86-64 release after V9.2-2.

7. Images or Files Replaced

[SYS$LDR]EXCEPTION.EXE

Image name	"EXCEPTION"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:43.76
Image Checksum (MD5)	E712AAD6503690DC41E36EF25E47CD42

[SYS$LDR]EXCEPTION.STB

File creation date and time	30-MAR-2024 10:32:47.20
Checksum (MD5)	F6D8257AE522C0B7E9F1DC9EFA478DAD

[SYS$LDR]EXCEPTION_MON.EXE

Image name	"EXCEPTION_MON"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:54.87
Image Checksum (MD5)	982C5838DC79D1E9E72431D77AC6A820

[SYS$LDR]EXCEPTION_MON.STB

File creation date and time	30-MAR-2024 10:32:55.60
Checksum (MD5)	52DD4D5F84EBBCFE7091E221986B0356

[SYS$LDR]IMAGE_MANAGEMENT.EXE

Image name	"IMAGE_MANAGEMENT"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:02.51
Image Checksum (MD5)	79B4F1F225BEC2B944747006FDA39DCE

[SYS$LDR]IMAGE_MANAGEMENT.STB

File creation date and time	30-MAR-2024 10:33:02.64
Checksum (MD5)	DD7A66BC2AE82D8CB00CD56C45A05281

[SYS$LDR]IO_ROUTINES.EXE

Image name	"IO_ROUTINES"

Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:04.80
Image Checksum (MD5)	EEA70CDEADEB47411249C3D09E557E2D

[SYS$LDR]IO_ROUTINES.STB

File creation date and time	30-MAR-2024 10:33:05.45
Checksum (MD5)	6D1DD6A90235C789028816ED9BEF3AEA

[SYS$LDR]IO_ROUTINES_MON.EXE

Image name	"IO_ROUTINES_MON"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:06.41
Image Checksum (MD5)	CC97829FC8AF402F19096EEA276236C5

[SYS$LDR]IO_ROUTINES_MON.STB

File creation date and time	30-MAR-2024 10:33:06.95
Checksum (MD5)	29DDAF190FCBA1DA2F7DBB77A96FAC12

[SYS$LDR]SYS$PKDDRIVER.EXE

Image name	"SYS$PKDDRIVER"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:04.11
Image Checksum (MD5)	C6AB632D7BF8AA8CC81B4B14EF3BA49E

[SYS$LDR]SYS$PKMDRIVER.EXE

Image name	"SYS$PKMDRIVER"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:02.82

Image Checksum (MD5)	DFAADC7656ED60783E9B16499ACC1BAB

[SYS$LDR]SYS$VM.EXE

Image name	"SYS$VM"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:04.98
Image Checksum (MD5)	C62FCFA6EA0371E0F636B07CA667C925

[SYS$LDR]SYS$VM.STB

File creation date and time	30-MAR-2024 10:33:05.92
Checksum (MD5)	BDB0B515C0D9F1BC6123ADC603BC2D73

[SYS$LDR]SYS$VM_MON.EXE

Image name	"SYS$VM_MON"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:06.40
Image Checksum (MD5)	F9073601CECD37F82904275030F29E02

[SYS$LDR]SYS$VM_MON.STB

File creation date and time	30-MAR-2024 10:33:07.02
Checksum (MD5)	E8A8F8CFA094F7B806315E51B78BF021

[SYS$LDR]SYSTEM_PRIMITIVES_0.EXE

Image name	"SYSTEM_PRIMITIVES_0"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:12.16
Image Checksum (MD5)	F17C3D96613267FE7417E7DD6A61B55A

[SYS$LDR]SYSTEM_PRIMITIVES_0.STB

File creation date and time	30-MAR-2024 10:33:13.07

Checksum (MD5)	F663AADBB54E96E9A5C6EE362E4ADBAC

[SYS$LDR]SYSTEM_PRIMITIVES_0_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_0_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:13.65
Image Checksum (MD5)	633943B42723E8D85D17B06A4A79BA51

[SYS$LDR]SYSTEM_PRIMITIVES_0_MIN.STB

File creation date and time	30-MAR-2024 10:33:13.90
Checksum (MD5)	D3E70F57EC5FABBF7D93EBF3136E1B46

[SYS$LDR]SYSTEM_PRIMITIVES_2.EXE

Image name	"SYSTEM_PRIMITIVES_2"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:14.28
Image Checksum (MD5)	4EEDDEBAB6CED3DD5BEFE82D62E27E33

[SYS$LDR]SYSTEM_PRIMITIVES_2.STB

File creation date and time	30-MAR-2024 10:33:14.56
Checksum (MD5)	97C23DD4791F9C82B616211851D29F39

[SYS$LDR]SYSTEM_PRIMITIVES_2_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_2_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:15.00
Image Checksum (MD5)	917E2F4853E9988174BE2D7C90046876

[SYS$LDR]SYSTEM_PRIMITIVES_2_MIN.STB

File creation date and time	30-MAR-2024 10:33:15.21

Checksum (MD5)	057ABF3A07CA8B1F41CB3A44322D41E5

[SYS$LDR]SYSTEM_PRIMITIVES_3.EXE

Image name	"SYSTEM_PRIMITIVES_3"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:15.73
Image Checksum (MD5)	3A08D075F7F6BFC229E04469322054CD

[SYS$LDR]SYSTEM_PRIMITIVES_3.STB

File creation date and time	30-MAR-2024 10:33:15.95
Checksum (MD5)	2121DC1D8BFFB02F0ECD803F93D7ABA5

[SYS$LDR]SYSTEM_PRIMITIVES_3_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_3_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:16.31
Image Checksum (MD5)	E655324CCE6FC591BE82FD1C696DCD60

[SYS$LDR]SYSTEM_PRIMITIVES_3_MIN.STB

File creation date and time	30-MAR-2024 10:33:16.50
Checksum (MD5)	03969111FA917821F776CB39C779A4B2

[SYS$LDR]SYSTEM_PRIMITIVES_4.EXE

Image name	"SYSTEM_PRIMITIVES_4"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:17.03
Image Checksum (MD5)	FF8F91F0B0AFAF9E935A0AB92010BCDE

[SYS$LDR]SYSTEM_PRIMITIVES_4.STB

File creation date and time	30-MAR-2024 10:33:17.22

Checksum (MD5)	89D0AEFE6BED8C0B8734AE18F6C07E0D

[SYS$LDR]SYSTEM_PRIMITIVES_4_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_4_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:17.58
Image Checksum (MD5)	4E9297E584E307225A82B3838775BD64

[SYS$LDR]SYSTEM_PRIMITIVES_4_MIN.STB

File creation date and time	30-MAR-2024 10:33:17.77
Checksum (MD5)	1C86643C1B7F9D414257B9F1FDEC0CC5

[SYS$LDR]SYSTEM_PRIMITIVES_6.EXE

Image name	"SYSTEM_PRIMITIVES_6"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:18.24
Image Checksum (MD5)	11CC2FB2A70E6DC740108516D46F5DB8

[SYS$LDR]SYSTEM_PRIMITIVES_6.STB

File creation date and time	30-MAR-2024 10:33:18.44
Checksum (MD5)	26585B3C4FDDFCBEFAF57D095B8518ED

[SYS$LDR]SYSTEM_PRIMITIVES_6_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_6_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:18.78
Image Checksum (MD5)	525F9F66EEC1678252235B14040C4EF4

[SYS$LDR]SYSTEM_PRIMITIVES_6_MIN.STB

File creation date and time	30-MAR-2024 10:33:18.98

Checksum (MD5)	90DED419BC5107F1C77D7FC6D2E27C32

[SYS$LDR]SYSTEM_PRIMITIVES_7.EXE

Image name	"SYSTEM_PRIMITIVES_7"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:19.30
Image Checksum (MD5)	C164416B6DB4AAE33EC2810548477043

[SYS$LDR]SYSTEM_PRIMITIVES_7.STB

File creation date and time	30-MAR-2024 10:33:19.55
Checksum (MD5)	CF217269DB302BCA21B96114852A3A79

[SYS$LDR]SYSTEM_PRIMITIVES_7_MIN.EXE

Image name	"SYSTEM_PRIMITIVES_7_MIN"
Image file identification	"X-16"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:19.84
Image Checksum (MD5)	49A784F6EFD9042B3D3FF925297EAB3E

[SYS$LDR]SYSTEM_PRIMITIVES_7_MIN.STB

File creation date and time	30-MAR-2024 10:33:20.03
Checksum (MD5)	AA569562474092402244 1157C8854B03

[SYS$STARTUP]CLUE$STARTUP.COM

File creation date and time	16-NOV-2023 09:05:24.75
Checksum (MD5)	8D7AB0E9329BB815B4642195CFC7E908

[SYSEXE]AUTHORIZE.EXE

Image name	"AUTHORIZE"
Image file identification	"X-38"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:52.88

Image Checksum (MD5)	C375DD6D839A413E1F6BDC7F4929AF1C

[SYSEXE]DBGHK$PRCDUMP_KERNEL.EXE

Image name	"DBGHK$PRCDUMP_KERNEL"
Image file identification	"V9.2-009"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:43.39
Image Checksum (MD5)	862983063ABF3772772CC8B017DA9A9F

[SYSEXE]DCL.EXE

Image name	"DCL"
Image file identification	"X-61"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:52.72
Image Checksum (MD5)	D6831AF27584E1F8E7100CEDE4D902CC

[SYSEXE]DDTM$XG_SERVER.EXE

Image name	"DDTM$XG_SERVER"
Image file identification	"V2.1-X035-XGLJ"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:00.89
Image Checksum (MD5)	60931BF687A7D0584BB9B576AEFC17E8

[SYSEXE]DIRECTORY.EXE

Image name	"DIRECTORY"
Image file identification	"X02-01"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:41.61
Image Checksum (MD5)	3AE2CF69A86490CF10AFB9FB1F66093D

[SYSEXE]DSRINDEX.EXE

Image name	"DSRINDEX"

Image file identification	"V3.2-01"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:00.17
Image Checksum (MD5)	6EBB2C31EA328C8F0D2FC594FB3733EA

[SYSEXE]DSRTOC.EXE

Image name	"DSRTOC"
Image file identification	"V3.2-01"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:58.69
Image Checksum (MD5)	C842D5CE973A6A6D1C128A6E32FBD28C

[SYSEXE]LANACP.EXE

Image name	"LANACP"
Image file identification	"X-37"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:07.04
Image Checksum (MD5)	DF1675AFFD229F3E19986CCC259DE602

[SYSEXE]LANACP_BFS.EXE

Image name	"LANACP_BFS"
Image file identification	"X-37"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:07.61
Image Checksum (MD5)	A1C5E714A3F032672415E1EF6637EA10

[SYSEXE]LINK.EXE

Image name	"LINK"
Image file identification	"I02-97"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:50.20

Image Checksum (MD5)	59D6FB4697D002197567C7FFE5C011B1

[SYSEXE]MTAAACP.EXE

Image name	"MTAAACP"
Image file identification	"X-14"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:51.67
Image Checksum (MD5)	C78736AAA672424A3C3DA3064DB82CAD

[SYSEXE]RUNOFF.EXE

Image name	"RUNOFF"
Image file identification	"V3.2-01"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:55.85
Image Checksum (MD5)	173F048F75FADEDF9BE984B495A06ABA

[SYSEXE]SECURITY_SERVER.EXE

Image name	"SECURITY_SERVER"
Image file identification	"X-4"
Image build identification	""
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:52.51
Image Checksum (MD5)	3C52674D4FC69A3CF2D995B1B54A53D7

[SYSEXE]SETSHOSHADOW.EXE

Image name	"SETSHOSHADOW"
Image file identification	"X-02"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:04.72
Image Checksum (MD5)	247DD00F30336C1D55615C3A78B99100

[SYSEXE]SETSHOWDUMP.EXE

Image name	"SETSHOWDUMP"

Image file identification	"X-14"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:53.07
Image Checksum (MD5)	E66FAE8BF6F46FEB801A5A42059E6BEB

[SYSEXE]SYS$CONFIG.DAT

File creation date and time	23-FEB-2024 12:09:45.16
Checksum (MD5)	BD08B3C2CA1CCAB3D820FFD9FA267980

[SYSLIB]DEBUG.EXE

Image name	"DEBUG"
Image file identification	"V9.2-009"
Image build identification	"XGLJ-P4A-000011"
Link identification	"Linker I02-94"
Link Date/Time	12-FEB-2024 07:39:01.81
Image Checksum (MD5)	3B29B822F46C2F660F95C30DA9A6E71B

[SYSLIB]DEBUGSHR.EXE

Image name	"DEBUGSHR"
Image file identification	"V9.2-009"
Image build identification	"XGLJ-P4A-000011"
Link identification	"Linker I02-94"
Link Date/Time	12-FEB-2024 07:38:53.29
Image Checksum (MD5)	D3B8869AD350915BFE32CA6937A5744B

[SYSLIB]DEC$BASRTL.EXE

Image name	"DEC$BASRTL"
Image file identification	"X01-039"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:25.11
Image Checksum (MD5)	BFCFF11649C8189300CAFBC318B9E09A

[SYSLIB]DEC$COBRTL.EXE

Image name	"LIBCOB"

Image file identification	"V2.9-786"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:15.74
Image Checksum (MD5)	D7819E3A26B58F1EFCDC15FFC6C691EA

[SYSLIB]DPML$SHR.EXE

Image name	"DPML$SHR"
Image file identification	"T01-019"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:31:47.22
Image Checksum (MD5)	544198C1108300E318EFECC7BD40D9F8

[SYSLIB]DTI$SHARE.EXE

Image name	"DTI$SHARE"
Image file identification	"V2.1-X035-XGLJ"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:31:43.41
Image Checksum (MD5)	C54EAF98DA43CB47EA3B2EBDAB92097A

[SYSLIB]IMGDMP.EXE

Image name	"IMGDMP"
Image file identification	"X-86"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:49.97
Image Checksum (MD5)	DE0B4EA070427D9ECABCA80B9482629F

[SYSLIB]LIBOTS.EXE

Image name	"LIBOTS"
Image file identification	"V1.0-1"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:31:43.33

Image Checksum (MD5)	5C60DFB8CF588EE9449949C4E222D893

[SYSLIB]LIBOTS.STB

File creation date and time	30-MAR-2024 10:31:43.37
Checksum (MD5)	6C25C9233796FE1815640871C265C75F

[SYSLIB]LIBOTS2.EXE

Image name	"LIBOTS2"
Image file identification	"V2.8-77"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:08.79
Image Checksum (MD5)	A7B7A02A800946EDDB9E74FFAF7DB51B

[SYSLIB]LIBRTL.EXE

Image name	"LIBRTL"
Image file identification	"X01-001"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:31:43.36
Image Checksum (MD5)	B12B1DF6831F46DD3BF90862EBDA4F55

[SYSLIB]LIBRTL.STB

File creation date and time	30-MAR-2024 10:31:43.45
Checksum (MD5)	86F7FB93029F358BE3A48FE61348DD8F

[SYSLIB]SDA$SHARE.EXE

Image name	"SDA$SHARE"
Image file identification	"X-2"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 14:54:33.80
Image Checksum (MD5)	EA68A7CDBEA172C759AC399BBAE9F7A9

[SYSLIB]TRACE.EXE

Image name	"TRACE"

Image file identification	"V9.2-001"
Image build identification	""
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:32:15.67
Image Checksum (MD5)	109D67959DA97934E8004ABACE49483D

[SYSLIB]XFC$SDA.EXE

Image name	"XFC$SDA"
Image file identification	"V1.0"
Image build identification	"XGLJ-P4A-000017"
Link identification	"Linker I02-94"
Link Date/Time	30-MAR-2024 10:33:02.57
Image Checksum (MD5)	AC7F94BC47F41A4131C86B43A0202C38

[SYSUPD]AUTOGEN.COM

File creation date and time	30-MAR-2024 10:11:32.46
Checksum (MD5)	3DFE900CF9128285B45739E2ED2B004F

Note

VMS Software, Inc. will only distribute kits in signed form. There is no need for most customers to compare file checksums for security or kit integrity reasons.

However, some sites may require such checking even when using signed kits. The image or file checksums are supplied (in MD5 format) to provide comparisons to the extracted final kit files. To find a file checksum, use:

```
$ CHECKSUM/ALGORITHM=MD5 filename
$ SHOW SYMBOL CHECKSUM$CHECKSUM
```

Note

Because a file or image may be replaced by multiple patch kits over time, a PCSI generation number is used to ensure that the latest version of the file or image is preserved on your system during **`PRODUCT INSTALL`** of a patch kit. Should a particular kit installation discover a newer version of a file or image in place on the system disk, the following message will be displayed:

```
%PCSI-I-RETAIN, file <filename> will not be replaced because file from kit
 has lower generation number
```

This is a normal occurrence depending on the order of kit installation. The correct version of the file or image will remain on the system after the current kit installation. The `%PCSI-I-RETAIN` message is informational only and does not indicate a problem.

8. Installation Instructions

8.1. Compressed File

This kit is provided for download within a ZIP archive container file.

Info-ZIP's freeware ZIP and UNZIP tools are provided for use on this VSI OpenVMS version. Your site may have already set up symbols for these tools or other equivalent ZIP tools. If not, use the following command to define a symbol to run the UNZIP image:

```
$ UNZIP == "$SYS$SYSDEVICE:[VMS$COMMON.SYSHLP.UNSUPPORTED.UNZIP]UNZIP"
```

Then invoke UNZIP to unpack the kit using the command:

```
$ UNZIP VMS922X_UPDATE-V0200
```

This will extract the installable PCSI product kit file and its associated signed manifest (_VNC file), used for kit validation during **PRODUCT** commands.

VSI strongly recommends always using the manifest to validate the kit content during any **PRODUCT** commands. This will occur automatically if the files are both contained in the same directory.

8.2. Installation Command

Install this kit with the POLYCENTER Software Installation Utility by logging into the SYSTEM account, and typing the following at the DCL prompt:

```
$ PRODUCT INSTALL VMS922X_UPDATE [/SOURCE=location_of_kit]
```

The kit location may be a CD/DVD or a disk directory that contains the kit. The **/SOURCE** qualifier is not needed if the **PRODUCT INSTALL** command is executed from the same directory as the kit location.

This kit requires the use of **/RECOVERY_MODE** and **/SAVE_RECOVERY_DATA** and will automatically set them; they do not need to be present on the command line.

The release notes for any kit may be extracted prior to kit installation using the **PRODUCT EXTRACT RELEASE_NOTES** command.

User-selectable options for installation behavior and scripting are available in this kit, refer to *Appendix A, "User-Selectable Control Options and Scripting Considerations"* for further details.

Additional help on installing PCSI kits can be found by typing **HELP PRODUCT INSTALL** at the system prompt.

8.3. Special Installation Instructions

Should you need to remove this kit via **PRODUCT UNDO PATCH**, the same mandated reboot requirement is in effect as the memory disk image is changed back to the prior system content.

The kit will update the memory disk image automatically as the final part of installation.
There is currently no mechanism within the PCSI utility to cleanly invoke a system reboot for **PRODUCT UNDO PATCH**.

You will be instructed as the kit exits that you must perform this function manually in this case.

Note

When the SYS$MD.COM procedure is executing to update the memory disk image, some errors may be reported similar to these:

```
%INSTALL-I-NONRESSHRADR, image installed ignoring '/RESIDENT' image_name
                -INSTALL-E-NOGHREG, insufficient memory in the code or data granularity
                 hint region
```

or

```
%INIT-F-GHREGIONFULL, An allocation was attempted from GH region GH_RES_CODE_S2 but
                there is not enough space in the region for the allocation.
```

These are due to having both old and new copies of some images which are still being used until the system is rebooted. Typically they may be ignored as they will clear up during the reboot. Should there still be similar messages during system startup after reboot, you may need to use AUTOGEN to adjust the related system parameters.

Note

During **PRODUCT INSTALL** or **PRODUCT UNDO PATCH**, the PCSI utility may issue the following message:

```
There is not enough space on the memory disk.
                You must take these steps now to complete this installation:

                -  Run @SYS$UPDATE:SYS$MD
                -  Reboot the system
```

In all cases the kit procedure will run SYS$MD, there is no need for you to do this.

For **PRODUCT INSTALL**, the reboot is also automatically handled by the kit procedure and you need not do a reboot yourself.

For **PRODUCT UNDO PATCH**, you must manually reboot the system after the operation completes. The kit dialogue will remind you of this requirement at the end of the operation. There is currently no mechanism in PCSI to automatically invoke the system shutdown and reboot for **UNDO PATCH** operations.

9. Copyright

VMS SOFTWARE, INC. CONFIDENTIAL. This software is confidential proprietary software licensed by VMS Software, Inc., and is not authorized to be used, duplicated or disclosed to anyone without the prior written permission of VMS Software, Inc.

Copyright 2024 VMS Software, Inc.

10. Disclaimer of Warranty and Limitation of Liability

This patch is provided as is, without warranty of any kind. all express or implied conditions, representations and warranties, including any implied warranty of merchantability, fitness for particular

purpose, or non-infringement, are hereby excluded to the extent permitted by applicable law. In no event will VMS Software, Inc. be liable for any lost revenue or profit, or for special, indirect, consequential, incidental or punitive damages, however caused and regardless of the theory of liability, with respect to any patch made available here or to the use of such patch.

11. Patch ID

X86VMS-VMS922X_UPDATE-V0200--4

A. User-Selectable Control Options and Scripting Considerations

A.1. Controlling Kit Behavior for Introductory Questions

This kit provides user-selectable control options for kit dialogue interaction and automated scripting capability as described here in this appendix.

The general form of a VSI ECO patch kit, when using **`PRODUCT INSTALL`**, consists of three initial questions regarding these topics:

1. System disk backup: A reminder that VSI recommends backing up the system disk before installing patches, followed by a `Do you want to continue?` yes/no question, default is `YES`.

2. Reboot requirement: A summary of whether the kit being installed requires a system reboot, followed by a `Do you want to continue?` yes/no question, default `YES`.

3. Archival of updated files: A description of saving an "_OLD" copy of each image or file updated by the kit, followed by a `Do you want to save "_OLD" copies of replaced files?` yes/no question, default is `NO`.

Other questions may be asked later, depending on the target disk or system environment or other kit-specific requirements.

Note

An initial `Do you want to continue?` question may be asked directly by the PCSI utility during any **`PRODUCT`** command - this has nothing to do with the kit being used. To avoid that question, you must supply sufficient detail to uniquely identify the product you wish to use and specify **`/OPTIONS=NOCONFIRM`** on the **`PRODUCT`** command.

Control options are available to customize the dialogue for the initial three kit questions. The controls are logical names, which may be defined in the process logical name table with a value of YES or NO.

To modify the behavior of the VSI ECO patch kit regarding the initial questions, define one or more of the following logical names before issuing the **`PRODUCT INSTALL`** command.

- To skip one or more of the questions, define the corresponding logical name shown here to YES:

SKIP$BACKUP	Skips system backup awareness question.

SKIP$REBOOT	Skips system reboot awareness question.
SKIP$ARCHIVE_OLD	Skips question about saving "_OLD" files. This will take the default, which is NO.
SKIP$INTRO	Skips all three of the above questions.

- To specifically override the default for saving "_OLD" files, define this logical name to YES or NO:

VSIKIT$ARCHIVE_OLD	Sets an answer for saving "_OLD" files behavior. This will skip the archive "_OLD" files question regardless of the above SKIP$* logical names.

- Two additional logical names may be defined as YES to modify the amount of explanatory text displayed for each question:

VSIKIT$VERBOSE	Shows all explanatory text for questions.
VSIKIT$BRIEF	Skips some general details in the explanations.

  The default if neither name is defined is VERBOSE. If both names are defined to YES, VERBOSE overrides BRIEF. The BRIEF form is displayed for any questions that are skipped.

For example, to skip all three questions but save an archive "_OLD" copy of each replaced file:

```
$ DEFINE VSIKIT$ARCHIVE_OLD YES
$ DEFINE SKIP$INTRO YES
$ PRODUCT INSTALL kitname
```

A.2. Standard Behavior for YES/NO Questions Asked During Kit Installation

Any YES/NO questions asked during kit installation now follow these rules:

1. **Ctrl/Y** issued while a question is being asked will force the current **PRODUCT** operation to terminate. This is completely safe to do while the initial three questions are being asked during **PRODUCT INSTALL** as no changes have yet been made to the target disk.

2. Some questions may ignore **Ctrl/Y** and ask for a specific answer, if aborting the current operation may have side effects. PCSI may trap **Ctrl/Y** directly for some **PRODUCT** operations. **Ctrl/Y** may be disabled during some sensitive kit processing.

3. The default YES/NO answer is automatically chosen if a kit is installed from a batch job, unless explicitly overridden by a logical name which provides the particular value, such as VSIKIT$ARCHIVE_OLD.

A.3. Installing a Kit From a Batch Job

To install a kit from a batch job, you will need to fully qualify the kit name so PCSI will have enough information to select the kit without asking for confirmation. For example, to install this kit:

```
$ PRODUCT INSTALL VMS922X_UPDATE/VERSION=V2.0/OPTIONS=NOCONFIRM
```

If the kit is located in a directory other than the current default directory, you will also need to add the qualifier:

`/SOURCE=`*`location_of_the_kit`*

For a batch job, any YES/NO question will automatically select the default answer. Use the control logical names explained above to modify the behavior if necessary. For the system disk backup and reboot questions, the batch behavior is identical to the default. For the save "_OLD" files question, define the VSIKIT$ARCHIVE_OLD logical name to YES if you want to save copies of the files, since the batch default is NO.

A.4. Deprecated Logical Names From HPE Patch Kits

The three names listed below were used by older VSI OpenVMS patch kits for compatibility with HPE patch kit behavior. These old names continue to function, but VSI encourages you to modify any scripts you may have to use the new names shown instead:

Old Name	**New Name**
NO_ASK$BACKUP	SKIP$BACKUP
NO_ASK$REBOOT	SKIP$REBOOT
ARCHIVE_OLD	VSIKIT$ARCHIVE_OLD