V9.2-3 Update V2.0 ECO Kit for VSI OpenVMS x86-64

Release Notes

Publication Date: July 2025

Operating System: VSI OpenVMS x86-64 V9.2-3

Kit Name: VMS923X_UPDATE-V0200

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

Table of Contents

1. Kit Name

VMS923X_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 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 V9.2-3

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 any necessary corrective actions to free up disk space or defragment the volume, 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

VMS923X_UPDATE-V0100

4. Kit Dependencies

VMS923X_PCSI-V0100

Important

The VMS923X_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 and Changes Addressed in This Kit

The problems and changes addressed in this kit are grouped into the following sections:

- SYS Changes (Sections 5.1 to 5.7)
- RTL Changes (Sections 5.8 to 5.9)
- DEBUG Changes (Section 5.10)
- ACME Changes (Sections 5.11 to 5.15)
- Security Server Changes (Section 5.16)
- I/O Device Driver Changes (Sections 5.17 to 5.19)
- LAN Changes (Section 5.20)

SYS Changes

5.1. Condition Handler May Not Be Found for Some Exceptions

Problem Description

A routine that had "callq" as the last instruction would sometimes not have its associated condition handler invoked if the routine encountered an exception condition.

The common unwind routine in LIBRTL used in condition handler searches is now corrected (see *Section 5.8, "Condition Handler Stack Unwind May Fail on Certain Exceptions"*).

The OpenVMS executive exception handling now correctly locates the associated condition handler routine for this case.

Images and/or Files Affected

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

VSI Case Identifier

Jira BO-1973

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.2. MONITOR DISK/ITEM=QUEUE_LENGTH Issue With Bound Volume Sets

Problem Description

The **`MONITOR DISK/ITEM=QUEUE_LENGTH`** command would sometimes display a large, non-zero queue length for members of bound volume sets, even though the disk may actually have been idle.

This issue only occurred for bound volume sets consisting of multiple disk devices using the Fast I/O mechanism. The queue length fields were being updated inconsistently between the various volume set members.

This issue has been corrected with this update.

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

VSI Case Identifiers

Jiras DRIV-553, DRIV-554

Netsuite NS8211

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.3. System May Rarely Encounter KRNLSTAKNV Bugcheck

Problem Description

During procedure call handling, a very rare circumstance could cause the system to fail with the following crash:

```
KRNLSTAKNV, Kernel stack not valid
```

This crash is unpredictable in nature and very unlikely in production environments.

This issue has been corrected with this update.

Images and/or Files Affected

[SYS$LDR]SYSTEM_PRIMITIVES.EXE
[SYS$LDR]SYSTEM_PRIMITIVES.STB
[SYS$LDR]SYSTEM_PRIMITIVES_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_MIN.STB

VSI Case Identifier

Jira BO-2009

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.4. Rare Access Violation Under Unlikely Circumstances

Problem Description

Access checks against a buffer that has access to the start address but not the end address will correctly signal an access violation. However, an immediate re-check against the same buffer start address where the buffer was now accessible would sometimes trigger a spurious access violation.

VSI has only encountered this behavior in a regression test that was specifically inducing access failures. To encounter it with production software is extremely unlikely.

This issue has been corrected with this update.

Images and/or Files Affected

[SYS$LDR]SYSTEM_PRIMITIVES.EXE
[SYS$LDR]SYSTEM_PRIMITIVES.STB
[SYS$LDR]SYSTEM_PRIMITIVES_MIN.EXE
[SYS$LDR]SYSTEM_PRIMITIVES_MIN.STB

VSI Case Identifiers

Jiras QTV-1168, BO-2071

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.5. Spurious Access Violation During Exception Handling

Problem Description

If a buffer PROBE access check exceeds the end of the current user stack, the stack is automatically extended. However, an immediate re-PROBE using the same starting address would sometimes incorrectly signal an access violation.

This issue has been corrected with this update.

Images and/or Files Affected

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

VSI Case Identifier

Jira BO-2059

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.6. System May Crash With DELCONPFN Bugcheck During Process Exit

Problem Description

When a process is logging out or exiting, the virtual address space for that process is torn down and resources are returned to the system. However, if a doubly-mapped page was encountered during this processing, the system would sometimes crash with a DELCONPFN bugcheck.

This issue has been corrected with this update.

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

VSI Case Identifiers

Jiras SPS-1273, BO-1971

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.7. ACPI Interrupt Storm May Consume CPU Time

Problem Description

The ACPI (Advanced Configuration and Power Interface) mechanism on x86-64 systems may interrupt the system to deliver information regarding device configuration or power management.

On some x86-64 hosts, an "interrupt storm" from the ACPI subsystem mechanism may occur, delivering on the order of 50,000 interrupts per second for an unknown period of time. This can consume from 30% to 60% of an entire CPU in interrupt context to field and dismiss the interrupts. As far as we can determine, there is no underlying issue of any particular significance to the operating system for these events.

The ACPI handling code for OpenVMS has been enhanced to detect this condition. If it occurs, the ACPI interrupts are disabled and the code will switch to a polling mode to learn of any events which are relevant to OpenVMS.

Images and/or Files Affected

[SYS$LDR]SYS$ACPI.EXE
[SYS$LDR]SYS$ACPI0006.EXE
[SYS$LDR]SYS$SEDRIVER.EXE

VSI Case Identifier

Jira BO-1833

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

RTL Changes

5.8. Condition Handler Stack Unwind May Fail on Certain Exceptions

Problem Description

A routine that had "callq" as the last instruction would sometimes not have its associated condition handler invoked if the routine encountered an exception condition.

The common unwind routine in LIBRTL used in condition handler searches is now corrected.

The corrected unwind mechanism is built into LIBRTL for run-time use and in the STARLET.OLB object module for programs that must link the code statically.

Images and/or Files Affected

[SYSLIB]LIBRTL.EXE
[SYSLIB]LIBRTL.STB
LIB$$X86_UNWIND.OBJ, a module in [SYSLIB]STARLET.OLB

VSI Case Identifier

Jira BO-1973

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.9. Debugger STEP/INSTRUCTION May Return %DEBUG-W-BADSTACK Error

Problem Description

The Debugger would sometimes return a BADSTACK error during a **STEP/INSTRUCTION** command. For example:

```
%DEBUG-I-INITIAL, Language: BASIC, Module: ALEXA

DBG> step/inst
%DEBUG-W-BADSTACK, WARNING: stack corrupted; session integrity not
 guaranteed
stepped to ALEXA\ALEXA+1: movq    %rsp,%rbp
DBG>
```

This problem occured when multiple procedure ICB (Invocation Control Block) structures were used in parallel. The Debugger uses these to provide procedure tracing or stack unwinding operations. Some of the context from one ICB was mistakenly reused in a subsequent ICB.

The problem occurs in the run-time library routine that allocates and initializes an ICB.

This issue has been corrected with this update.

Images and/or Files Affected

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

VSI Case Identifiers

Jira BO-2101, DEV-2860

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

DEBUG Changes

5.10. Various Corrections to OpenVMS Debugger Behavior

Problem Description

The following issues in the Debugger have been corrected with this update:

- Many %DEBUG-I-EXPMEMPOOL messages errors are reported, eventually causing the Debugger to become unstable.
- %DEBUG-W-BADSTACK errors are often reported, usually occurring after the first **STEP** or **GO** command.
- Access violation errors occur under some circumstances.

Images and/or Files Affected

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

VSI Case Identifier

Jira DEV-2860

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

ACME Changes

Note

If you change the login mechanism from ACME to UAF (or vice versa) after installing this kit, you will need to take an additional step if you later remove the kit using the **PRODUCT UNDO PATCH** command. Refer to *Section 8.4, "Special Instructions for PRODUCT UNDO PATCH"* for details.

5.11. Generated Passwords May Be Restricted to a Maximum of 32 Characters

Problem Description

The **SET PASSWORD/GENERATE** command in a command procedure with the MIXED_CHARACTER generation algorithm could not generate passwords longer than 32 characters. Even if PWDMINIMUM was greater than 32, only a list of 32-character passwords was generated.

This issue has been corrected with this update. Passwords may be generated up to the supported limit of 64 characters.

Images and/or Files Affected

[SYSEXE]SETP0_ACME.EXE

VSI Case Identifier

Jira UT-315

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.12. OpenVMS ACME Agent Message Improvements

Problem Description

Trace content and error log entries for the OpenVMS ACME Agent have been updated for more consistent and accurate reporting.

A few minor behavioral errors have also been corrected.

Images and/or Files Affected

[SYSLIB]VMS$VMS_ACMESHR.EXE

VSI Case Identifier

Jira BO-1710

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.13. Batch Job With Multiple Submitted Files May Fail With Access Violation

Problem Description

Multiple command procedure files may be submitted for batch processing using a single **SUBMIT** command. The procedures will execute serially within the same batch job process.

If the ACME login mechanism was enabled for user authorization on the system, such a batch job would fail with %SYSTEM-F-ACCVIO status when attempting to start the second procedure.

This issue has been corrected with this update.

Images and/or Files Affected

[SYSEXE]LOGIN_ACME.EXE

VSI Case Identifier

Jira BO-2041

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

Workaround

The suggested workaround for this issue is to create a single command procedure file that invokes each of the specific procedures in order. For example, instead of using the following command:

```
$ SUBMIT A,B,C
```

submit a single procedure consisting of the following lines:

```
$ @A
$ @B
$ @C
```

5.14. Malformed Domain\Username Login May Trigger ACME Server Issues

Problem Description

When configured for external authentication, attempting to login with a username string that ends in the separator character "\" would trigger errant audits and possibly an ACME Server restart.

The following is an example of a malformed username at login:

```
Username: SALES\
```

This issue has been corrected with this update.

Images and/or Files Affected

[SYSLIB]LDAPACME$LDAP-STD_ACMESHR.EXE

VSI Case Identifiers

Jiras BO-673, BO-706

Netsuite NS2054

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

Workaround

The suggested workaround for this issue is to not end a username with "\".

5.15. Detached process creation failure with ACME-enabled LOGINOUT.EXE

Problem Description

If the OpenVMS ACME login was enabled, a process created using LOGINOUT.EXE as the image parameter to $CREPRC from a program in user mode would fail to authenticate and the created process would immediately terminate.

This issue has been corrected with this update.

Images and/or Files Affected

[SYSLIB]VMS$VMS_ACMESHR.EXE

VSI Case Identifier

Jira BO-2038

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

Security Server Changes

5.16. Security Server Failure and Spurious Server Restart

Problem Description

A bug in the Security Server led to an incorrectly initialized structure. This eventually caused stack corruption and a resultant access violation, which in turn caused the server to restart.

This issue has been corrected with this update.

Images and/or Files Affected

[SYSEXE]SECURITY_SERVER.EXE

VSI Case Identifiers

Jiras BO-2023, BO-2024

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

I/O Device Driver Changes

5.17. PCI Passthrough Support for HPE QLogic Fibre Channel Cards

Problem Description

This change enables PCI passthrough support for HPE SN1100Q, SN1600Q, and SN1610Q QLogic Fibre Channel cards on ESXi, Red Hat Enterprise Linux, and Oracle QEMU/KVM Linux Hypervisors.

This provides support only for Fibre Channel data disks. Booting from a Fibre Channel disk is not supported.

Images and/or Files Affected

[SYS$LDR]SYS$PGQDRIVER.EXE
[SYSEXE]SYS$CONFIG.DAT

VSI Case Identifiers

Jiras DRIV-192, DRIV-503, DRIV-511, DRIV-516, DRIV-560

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.18. Configuration Enhancements for SATA Devices

Problem Description

The OpenVMS SATA device driver (PKDDRIVER) has been modified to enable the following enhancements:

- Allow LUNs higher than 7 for secondary SATA volumes. This allows more devices to be configured.
- Allow shared interrupts with other devices on the same PCI bus. This prevents potential lost interrupts.

Images and/or Files Affected

[SYS$LDR]SYS$PKDDRIVER.EXE

VSI Case Identifiers

Jiras BO-1763, DRIV-523

Netsuite NS7612

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

5.19. Heavy Load Causes virtio-scsi Device to Hang

Problem Description

A synchronization error in the virtio-scsi device driver would sometimes lead to virtio-scsi device stalls or system hangs.

This issue has been corrected with this update.

Images and/or Files Affected

[SYS$LDR]SYS$VSPDRIVER.EXE

VSI Case Identifier

Jira BO-1885

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

LAN Changes

5.20. Net Applications Using a virtio-net Device May Hang on Start

Problem Description

Network connectivity would sometimes be lost due to stale MAC address use during net application initialization.

A network application (such as DECnet) may change the MAC address. Should that application be stopped when no other applications are present, the MAC address should be changed back to the default value.

A bug in the virtio-net driver (EVDRIVER) incorrectly left the device with the now-closed application MAC value.

A new application starting would assume it was using the default MAC address, but the device was still using the stale MAC address. This resulted in failure to transmit and receive packets using the default MAC address.

This issue has been corrected with this update.

Images and/or Files Affected

[SYS$LDR]SYS$EVDRIVER.EXE

VSI Case Identifier

Jira DRIV-555

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

Workaround

The suggested workaround for this issue is to reset the device when it is found to be non-functional. For example, the following command would reset the device `EIA`:

```
$ MCR LANCP UPDATE DEVICE/RESET EIA
```

This will reset the default MAC address properly.

MACRO Programming Support

5.21. Issues With MACRO Programs Using Floating-Point Instructions

Problem Description

Programs written in MACRO that use floating-point instructions for arithmetic, conversions, or comparisons may not work properly. The code that emulates VAX floating-point instructions sometimes returns incorrect operation condition codes.

For example:

- Floating-point instructions may not report the result of an operation is greater than zero.
- Conversion from floating-point to integer may not indicate an overflow error.
- Compare instructions with floating-point operands may not return expected status.

These issues do not affect programs written in high-level languages.

Only programs written in MACRO that use floating-point instructions are affected. Such programs must be re-linked after installing this ECO kit in order to use the corrected floating-point emulation.

Images and/or Files Affected

XMAC_FLOAT.OBJ, a module in [SYSLIB]STARLET.OLB

VSI Case Identifiers

Jiras BO-2085, BO-2090

Release Version of VSI OpenVMS That Will Contain This Change

The next VSI OpenVMS x86-64 release after V9.2-3.

6. Problems Addressed From Previous Kits

Refer to the relevant section in the *V9.2-3 Update V1.0 ECO Kit for VSI OpenVMS x86-64 Release Notes* [https://docs.vmssoftware.com/v9-2-3-update-v1-eco-kit-for-openvms-x-86-64-release-notes/#d0e124] for details of the problems addressed in the Update V1.0 kit.

7. Images or Files Replaced

[SYS$LDR]EXCEPTION.EXE

Image Name:	"EXCEPTION"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:27.29
Image Checksum (MD5):	E2A0224B4E0AE6D81C195AC0F795BF5B

[SYS$LDR]EXCEPTION.STB

File Creation Date and Time:	7-JUN-2025 05:09:27.44
Checksum (MD5):	A5A09FF100248299FB1034C9DCB848C3

[SYS$LDR]EXCEPTION_MON.EXE

Image Name:	"EXCEPTION_MON"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:27.65
Image Checksum (MD5):	D8DE82AFA06E0A138CED8D76B96B0050

[SYS$LDR]EXCEPTION_MON.STB

File Creation Date and Time:	7-JUN-2025 05:09:27.73
Checksum (MD5):	ACFEC17AB55473E58E762F4434025D1F

[SYS$LDR]IO_ROUTINES.EXE

Image Name:	"IO_ROUTINES"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:31.62
Image Checksum (MD5):	1D726B4DAD7D930C4CC6F12B16FD60D7

[SYS$LDR]IO_ROUTINES.STB

File Creation Date and Time:	7-JUN-2025 05:09:32.30

Checksum (MD5):	ECE77129EE5342C31BD049D4E2FDC5E9

[SYS$LDR]IO_ROUTINES_MON.EXE

Image Name:	"IO_ROUTINES_MON"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.18
Image Checksum (MD5):	012E09D38B727FFB821B1B24838A0E88

[SYS$LDR]IO_ROUTINES_MON.STB

File Creation Date and Time:	7-JUN-2025 05:09:33.57
Checksum (MD5):	1B0B98F9D69BC2EF18885DB284B92CD8

[SYS$LDR]PROCESS_MANAGEMENT.EXE

Image Name:	"PROCESS_MANAGEMENT"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:37.69
Image Checksum (MD5):	CA342DED322E738CE6A7B8EB78CAA463

[SYS$LDR]PROCESS_MANAGEMENT.STB

File Creation Date and Time:	7-JUN-2025 05:09:38.03
Checksum (MD5):	17132AA7F346081B543E85DB9AA10244

[SYS$LDR]PROCESS_MANAGEMENT_MON.EXE

Image Name:	"PROCESS_MANAGEMENT_MON"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:38.45
Image Checksum (MD5):	3E161F162693E7E17AE3F0168992FA7D

[SYS$LDR]PROCESS_MANAGEMENT_MON.STB

File Creation Date and Time:	7-JUN-2025 05:09:38.78
Checksum (MD5):	12E91B5E8A2B3D8D61E2244BB830BBC8

[SYS$LDR]RMS.EXE

Image Name:	"RMS"

Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.60
Image Checksum (MD5):	B40AAA45EE007D6CEDCFB8C5E831F423

[SYS$LDR]RMS.STB

File Creation Date and Time:	7-JUN-2025 05:09:33.78
Checksum (MD5):	88359AEAE9CEE8E0BCA8738438B73B90

[SYS$LDR]SYS$ACPI.EXE

Image Name:	"SYS$ACPI"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:30.22
Image Checksum (MD5):	36354647D6B41301BA817D338B2877CD

[SYS$LDR]SYS$ACPI0006.EXE

Image Name:	"SYS$ACPI0006"
Image File Identification:	"X-35"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:37.86
Image Checksum (MD5):	F74B4C747469EAEBE73BB161D7871F55

[SYS$LDR]SYS$EI1000X.EXE

Image Name:	"SYS$EI1000XDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:30.88
Image Checksum (MD5):	FF8434CC675B5BB1F647805048030687

[SYS$LDR]SYS$EVDRIVER.EXE

Image Name:	"SYS$VIRTIODRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:31.93

Image Checksum (MD5):	32B50E76ECE7A771211E605A574EEA9C

[SYS$LDR]SYS$LAN.EXE

Image Name:	"SYS$LAN"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:36.41
Image Checksum (MD5):	3B7169CFDC73FC8CFD6634D34D78E4FD

[SYS$LDR]SYS$LAN_CSMACD.EXE

Image Name:	"SYS$LAN_CSMACD"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:36.57
Image Checksum (MD5):	6622CFE5BB8BD76F81296047F5DC52A2

[SYS$LDR]SYS$LLDRIVER.EXE

Image Name:	"SYS$LLDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.70
Image Checksum (MD5):	DFEDB7646CA8E4527D8D604E10EC928A

[SYS$LDR]SYS$PGQDRIVER.EXE

Image Name:	"SYS$PGQDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:35.74
Image Checksum (MD5):	887C34BEF485F62E459874B75D9B2F4D

[SYS$LDR]SYS$PKDDRIVER.EXE

Image Name:	"SYS$PKDDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:34.22

Image Checksum (MD5):	2A5299C7A1240E86A176D9F35DA3386F

[SYS$LDR]SYS$SEDRIVER.EXE

Image Name:	"SYS$SEDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:30.73
Image Checksum (MD5):	1C38BA8FD30183689DD83D6F975602B7

[SYS$LDR]SYS$VLANDRIVER.EXE

Image Name:	"SYS$VLANDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.25
Image Checksum (MD5):	63822CB38FC76777CBC9A171562BABF4

[SYS$LDR]SYS$VM.EXE

Image Name:	"SYS$VM"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.65
Image Checksum (MD5):	E57B162BE805D50663C3740967889E74

[SYS$LDR]SYS$VM.STB

File Creation Date and Time:	7-JUN-2025 05:09:34.66
Checksum (MD5):	3182ADDA34CFA46F59DFF6C3C0AEA9EB

[SYS$LDR]SYS$VM_MON.EXE

Image Name:	"SYS$VM_MON"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:34.90
Image Checksum (MD5):	2F94BE30FC333CDF6A0B6D3C4BCA2FF0

[SYS$LDR]SYS$VM_MON.STB

File Creation Date and Time:	7-JUN-2025 05:09:35.12

Checksum (MD5):	C0F2200516FECB3F35E9522139B32266

[SYS$LDR]SYS$VMXNET3.EXE

Image Name:	"SYS$VMXNET3DRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:32.79
Image Checksum (MD5):	1F13BA90F6B58FC431F7D4E4FD37B0F4

[SYS$LDR]SYS$VSPDRIVER.EXE

Image Name:	"SYS$VSPDRIVER"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:36.85
Image Checksum (MD5):	B5B8DC9EDA8CE799DD96D68A6ED11AFB

[SYS$LDR]SYSGETSYI.EXE

Image Name:	"SYSGETSYI"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:35.80
Image Checksum (MD5):	F889D6966A5AD8DD7016CC7AEC1A4F53

[SYS$LDR]SYSGETSYI.STB

File Creation Date and Time:	7-JUN-2025 05:09:35.86
Checksum (MD5):	7416CAC951DCDFC06B68D1EF59374CE4

[SYS$LDR]SYSTEM_PRIMITIVES.EXE

Image Name:	"SYSTEM_PRIMITIVES"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:38.57
Image Checksum (MD5):	B424484C7F44E54953AFAA7BAA1E3664

[SYS$LDR]SYSTEM_PRIMITIVES.STB

File Creation Date and Time:	7-JUN-2025 05:09:38.92

Checksum (MD5):	3E961896C2050B0358774D3156F282F8

[SYS$LDR]SYSTEM_PRIMITIVES_MIN.EXE

Image Name:	"SYSTEM_PRIMITIVES_MIN"
Image File Identification:	"X-16"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:40.01
Image Checksum (MD5):	1FAF4F87B60199CDAC5CC9C45183CC18

[SYS$LDR]SYSTEM_PRIMITIVES_MIN.STB

File Creation Date and Time:	7-JUN-2025 05:09:40.59
Checksum (MD5):	5AF883A9BC2BE86408FAD3B962BE4452

[SYSEXE]ACME_SERVER.EXE

Image Name:	"ACME_SERVER"
Image File Identification:	"X-46"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:27.41
Image Checksum (MD5):	C27A2EC8D5728276ECCF38D5C00205A9

[SYSEXE]LANACP.EXE

Image Name:	"LANACP"
Image File Identification:	"X-40"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:34.66
Image Checksum (MD5):	94B7BC36D62A5B01E1E495D387ED31CB

[SYSEXE]LANCP.EXE

Image Name:	"LANCP"
Image File Identification:	"X-126"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:34.37
Image Checksum (MD5):	E98330DE51F8C97BE24BDFD83BB8BEDE

[SYSEXE]LINK.EXE

Image Name:	"LINK"

Image File Identification:	"I02-99"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:27.98
Image Checksum (MD5):	B7DE153549F7EF74E04A0532EA138712

[SYSEXE]LOGIN_ACME.EXE

Image Name:	"LOGIN_ACME"
Image File Identification:	"LOGIN_ACME"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:29.55
Image Checksum (MD5):	DFC63410C2CF3C7E36230B50D2BDD4FB

Note

If the system is using the ACME login mechanism when this kit is installed, the image content from LOGIN_ACME will also be copied to [SYSEXE]LOGINOUT.EXE.

[SYSEXE]MACRO.EXE

Image Name:	"MACRO"
Image File Identification:	"60-117-50F9M"
Image Build Identification:	""
Link Identification:	"Linker I02-98"
Link Date/Time:	31-MAR-2025 11:50:01.13
Image Checksum (MD5):	AF54D00566913B748E1CE995B163C117

[SYSEXE]SCACP.EXE

Image Name:	"SCACP"
Image File Identification:	"X-36"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:29.58
Image Checksum (MD5):	9DCC51A83819C5DE5C0418275CC0184E

[SYSEXE]SDA.EXE

Image Name:	"SDA"
Image File Identification:	"X-6"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:29:51.13

Image Checksum (MD5):	0885C7E8D460DB540420E6AEBD85F9DE

[SYSEXE]SEARCH.EXE

Image Name:	"SEARCH"
Image File Identification:	"X02-08"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:34.41
Image Checksum (MD5):	A34D1432B4DE48CE24901829845DA123

[SYSEXE]SECURITY_SERVER.EXE

Image Name:	"SECURITY_SERVER"
Image File Identification:	"X-4"
Image Build Identification:	""
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:32.78
Image Checksum (MD5):	712141737350FD2E9F5F81F5C08E6C81

[SYSEXE]SETP0_ACME.EXE

Image Name:	"SETP0_ACME"
Image File Identification:	"LOGIN_ACME"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:10:15.58
Image Checksum (MD5):	7994FCECD37AA87771858B542A549DA4

Note

If the system is using the ACME login mechanism when this kit is installed, the image content from SETP0_ACME will also be copied to [SYSEXE]SETP0.EXE.

[SYSEXE]SYS$CONFIG.DAT

File Creation Date and Time:	2-JUN-2025 08:52:04.05
Checksum (MD5):	CC5923FEEF83C10591A37E12DD0946EC

[SYSEXE]SYSBOOT.EXE

Image Name:	"SYSBOOT"
Image File Identification:	"X-3"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.02

Image Checksum (MD5):	E6752BCC64C213A913C8AE2CDF835D99

[SYSHLP]ACMELDAP_STD_CONFIG_INSTALL.PDF

File Creation Date and Time:	7-JUN-2025 04:44:21.22
Checksum (MD5):	74E4BB873FBBB2BD668A3A67610D6FC1

[SYSHLP]ACMELDAP_STD_CONFIG_INSTALL.TXT

File Creation Date and Time:	23-JAN-2025 09:26:58.68
Checksum (MD5):	AADC5FC7665427D39443932C36DEAA7A

[SYSHLP]LANCP$HELP.HLB

File Creation Date and Time:	7-JUN-2025 05:04:36.46
Checksum (MD5):	7DBCCDF7E1CDEA271EA36C3CA8EF03E3

[SYSLIB]CLUE$SDA.EXE

Image Name:	"CLUE$SDA"
Image File Identification:	"X-92"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:26.51
Image Checksum (MD5):	8C8C70C67F4BED423566AA5D7B160856

[SYSLIB]DBG$HA_KERNEL.EXE

Image Name:	"DBG$HA_KERNEL"
Image File Identification:	"V9.3-012"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:40.78
Image Checksum (MD5):	4ED6A55F948F3EF6B9DA76AC1D893E44

[SYSLIB]DBG$HA_MAIN.EXE

Image Name:	"DBG$HA_MAIN"
Image File Identification:	"V9.3-012"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:40.33
Image Checksum (MD5):	89413B0F55CBC9E78F74E1B00A07C195

[SYSLIB]DEBUG.EXE

Image Name:	"DEBUG"
Image File Identification:	"V9.3-012"

Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.62
Image Checksum (MD5):	28BBD555C5B58267059D46474D080980

[SYSLIB]DEBUGSHR.EXE

Image Name:	"DEBUGSHR"
Image File Identification:	"V9.3-012"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:30.80
Image Checksum (MD5):	F180ECA7A0DA157A8827323E51DDF713

[SYSLIB]DEBUGUISHR.EXE

Image Name:	"DEBUGUISHR"
Image File Identification:	"V9.3-012"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:33.62
Image Checksum (MD5):	C0179DA376A9E79BE88BED901DF007B8

[SYSLIB]DEC$BASRTL.EXE

Image Name:	"DEC$BASRTL"
Image File Identification:	"X01-039"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:16.79
Image Checksum (MD5):	C081E4A7A37663BAC2255510C6DE8B78

[SYSLIB]IOGEN$VIRTIO_CONFIG.EXE

Image Name:	"IOGEN$VIRTIO_CONFIG"
Image File Identification:	"X-6"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:03.40
Image Checksum (MD5):	6EEE7C62813FDD11204639AE78D29915

[SYSLIB]LAN$SDA.EXE

Image Name:	"LAN$SDA"
Image File Identification:	"X-92"

Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:35.95
Image Checksum (MD5):	AAAA6B21894ECDAC74BEC593D6284E0D

[SYSLIB]LDAPACME$LDAP-STD_ACMESHR.EXE

Image Name:	"LDAPACME$LDAP-STD_ACMESHR"
Image File Identification:	"LDAP-STD V1.30"
Image Build Identification:	""
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:27.41
Image Checksum (MD5):	00D0BE4FAEB03CE05F7EB7D5B2C8D039

[SYSLIB]LIBRTL.EXE

Image Name:	"LIBRTL"
Image File Identification:	"X01-001"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:08:39.80
Image Checksum (MD5):	706C5EC4AE9C953596F6BC4CC2AF426D

[SYSLIB]LIBRTL.STB

File Creation Date and Time:	7-JUN-2025 05:08:39.97
Checksum (MD5):	90696B3F6E1E535A09AD8428AF2CE470

[SYSLIB]PAS$RTL.EXE

Image Name:	"PAS$RTL"
Image File Identification:	"PAS$RTL V5.0-31"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:03.47
Image Checksum (MD5):	E3CA94F77C9A6CAC214A1BCDE81B0C36

[SYSLIB]SDA$SHARE.EXE

Image Name:	"SDA$SHARE"
Image File Identification:	"X-2"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:29:50.88
Image Checksum (MD5):	61DA8301FBE41BBDF7C902DB8832B946

[SYSLIB]VMS$VMS_ACMESHR.EXE

Image Name:	"VMS$VMS_ACMESHR"
Image File Identification:	"VMS_AGENT V2.0"
Image Build Identification:	""
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:26.80
Image Checksum (MD5):	8DCB7B1BD0B93EB7E00B88EF9D194A42

[SYSMGR]VMS$IMAGES_MASTER.DAT

File Creation Date and Time:	7-JUN-2025 04:44:31.35
Checksum (MD5):	A9D317C7D2E59192031A796BDA4CF90C

[SYSMSG]CLIUTLMSG.EXE

Image Name:	"CLIUTLMSG"
Image File Identification:	"0"
Image Build Identification:	"XGRV-K6N-000029"
Link Identification:	"Linker I02-97"
Link Date/Time:	7-JUN-2025 05:09:30.97
Image Checksum (MD5):	C9055CD55D12672FAC2B45BE27D6C983

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

As a file or image may be replaced by multiple ECO 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 an ECO 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 VMS923X_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 command at the DCL prompt:

```
$ PRODUCT INSTALL VMS923X_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 similar to the following may be reported:

```
%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 that 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 both cases, the kit procedure will run SYS$MD automatically. There is no need for you to execute SYS$MD before the reboot.

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.

8.4. Special Instructions for PRODUCT UNDO PATCH

If you change the login mechanism from ACME to UAF (or vice versa) after installing this kit, you will need to take an additional step if you later remove the kit using the **PRODUCT UNDO PATCH** command.

During **PRODUCT UNDO PATCH**, the PCSI utility is unaware that the LOGINOUT and SETP0 images in use by the system have different content from when the kit was installed. This can result in stale content or the wrong login style images in use after **PRODUCT UNDO PATCH**.

To avoid a spurious change in the system login mechanism, use the SYS$MANAGER:SYS$LOGIN_SWITCH.COM procedure to reset the login style to the desired mechanism, using the following command:

```
$ @SYS$MANAGER:SYS$LOGIN_SWITCH
```

You may use SYS$LOGIN_SWITCH.COM to either:

- Restore the initial login style before using **PRODUCT UNDO PATCH** if you want to return to the original style.
- Reset the login type back and forth from the new style after using **PRODUCT UNDO PATCH**. For example, if you are now using the ACME login and want to continue to use it, invoke the SYS$LOGIN_SWITCH.COM procedure twice, changing it to UAF and then back to ACME. This will correctly reset the ACME login images using the pre-kit installation images.

If you have not changed the login style, the **PRODUCT UNDO PATCH** command will correctly restore the login state with no additional user actions required.

9. Copyright

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-VMS923X_UPDATE-V0200--4

Note

The terms "ECO kit" and "patch kit" may be used interchangeably in this document. This also applies for other VSI OpenVMS documentation when describing PCSI kits that provide remedial updates to a particular product.

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 OpenVMS ECO 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 updates, 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 is `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 a VSI OpenVMS ECO 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 (for example, if aborting the current operation may have side effects for the system). Additionally, note the following:

   - 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 that 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 VMS923X_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 ECO Kits

The three names listed below were used by older VSI OpenVMS ECO kits for compatibility with HPE ECO 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

Old Name	New Name
ARCHIVE_OLD	VSIKIT$ARCHIVE_OLD