StorageTek Automated Cartridge System Library Software

Release Notes

Version 8.1

Part Number: E25015-02
April 2012

Submit comments about this document to STP_FEEDBACK_US@ORACLE.COM.

StorageTek Automated Cartridge System Library Software Release Notes

E25015-02

Oracle welcomes your comments and suggestions for improving this book. Contact us at STP_FEEDBACK_US@ORACLE.COM. Please include the title, part number, issue date, and revision.

ACSLS 8.1 Release Notes

ACSLS 8.1 includes several product enhancements and multiple fixes to bugs reported in Release 8.0.2.

Download ACSLS from the Oracle Software Delivery Cloud website. Typical installations download the package to the `/opt` directory and extracts it from there. A separate zip file is created for Solaris SPARC and Solaris x86. If you need both platforms, you must download both zip files.

Requirements

Software Requirements

- ACSLS 8.1 has been fully tested and verified on Oracle's Sun SPARC and X86 platforms running Solaris-10 Updates 7, 8, and 9. Other operating systems, including AIX and virtual environments, are not tested or supported.
- Because of special device driver requirements to enable virtual libraries, ACSLS 8.1 cannot run in a Solaris Zoned environment. However, ACSLS 8.1 can run in a logical domain on a SPARC system with Chip Multithreading (CMT) technology. ACSLS 8.1 HA systems must be installed on their own dedicated platform pair.
- The graphical user interface and SMCE service in ACSLS 8.1 requires Java 1.6 U14 or later. You cannot install ACSLS unless this requirement is met.

  The latest update for JDK 6 is available from the Oracle technetwork download site.

  http://www.oracle.com/technetwork/java/javase/downloads/index.html

  Please note that:

  - Should you choose to install Java in a different directory, a soft link must be provided that points to the java binaries located in: `/usr/java/bin/java`
  - To verify the version, enter the command: `java -version`.
- If you have an IPv4-based edge firewall, it should be configured to drop all outbound IPv4 protocol 41 packets and UDP port 3544 packets to prevent Internet hosts from using any IPv6-over-IPv4 tunnelled traffic to reach internal hosts.

Please check the website for any maintenance releases.

System Requirements

- Memory: 2GB minimum
- Swap: A minimum of 2GB swap is required. For systems configured with more than 6GB of memory, the rule of thumb for swap is approximately 30% of physical memory.
- File systems:

  Installation fails if the following filesystems do not exist as separate filesystems.

  - `/export/home` - 5GB or greater
  - `/export/backup` - 5GB or greater
- Optional Fibre card

  A fibre HBA is required if you intend to make use of the 'Logical Library' target feature in ACSLS 8.x.

  - To operate in target-mode, this HBA must be a contemporary QLogic fibre card (4GB or higher).
  - If you intend to operate a fibre-connected library such as the SL500, you can use any standard 2GB or higher fibre HBA.

Browser Requirements:

ACSLS 8.1 has been tested and fully verified on the following browsers:

- Internet Explorer 8 and 9
- FireFox 7 and 8
- Chrome 15

Co-Hosting

To ensure uninterrupted library service and to avoid unanticipated problems due to resource contention, it is generally recommended that ACSLS run in a stand-alone environment on a dedicated server. However, some systems are specifically designed to allow multiple applications to run in co-hosted fashion as though they are completely isolated from one another. Specifically, Solaris Containers and Oracle Solaris VM Server for SPARC enable conditional co-hosting possibilities for use with ACSLS.

The following details the conditions and limitations associated with the various co-hosting options for an ACSLS application.

- Solaris Containers (zones)

  Solaris Containers (or zones) enable a system administrator to partition a standard, low cost server into four independent Solaris systems, each with its own isolated file system, and its own instance of Solaris. You can assign network resources to each container and you can reboot any local (non-global) zone without affecting applications in other zones on the same platform.

However, the ability to share kernel resources (such as device drivers) across multiple zones is tenuous at best. Ideally, an application that requires kernel drivers would reside in the global zone. However, it is generally not good practice to install an application in the global zone since any fatal condition with the application could impact all other applications running in the other zones.

ACSLS 8.x can reside in a Solaris container only if it does not require drivers beyond the network interface. If you intend to use the target-mode fibre-channel driver (*qlt*) which is required for logical libraries, then your application should not be installed in a Solaris container. Or, if you intend to make use of a fibre-attached library which requires the *mchanger* driver, the application should not be installed in a Solaris container.

> **Note –** There are no versions of ACSLS-HA that are supported for use in Solaris Containers.

- Oracle VM Server for SPARC

  Oracle VM Server for SPARC (formerly Logical Domains or LDOMs) is technology available on SPARC T-series servers with Chip Multithreading (CMT) technology. This technology offers significant advantages over Solaris Containers to the extent that each domain is in control of its own Solaris kernel.

  A Solaris administrator can partition hardware resources across the system, assigning a specific resource to a specific domain. Network resources on this virtual machine can easily be shared across any of up to 128 'guest domains' on the server. But applications that require access to I/O devices through the PCIe bus must be installed in special 'I/O domains'. The number of I/O domains that you can create on the VM Server depends on the number of discrete PCIe busses on the SPARC platform. On a system with a single PCIe bus, you can have two I/O domains, and one of these must be the control domain.

  Any ACSLS application that relies solely on network connectivity to the library and for client applications can be installed in a guest domain on this server. The virtual network set-up procedure is described in the document, *Oracle VM Server for SPARC 2.1 Administration Guide* in the section, entitled "Using Virtual Networks".

  If your ACSLS 8.x application is intended for use with logical libraries, or if you intend to connect to a fibre-channel library such as the SL500 or L700, then ACSLS must be installed in an I/O domain. Refer to the section "Setting up I/O Domains" in the *Oracle VM Server for SPARC 2.1 Administration Guide*.

  Solaris Cluster Software is supported on the Oracle VM Server for SPARC and this platform can be employed in an ACSLS-HA application. Refer to the *Oracle Solaris Cluster Data Service for Oracle VM Server for SPARC Guide*.

What’s New

Enhancements

- Fast Load support for SCSI clients

  ACSLS provides Fast Load support for Fibre Channel (FC) SCSI clients. With Fast Load, an FC initiator that issues a mount operation receives a successful response once the operation has been validated and accepted by ACSLS, but before cartridge movement begins. Confirming that the cartridge is loaded and available for use (for example, by issuing Test Unit Ready or Read Element Status requests) is the responsibility of the client.

To set Fast Load reporting, click the Preferences button on the masthead from any screen. Once the Preference screen displays, you can enable Fast Load reporting by selecting "Yes" from the pull down menu. The default value is "No".

Changes to this setting are effective for new mount operations that are received from FC clients, and any mount operations that were already in progress are unaffected by the change.

- Oracle's WebLogic replaces the Sun Java Web Console as the application server for the ACSLS GUI. The change assures on-going support for the GUI infrastructure.
- New `userAdmin.sh` utility

  This utility allows the `acsls_admin` user to manage user accounts for the ACSLS GUI. With this utility, the acsls_admin user can create user accounts, delete user accounts, change passwords for any user account, and list all user accounts.
- New `fixVols.sh` utility

  This utility lets you manually correct the status of a logical volume that is stuck with an "in-transit" status when the physical volume is actually home or in the drive.
- Allow *`dismount force`* of a reserved drive

  Allow a "dismount force" to proceed even if a drive was left marked as "reserved" by a mount or dismount operation that never got a response.

  > **Note –** A reserved drive is reported as "in use" to `cmd_proc` or to an ACSAPI client.
- Send Requests to SL8500 that processes the request

  To optimize performance of a string of SL8500s connected by pass-thru-ports, send requests to a library that is the source or destination of a movement operation or will be responding to a status. For requests that are for any library and for requests whose library(s) are not connected, spread out these requests over all connected libraries. Make other changes to improve library performance.
- You can now change ACSLS environment variables without having to logout/login so these variables are used the next time that ACSLS is enabled. The ACSLS start-up script now sources the static variables.
- The `LM_RP_TRAIL` dynamic variable provides a audit trail (in `$ACS_HOME/log/rpTrail.log`) of all ACSLS requests. The associated `RP_TRAIL_LOG_SIZE`, `RP_TRAIL_FILE_NUM`, and `RP_TRAIL_DIAG` variables let you configure this audit trail. Access the `RP_TRAIL` variables via `acsss_config` (Option 2: Set event logging variables), and see the help for more details.
- Set the `CSI_MULTI_HOMED_CL` (client) dynamic variable to TRUE when an ACSAPI client server has multiple host addresses, is behind NAT with a private IP address that is not accessible externally, or the client's server is connected to ACSLS via a VPN. Access `CSI_MULTI_HOMED_CL` via `acsss_config` (Option 1: Set CSI tuning variables), and see the help for more details.

- The STMF service is now under `acsss enable` and `acsss disable` control. This service is a prerequisite for the SMCE service. It is enabled whenever SMCE is enabled, and it is disabled whenever SMCE is disabled.
- `psacs` now reports all ACSLS transient processes, including recovery, vary, audit, enter, eject, and config.
- ACSLS CAP enhancements include:
  - Improvements to automatic recovery:
    - During startup - detection of cartridges left in a CAP as it comes online.
    - During runtime - recovery from unusual situations, such as an inoperative CAP.
  - ACSLS no longer reports CAP as "in use" (although it appeared available and idle).

    Now when ACSLS detects that the CAP is reserved, but that the reservation could be overridden (meaning it appears to be reserved by ACSLS, and no conflicting ACSLS operation is underway), the operation is now allowed to proceed instead of being rejected with "in use" as before.
  - Improvements to SL3000 AEM eject:
    - When ejecting cartridges to an SL3000 AEM, skips over missing CAP magazines.
    - Directs ejected cartridges so they are easy to remove from the AEM by:

      Filling up magazines in the front door first, because the front door is easier to access.

      Filling up one magazine at a time, instead of striping cartridges across all columns.
- ACSLS GUI enhancements include:
  - Enhanced logical library volume counts. On the Logical Library Detail Information page, the total number of assigned volumes for a logical library is now provided as two separate counts:
    - Number of Volumes - This represents volumes that are assigned, are in the library, and are generally available for client use.
    - Number of Volumes Outside Library - This represents volumes that are assigned, but are not available for client use because they have been ejected – either physically ejected, or marked for eject by the FC client.
- During ACSLS installation, you can choose whether to install the GUI or to delay installation.

  The packages for the ACSLS GUI, etc. are still installed. Should you wish to install the GUI later, you need to: export the database and control files; rerun `install.sh` and install the GUI, etc.; then import your saved database and control files.
- 508 Accessibility: Alternate text tooltips have been added to support every graphic component in the GUI.

Fixes

- Added checks during `pkgadd` for 3GB or more available disk space. Aborts if there's insufficient space.
- The `checkinstall` script was changed to better parse the Solaris update level from the system file, `/etc/release`. The script compares this update level with the prescribed levels for ACSLS 8.1, and it will warn you if the update level is less than the minimum for the release.
- If you are not installing on a 64-bit system, `install_slm.sh` warns you about the Logical Library support requirement and prompts whether to continue the installation.
- Updated `showDevs.sh`

  Updated lookup table for contemporary QLogic and Emulex HBA model names. Restored LUN and WWN information on Solaris 10.
- Updated `testports` for IPv6 configurations.
- Prevents false reporting of an already-running system by the status_bar startup display.
- Avoids false timeouts during startup on a fully-configured SL3000

  Revised the algorithm that calculates allotted startup-recovery time to include the number of drives and CAPs. This allows a fully configured SL3000 come online within a reasonable time.
- When no volume was in the drive, 16 Blanks were returned as the *`vol_id`* to `display drive`.
- `switch lmu` command from ACSLS succeeds, but LSMs do not come back online.
- ACSLS 8.0+ volrpt might not accept valid legacy volume ranges (specified via the -v option).
- Removed the old versions of `licensekey.sh` and `get_license_info` that were included in ACSLS 8.0.2.
- Avoids `config acs` timeouts for large SL8500 configurations.
- When the `config` utility was executed multiple times, some ACSLS operations hung (`audit`, `query acs`, `query lmu`, etc.). Mounts and dismounts continued to work.
- `watch_vols` start failed to service more than one *`pool-id`* from the user-defined `vol_attr.dat` file.
- If you ran the command `watch_vols stop`, this caused the `acsls` service to be restarted by the Solaris Service Management Facility (SMF).
- GUI fixes include:
  - Selection of automatic CAP for eject.

    When starting an eject operation from the ACSLS GUI, you could correctly select a CAP which is in automatic mode (it must also be online and available). After volume and CAP selections were made, however, the eject operation failed and reported that you did not provide a CAP that was required for the operation. The same eject could be successfully performed in cmd_proc.

  - Allows vary diagnostic from GUI for drives and LSMs.

    You were able to initiate vary diagnostic operations for drives and LSMs, but the operations would fail. The GUI status screen for the operation reported a state of “terminated” and an outcome of “internal failure”. ACSLS event log messages identified the root cause as `STATUS_INVALID_VALUE` returned by database update routines. These operations should now execute as expected.

  - Fixed the 'refresh' button on CAPs table page.
  - Corrected URL in error page from Assign Logical Volumes page: `backToAssignOperationPage`.
- The Fibre Channel Read Element Status command supported by the ACSLS SMCE interface may have delivered slow performance as the number of volumes assigned to a logical library grows. This has been resolved by more efficient database access.