StorageTek Automated Cartridge System Library Software

Release Notes

Release 8.4

E64910-02

October 2015

ACSLS 8.4 introduces greater flexibility for customers with varied platform and file-system preferences. The ACSLS package installs in any file system on any contemporary Solaris 11.2 or Oracle Linux 6.5 platform.

For other ACSLS documentation, refer to the Oracle Technical Network (OTN) at:

http://docs.oracle.com/

Requirements

This section describes the software, system, browser, and co-hosting requirements.

Software Requirements

- ACSLS 8.4 has been tested and documented for:

    - Oracle's Sun SPARC and X86 platforms running Solaris 11 Update 2.

    - Oracle Linux 6.5.

        Oracle Linux testing was done in environments using Oracle's Unbreakable Enterprise Kernel.

- Other operating systems, including virtual environments, are not tested or supported.

- Special device drivers are provided in ACSLS for use with logical libraries and with fibre-attached libraries, such as the SL500 and SL150. This is an issue for Solaris zoned environments. Because such device drivers are attached to the system kernel, they must reside in the global zone. In cases where such drivers are used, ACSLS cannot be installed in the local zoned environment.

- ACSLS 8.4 High Availability (HA) systems must be installed on their own dedicated platform pair.

Operationally Approved

The product has been successfully installed to execute in the below ascribed operating environment. It has been demonstrated to provide its basic functionality without detriment to either the product or the associated execution environment.

Red Hat Enterprise Linux

System Requirements

- Memory: 4GB minimum

ORACLE®

To show system memory:

- Solaris

```
prtconf | grep Mem
```

- Linux

```
grep MemTotal /proc/meminfo
```

■ Swap: Today's systems (with Solaris 11.2 and Linux 6.5) should be equipped with no less than 4GB of memory and no less than 2GB of swap. Where system memory exceeds 6GB, a rule of thumb is to provide swap space of no less than 30% of physical memory. To check swap space on:

- Solaris

```
vmstat -S
```

The result is expressed in kilobytes.

- Linux

```
vmstat -s | grep total
```

The result is expressed in kilobytes.

■ File systems and required databases:

ACSLS 8.4 enables you to install in any file system. You must define the following directories before installing ACSLS.

- A base directory where the ACSLS components will be installed.

- A default directory for ACSLS backups. It is recommended (but not required) to place the ACSLS backup directory in a separate file system from the ACSLS base directory.

Although you can install ACSLS in any directory, the default directories used for ACSLS are:

- `/export/home` is the default ACSLS base directory.

- `/export/backup` is the default ACSLS backup directory.

The ACSLS base directory file system should have a minimum of 5GB free. An additional 5GB free should be reserved for ACSLS backups. To show filesystem sizes:

```
df -h
```

■ Fibre card (optional) is optional. A suitable HBA is required for Fibre Channel operations.

- For target mode operation, supporting the Logical Library feature, this HBA must be a contemporary QLogic fibre card (4 Gb or 8 Gb).

- For initiator mode operation, supporting a fibre-connected library such as the SL500 or SL150, ACSLS 8.4 is fully tested and certified with QLogic and Emulex HBAs.

Browser Requirements

The ACSLS 8.4 GUI can operate with most common contemporary browsers though formal testing has been limited to recent releases of FireFox, Chrome, and Internet Explorer. The Chrome browser and earlier versions of FireFox have tested well using the default settings for ACSLS in the WebLogic server. Internet Explorer V8 (and above) and FireFox V39 (and above) require configuration settings to provide a 2048-bit self-signed digital certificate for https. Refer to the section, "Configuring a Self-Signed Digital Certificate for HTTPS" in the *ACSLS 8.4 Installation Guide*.

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 standalone environment on a dedicated server. However, some systems are designed to allow multiple applications to run in co-hosted fashion as though they are completely isolated from one another. Solaris Containers and Oracle Solaris VM Server for SPARC enable conditional co-hosting possibilities for use with ACSLS.

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

- Solaris Zones (containers)

  Solaris 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 zone 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 8.4 can reside in a Solaris zone only if it does not require drivers beyond the network interface. Any use of Logical Libraries requires a target-mode fibre-channel driver, and any connection to an SL500 or SL150 library requires an initiator-mode fibre-channel driver. Either of these configurations dictates that ACSLS must be installed in the global zone.

  There is no version of ACSLS HA supported for use in Solaris zones.

- 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 buses 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 8.4 application is intended for use with logical libraries, or if you intend to connect to a fibre-channel library such as the SL500 or SL150, 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*.

Enhancements

This release contains multiple enhancements.

SL8500 Bulk CAP Support

With Bulk CAPs, the SL8500 now has two 36-cell CAPs on each rail, or 8 CAPs per library. ACSLS support for Bulk CAP includes:

- Major enhancements to the `ejecting.sh` utility as described below.

- New Dynamic Variables:

    - `BULK_CAP_EJECT_HANDLE` leaves a slot in each CAP magazine empty to be used as a handle during enters to and ejects from a Bulk CAP. The default is FALSE.

    - `ENTER_CLOSE_TO_DRIVES` enters cartridges into an SL8500 to home cells closest to the tape drives. This speeds up mounting the cartridge in the future. The default is TRUE.

    - `DISMOUNT_AWAY_FROM_DRIVES` – when cartridges are floated to a closer LSM on a dismount in an SL8500, allocates a new home cell away from the tape drives to leave room close to the drives for new cartridges. The default is FALSE.

    - `LIMIT_CAP_CONCURRENT_MOVES` limits the number of concurrent moves to or from a SL8500 Bulk CAP during an eject or an enter when ACSLS is not connected to every SL8500 in a string of connected libraries. This reserves library resources for mounts, dismounts, and other requests when doing enters and ejects to most CAPs in a string of SL8500s. The default is FALSE.

- When entering cartridges through a Bulk CAP, ACSLS tries to move the volume to a storage cell on the same side of the library as the CAP. This maximizes performance and minimizes contention between the robots on a rail.

- Customers can specify an operator panel message number describing the purpose of an enter or eject. For example: 5 for "Eject retain 1 month" or 7 for "Enter from local vault".

    Operator panel message (`opmsg`) numbers and associated descriptive text are specified using SL Console. The new `opmsg` argument is provided with the enter or eject command using `cmd_proc`. ACSLS sends the specified `opmsg` message number to the library on CAP Unlock requests. SL Console displays the message number and descriptive text when the CAP is unlocked to enter new cartridges or to remove cartridges that were ejected.

- A new `cap_type` column was added to the CAP table in the database. There are CAP type values for SL8500 rotational and Bulk CAPs and SL3000 rotational CAPs and AEMs. Display CAP types with the `display cap * -f type` command.

`ejecting.sh` Utility

- This enhancement supports small to massive vaulting operations.

- A user submits a volume-list file and specifies which CAPs to use.

- There is no limit to the number of volumes in the list.

- You can use any combination of CAPs for the operation.

- The utility directs each volume to its closest CAP for optimal performance.

- Multiple robotics are engaged for concurrent cartridge movement.

- Sequential ordering of volumes may be selected in lieu of optimized CAP selection.

- The operation completes when all volumes have been ejected

- A log summary is preserved up to ten days for each `ejecting.sh` operation.

Ability to perform mass ejects using `lib_cmd` with `eject` option

The `lib_cmd eject` command can eject a list of hundreds of volumes in a single eject. Use it to fill up an SL3000 AEM multiple times. The `ejecting.sh` utility invokes `lib_cmd eject` when ejecting volumes to an SL3000 AEM.

Security enhancements

ACSLS 8.4 includes multiple security enhancements. Some of these enhancements also improve ACSLS performance under heavy transaction loads.

Support of the XAPI interface from ELS client applications

ACSLS supports the XML API (XAPI) interface from Enterprise Library Software (ELS) client applications. This interface enables ACSLS to be a server for ELS client applications running on mainframes and other platforms.

The `watch_vols` utility can now automatically assign volumes to named scratch pools, used by XAPI clients.

Separate procedures for the installation and removal of the GUI and SMCE

During `install.sh`, choosing to install logical library support previously forced the installation of WebLogic and the ACSLS GUI, with no option or choice. Since a customer can use logical libraries using only the `lib_cmd` command-line interface (CLI), without the GUI, SMCE with logical library support can now be installed without the GUI.

WebLogic 10.3.6 bundled with ACSLS

WebLogic 10.3.6 is bundled with ACSLS.

Package `checkinstall` updates

The package install no longer aborts if users acsss, acssa, or acsdb are logged in. Instead, a warning message is posted to the installer's shell.

WebLogic start script adds TLS1 security protocol

WebLogic is now installed so that it starts with the TLS1 security protocol and with Secure Sockets Extensions enabled.

Supports automatic queuing and retry of mounts and dismounts in Fibre-attached libraries

ACSLS has supported queuing and retry of mounts and dismounts when resources are temporarily unavailable in SL8500 and SL3000 libraries for years. ACSLS now supports queuing and retry of mounts and dismounts in SCSI Media Changer libraries (such as SL150s and SL500s).

Only select read/write compatible scratch cartridges

When selecting a scratch cartridge for mount scratch, only select a cartridge that is read/write compatible with the specified tape drive.

Only report read/write compatible tape drives from `query mount scratch`

The query mount scratch command displays the drives compatible with the media in a specified scratch pool. It only displays drives that are read/write compatible with the media, excluding drives that are just read/only compatible.

Added `cap_id` to the "Remove cartridges from CAP" message

Added cap_id to the "Remove cartridges from CAP" message on cmd_proc.

New field for logical volume status

A new field for logical volume status was added to lib_cmd display volume.

Added new data field in `psacs`

The diagnostic utility psacs now includes a field for socket IDs associated with process ids and mchanger device IDs associated with device processes.

Enhanced the `get_data.sh` diagnostic utility

Enhanced the diagnostic utility, get_data.sh, to report the Linux RPM patch level.

Customer email addresses

The get_diags utility intentionally excludes configured email addresses when collecting custom configuration data from the ACSLS server.

Fixes

This release contains fixes to the following bugs reported in ACSLS 8.3 and earlier.

ACSLS Installation

The following fixes were made to ACSLS installation.

- Scratch-pad files are not cleared after completion of `db_import.sh`

  All exit points from `db_import.sh` have now been routed to routines that clear the `/tmp/acsls` directory.

- Inconsistent PostgreSQL PGPORT value

  The WebLogic installation script now verifies the current `PGPORT` to avoid problems whenever the default `PGPORT` has changed.

- The `acssa` login environment

  Upon logging in, the user `acssa` is immediately presented with the `cmd_proc` command-line interface (CLI). This is the only accessible environment available to `acssa` unless the ACSLS administrator establishes a GUI account for this user.

- Package install routine enforces system architecture checks

  This fix disallows the install if the architecture does not match.

ACSLS Startup and Shutdown

The following fixes were made to ACSLS startup and shutdown.

- ACSLS startup changes.

  The start-up logic no longer aborts if non-essential services are found to be in a *maintenance* state. The only essential services for a successful start-up sequence are `acsdb` and `acsls`. If these two services are brought online, ACSLS will be in a functional state for general library operations.

- Start-up routine loops endlessly if the database had been corrupted.

  If `acsls.startup` determines that conditions won't allow ACSLS to come up, then the start routine does not make repeated attempts to restart ACSLS.

- Archaic command in CSCI start script.

  Archaic UNIX BSD commands caused CSCI startup failure on Solaris 11.

- Fatal database error on startup causes startup display to hang with no error message.

  Fixed fatal database error on startup that caused startup display to hang with no error message.

- Report all errors loading Access Control flat files to the database.

  When the load of the Access Control flat files to the database fails, always report the reason for the failure.

- PostgreSQL logging on Linux is improperly set for debug level.

  PostgreSQL logging now reflects database transactions, but not additional dbug information.

- CAP not installed keeps LSM from coming on line automatically.

A CAP which was configured in the database but which was not installed in the library kept the LSM the CAP is in from coming online during ACSLS startup. A message now tells you to update your configuration and allows the LSM to come online.

- Prevent hang condition with start-up display when the `acsss_event.log` does not exist.

  This fix creates the `acsss_event.log` if it does not yet exist.

- On the status bar displayed during ACSLS startup, add compatibility for non-UNIX terminals such as `putty`.

  When ACSLS is starting, the status bar should display a single line with contiguous 'x' marks showing the start-up progress in the same fashion as the Solaris and Linux shell display.

- Auto-Caps are unlocked when ACSLS starts.

  Auto-CAPs that are in an online or diagnostic state are automatically unlocked during the `acsss enable` process. (Previously, auto-CAPs in SL8500 were not unlocked during ACSLS startup.)

ACSLS Logs

The following fixes were made to ACSLS logs.

- Database logs are automatically archived.

  PostgreSQL logs in the `pg_log` directory are automatically archived and compressed on all platforms.

- `rpTrail.log` is automatically archived.

  `rpTrail.log` (records all responses returned through `acslm`) is automatically archived and compressed.

Mounts and Dismounts

The following fixes were made to mount and dismount.

- Movement request fails.

  This fix now reports when a movement request fails and a cartridge is being recovered by the library.

- Erroneous messages produced when there are no more cleaning cartridges.

  When a tape drive requested cleaning but there were no compatible cleaning cartridges, other error messages were reported. Only relevant messages are now reported, and the underlying mount is performed.

- Allow `vary offline force` of `in_use` and reserved drives so they can be recovered by `vary online`.

  When a drive was in-use or reserved, it could not be varied offline force. This prevented you from easily recovering a drive that was left in an in-use or reserved status.

- More robust handling of mount and dismount operations in fibre channel libraries.

ACSLS was reporting some FC move operations as failed, when the requested operation may actually have been performed by the library. This could leave an empty drive marked as "in use" to ACSLS, requiring manual intervention (`dismount force`).

These problems are now handled and correct results are reported. The improved handling can also recover some operations that may have been interrupted by a temporary outage.

ACSLS Utilities

The following fix was made to ACSLS utilities.

- `volrpt`

  In `volrpt`, report volumes that have a home ACS or LSM that is not configured or no longer configured. This includes volumes whose home address was zeroed when they were ejected. Those volumes were not reported if an ACS 0 and LSM 0 was not configured.

Other ACSLS Processing

The following fixes were made to ACSLS processing.

- ACSLS could hang when a port scanner is run.

  This fix prevents a port scan from hanging ACSLS.

- `query drive all` could report drives multiple times when drives are assigned to logical libraries.

  When `query drive all` reports tape drives to an ACSAPI client, it filters out the drives assigned to logical libraries. This could result in reporting the same set of drives multiple times.

- Improve performance of `query server`, `query acs`, and `query lsm` with large request queues.

  When ACSLS is handling many concurrent requests, the performance of these `query` commands is now improved.

- Trim leading and trailing spaces from tape drive serial numbers.

  Spaces reported in the tape drive serial number field could cause valid serial numbers to be overlaid by spaces. This fix also trims leading and trailing spaces from drive world wide names.

- Fixed "No format for TYPE" messages on `acsss_event.log` and `cmd_proc`.

  The fix reports valid TYPE codes and IDENTIFIERs on errors to Event Notification and `cmd_proc`.

- `lib_cmd`: Support volumes with valid VOLSER characters # and $.

  Fix added `lib_cmd` support for volumes with valid VOLSER characters # and $.

- The `del_vol` utility no longer fails for VOLSERs longer than six characters.

- The `cmd_proc audit` response no longer declares 'valid' when a multi-acs audit is attempted with a single CAP.

- A `cmd_proc move` error message has been corrected.

When attempting to move a volume to an inaccessible panel, the error message now states that "the specified panel is full or the LSM is full".

- A misleading event log message has been corrected.

  After the user disables a CAP using the SL Console, ACSLS no longer suggests that the CAP will be varied offline.

Support for SCSI Media Changer Fibre-attached Libraries

The following fix was made to the SCSI mchanger Fibre-attached libraries.

- ACSLS provides better recovery of FC libraries after outages.

  When a communication outage with an FC library is detected, ACSLS now works more dynamically to reestablish a connection with the library. This is especially helpful in a Linux environment, where device paths can be volatile (the ACSLS mchanger provides persistent device paths).

  Automatic recovery of libraries now includes outages due to the reboot of an SL500 library. Previously, a manual vary operation was required in that scenario.

ACSLS GUI

The following fixes were made for the ACSLS GUI.

- GUI performance: Reduced frequency of query server polling.

  If the surrogate process is started as part of ACSLS (*true* if you install the GUI, SMCE, or `lib_cmd` features) then a query thread is launched to monitor ACSLS status periodically. The frequency has been changed from once every 11 seconds to once every minute, which reduces the number of commands executed and cuts back on database access.

- `userAdmin.sh` does not initialize on Linux. GUI users cannot be added.

  The `userAdmin.sh` utility on Linux now initializes and adds GUI users.

- Prevent `acsss_event.log` error messages after small ejects from the GUI.

  Ejects of 42 cartridges or less from the GUI would succeed, but error messages would appear in the event log and `cmd_proc` windows.

- GUI got Error 500 internal server error.

  The fix corrected the Error 500 internal server error when clicking on a CAP.

- GUI dashboard critical alarm is always illuminated when SMCE is not installed.

  The status of `smce` is now checked only if it has been installed.

ACSLS logical libraries accessed using SCSI target mode driver

The following fix was made to the SCSI target mode driver.

- QLT driver is aborted intermittently.

  If an FC initiator (client) timed out while ACSLS was still carrying out a mount or dismount operation, ACSLS could still continue carrying out the requested operation.

  Several outcomes were possible:

- The operation could complete successfully.

- The operation could fail with some error.

- The operation could eventually time out.

Once the client timed out on the operation, they experienced an outage on that medium changer device (and on other devices presented on that target port). The outage continued until ACSLS completed the mount - success, failure, or timeout. At that time the target port was reset and client operations could resume.

This problem was addressed by managing relative timeout values between the backup application and the ACSLS server presenting the logical libraries. This set of changes rolled in two COMSTAR-related updates to timeout values.

- Logical library operations appear erroneously in `lib_cmd` on Linux.

The `lib_cmd` utility was updated to mask logical library operations from the primary menu displays on Linux and on Solaris installs where logical library support has not been installed.

ACSLS HA

The following fixes were made to ACSLS HA.

- `acsls` and `acsdb` services in maintenance mode after file system is lost.

This fix applies to ACSLS HA, though the change is in ACSLS. If the file system is lost, SMF cannot disable the `acsls` and `acsdb` services and takes these services into maintenance mode. This fix provides an alternate method for SMF to disable the services, allowing the file system failure to not result in a maintenance mode condition after the services were stopped and the system failed over to the alternate node.

- HA start script fails, attempting to source the ACSLS environment.

ACSLS HA no longer sources (the non-essential) DATABASE_VERSION from `release.vars`.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.