CLI How To's

CLI (Command Line Interface) is available in CC in the following path:

..\Tlib_HD2_WEB\web\CLI\Gen4cli\TS4500CLI.jar

To run CLI you need to use command line, go to the path where you copy TS4500CLI.jar and type:

NOTE: The commands should be preceded by two '-' , i.e. --viewSystemSummary
NOTE 2: For Linux, the parameter password must be in ' ' . i.e. 'myPassword'
NOTE 3: When SSL is enable, "--ssl" flag must be added to all commands
> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –[Command] [parameters]

For "view" commands: The output will be formatted in CSV, you can save the results using redirect command in your command window.

Example:

>java -jar TS4500CLI.jar -ip 9.11.59.80 -u [userName] -p [password] –viewDataCartridges > fileName.txt

The output will be save in the file "fileName.txt". The file will be created in the same directory where you are running TS4500CLI.jar.

List of CLI commands

COMMANDS	STATUS
Data Cartridges	
-assignDataCartridges	Delivered (only Storage locations)
-bulkAssignDataCartridges	Delivered
-bulkAssignDataCartridgesByLL	Delivered
-moveFromAllDrives	Delivered
-moveFromDrive	Delivered
-moveFromIo	Removed - NOT for R1.0
-moveToDrive	Delivered
-prestageDataCartridges	Delivered
-destageDataCartridges	Delivered
-removeDataCartridges	Delivered
-rekeyCartridge	Moved to R2.0
-viewDataCartridges	Delivered
Cleaning Cartridges	
-setAutoEjectCleaningCarts	Delivered
-viewCleaningCartridges	Delivered
Volser Ranges	
-createVolserRanges	Delivered
-modifyVolserRanges	Delivered – Need changes to work as WEB action
-deleteVolserRanges	Delivered
-viewVolserRanges	Delivered
-viewVolserRangesByLL	Delivered
Logical Libraries	
-showQueuedExports	Delivered
-modifyVolserReporting	Delivered
-setMaximumVIOCartridges	Delivered
-getVIOStatus	Delivered
-modifyAdvancedEncSettings	Delivered
-viewLogicalLibraries	Delivered
-viewLogicalLibraryDetails	Delivered
-viewAdvancedEncryptionSettings	Delivered
Drives	
-resetDrive	Delivered

-cleanDrive	Delivered
-setDrivePortsID	Delivered
-modifyFibreChannelSettings	Delivered
-viewFibreChannel	Delivered
-viewDriveDetails	Delivered
-viewDriveSummary	Delivered
-viewDriveVPD	Delivered
Key Label Mapping	
-createKeyLabelMapping	Delivered
-editKeyLabelMapping	Delivered
-deleteKeyLabelMapping	Delivered
-viewKeyLabelMapping	Delivered
Encryption Key Managers / CEP & BEP	
-createBEP	Delivered
-modifyBEP	Delivered
-deleteBEP	Delivered
-viewBEP	Delivered
System	
--viewIoStation	Delivered
--viewNodeCards	Delivered
--viewLibraryVPD	Delivered
--viewSystemSummary	Delivered
--viewSystemSummaryDetails	Delivered
--viewAccessor	Delivered
--resetNodeCards	Delivered
--setLibraryTime	Delivered
--setNMADetection	Delivered
--saveConfiguration	Delivered
--restoreConfiguration	Delivered
Users & Roles	
--viewUsers	Delivered
-viewRoles	Delivered
-viewRolePermissions	Delivered

-setRolePermissions	Delivered
Security	
Enable all commands to run using RA	
-setSSL	Delivered
-setSSLForEKM	Delivered
Service	
-downloadLog	Delivered
-codeupdate	Delivered
-driveCodeUpdate	Delivered
-downloadPropertiesFile	Delivered
-viewSnapshots	Delivered
-downloadSnapshot	Delivered
-setScannerSpeed	Delivered
--downloadDrivesLog	Delivered
--setSlotOnline	Delivered
--setSlotOffline	Delivered
--viewOfflineComponents	Delivered
--startLibraryVerify	Delivered
--continueLibraryVerify	Delivered
--continueCloseLibraryVerify	Delivered
CLI	
-version	Delivered
-batch	Delivered
-verbose	Delivered
-ssl	Delivered
MANUFACTURING / INTERNAL USE ONLY	
-testMoveSlot	Delivered
-getFWVersion	Delivered
-enableSSH	Delivered
-setVIOStatus	Delivered
-sendPMR	Delivered
-installLicensedFunction	Delivered
-sendMessage	Delivered
-startDiscoverHW	Delivered
-startInventory	Delivered

-startCalibration	Delivered
-createLL	Delivered
-deleteLL	Delivered
-assignDriveToLL	Delivered
-setSerialNumber	Delivered
-unassignDrive	Delivered
-setSleep	Delivered
-setBaseWWNN	Delivered

Command Name:
`assignDataCartridges`

Description:
Assigns data cartridges to a logical library.
To move a data cartridge currently in I/O, and Virtual I/O enabled, you should use this command.

Parameters:
FileName.
Each line in the file will be interpreted as an assign operation.

The input file would take the following format:

From a storage location (even if tier = 0, you have to use the four values):

[F,C,R,T], Logical Library name

From a I/O station

[F,R], Logical Library name

There is also support for VOLSER instead of FCRT. The format will be as follows:

[VOLSER] (8 characters),Logical Library name

Expected output:

In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong number of parameters. *****
[F,C,R,T], Logical Library Name.
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
*****************************************************************

Moving to next element...
```

When the command is executed successfully then the message will be:

```
"The cartridge [FCRT] has been assigned successfully"
```

This command receives as input, a text file with one or more assigns, once all of the assigns are processed a message indicating the end of the whole process will be shown.
`"Done."`

Example:

fileName.txt will contain:

F1,C2,R8,T0, testltolib

F1,C3,R18,T5, lib2

---

with VOLSER:

E373NTJR,JAGR_C

JJS235JJ,JAGR_C

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –assignDataCartridges fileName.txt

---

Command Name:
`bulkAssignDataCartridges`

Description:
Assigns multiple cartridges at the same time instead of one at a time. The cartridges will be assigned in bulks of size 35 as maximum. The bulk will be assigned to a same logical library.

Parameters:
FileName., Logical Library Name
Each line in the file will be interpreted as a assign operation.

The input file would take the following format:

[F,C,R,T] #1

[F,C,R,T] #2

There is also support for VOLSER instead of FCRT. The format will be as follows:

[VOLSER] (8 characters)

Expected output:
In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong format for cartridge: [location/volser] *****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next bulk...
```

When the command is executed successfully then the message will be:

```
"The cartridges have been assigned successfully"
```

This command receives as input, a text file with one or more assigns, once all of the assigns are processed a message indicating the end of the whole process will be shown.
`"Done."`

Example:

fileName.txt will contain:

F1,C2,R8,T0

F2,C2,R8,T0

F3,C2,R8,T0

F1,C3,R18,T5

Or

A00023JA
A00024JA
A00025JA

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --bulkAssignDataCartridges fileName.txt, LibTest

---

Command Name:
`bulkAssignDataCartridgesByLL`

Description:
Assigns multiple cartridges at the same time instead of one at a time. This command allows to assign data cartridges to different logical libraries (different destiny LL). The cartridges will be assigned in bulks of size 35 as maximum.

Parameters:
FileName.
Each line in the file will be interpreted as a assign operation.

The input file would take the following format:

[F,C,R,T] #1, Logical Library name

[F,C,R,T] #2, Logical Library name

There is also support for VOLSER instead of FCRT. The format will be as follows:

[VOLSER] (8 characters),Logical Library name

Logical Library name can be the different for each of the entries.

Expected output:
In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong format for cartridge [location / volser]*****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next bulk...
```

When the command is executed successfully then the message will be:

```
"The cartridges have been assigned successfully"
```

This command receives as input, a text file with one or more assigns, once all of the assigns are processed a message indicating the end of the whole process will be shown.
`"Done."`

Example:

fileName.txt will contain:

F1,C2,R8,T0, testltolib1

F2,C2,R8,T0, testltolib2

F3,C2,R8,T0, testltolib3

Or

A00034JA, testltolib1

A00035JA, testltolib2

A00036JA, testltolib3

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –bulkAssignDataCartridgesByLL fileName.txt

---

Command Name:
`createVolserRanges`

Description:
Creates new volser ranges.
Parameters:
FileName.
Each line in the file will be interpreted as a volser range. The parameters are:

startRange,endRange,LLName,mediaType[LTO,3592],[TRUE/FALSE] (flag to enable this range only for new cartridges).

Expected output:

In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong number of parameters. *****
starRange,endRange,LL name, media Type [LTO,3592], Enable [true/false].

[startRange,endRange] are the only values that cannot be changed. *****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next element...
```

When the command is executed successfully then the message will be:

```
"The Volser Range was created successfully"
```

This command receives as input, a text file with one or more volser ranges, once all of the ranges are processed a message indicating the end of the whole process will be shown.
`"Done."`
Example:

fileName.txt will contain:

UAA9RH,UBB9RH,Library 1,3592,FALSE

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –createVolserRanges fileName.txt

---

Command Name:
`modifyVolserRanges`

Description:
Allows to modify **Logical Library, media Type and the flag** to enable the volser range only for new cartridges.

Parameters:
FileName.
Each line in the file will be interpreted as a volser range. The parameters are:

startRange*,endRange*,LLName,mediaType [LTO,3592],[TRUE/FALSE] (flag to enable only for new cartridges).

* : These values are used to get the range index in the table, so these values cannot be modified.

Expected output:

In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong number of parameters. *****
starRange,endRange,LL name, media Type [LTO,3592], Enable [true/false].

[startRange,endRange] are the only values that cannot be changed. You must included
all the parameters event if they are not changing.*****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next element...
```

When the command is executed successfully then the message will be:

```
"The Volser Range was modified successfully"
```

This command receives as input, a text file with one or more volsers, once all of the volsers are processed a message indicating the end of the process will be shown.
`"Done."`

Example:

fleName.txt will contain:

UAA9RH,UBB9RH,Library 1,3592,TRUE

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –modifyVolserRanges fileName.txt

---

Command Name:
`deleteVolserRanges`

Description:
Deletes all the volser ranges with the file.

Parameters:
FileName.
Each line in the file will be interpreted as a volser range. The parameters are:

Logical Library Name,startRange,endRange

Expected output:

In case the input file has a wrong format an error message will be shown:

```
"Error: Wrong format for Volser Range:

[LogicalLibraryName],[volserStart],[volserEnd]”
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next element...
```

When the command is executed successfully then the message will be:

```
"The Volser Range was removed successfully"
```

This command receives as input, a text file with one or more volsers, once all of the volsers are processed a message indicating the end of the process will be shown.
`"Done."`

Example:

fileName.txt will contain:

Library 1,TUU9RH,TXX9RH

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –deleteVolserRanges fileName.txt

---

Command Name:
`viewVolserRanges`

Description:
Shows all the volser ranges in the physical library.

Parameters:
None

Expected output:

Volser Start,	Volser End,	Logical Library,	Media Type,	Number of Cartridges
CCC9RH,	DDD9RH,	Library 1,	LTO,	40
EEE9RH,	FFF9RH,	Library 1,	LTO,	10
GGG9RH,	HHH9RH,	Library 1,	3592,	10
III9RH,	JJJ9RH,	Library 1,	3592,	10

*Sort by VOLSER range

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewVolserRanges

Command Name:
`viewVolserRangesByLL`

Description:
Shows all the volser ranges assigned to each logical library.

Parameters:
None

Expected output:

Logical Library,	Cartridges,	Media Type
Tuco_Salamanca,	14,	LTO
SG0011 - SG1100,	14	
Typoon,	1,	LTO
B00000 - B00001,	0	
D00000 - D00030,	0	

*Primary sort by Logical Library

*Secondary sort by VOLSER range

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewVolserRangesByLL

Command Name:
viewLogicalLibraries

Description:
Lists all the logical libraries.

Parameters:
None

Example of output:

Name, Type,Assigned Cartridges, Virtual I/0 cartridges, Drives, Encryption Method, Queued Exports,VOLSER Reporting (6/8 characters)

GVGNFLOW, 3592, 0, 0, 0, No , Hide, 6

Library 1, 3592, 0, 0, 0, No , Show, 6

* Sort by Name.

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewLogicalLibraries

NOTE:
Queued Exports,VOLSER Reporting (6/8 characters) are CLI only columns.

Command Name:
viewLogicalLibraryDetails

Description:
Shows the details of the given logical library.

Parameters:
Logical library name

Example of output:

Name, Type, Max Cartridges, Virtual I/0 slot addresses,Drive Addresses,

Library_1, 3592, 1000, 130(768-897), 68(257-321)

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewLogicalLibraryDetails Library_1

Command Name:
`showQueuedExports`

Description:
Set flag to Show/Hide (true, false) queued exports.

Parameters:
Logical library name, Flag value [True/false] : SHOW = TRUE, HIDE = FALSE

Example of output:

In case the input parameters have a wrong format an error message will be shown:

```
"***** ERROR: Wrong number of parameters. *****
[LL name],[Value]
```

In case Logical Library is missing:

```
***** ERROR: '' not a valid logical library *****
```

In case second parameter is different from the accepted values:

```
***** ERROR: 'value' not a valid value *****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"Queued exports flag  was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] --showQueuedExports Library_1,TRUE

Command Name:
`modifyVolserReporting`

Description:
Modifies Volser Reporting value to host (6 or 8) .

Parameters:
Logical library name, Flag value [6/8]

Example of output:

In case the input parameters have a wrong format an error message will be shown:

```
"***** ERROR: Wrong number of parameters. *****

[LL name],[Value]
```

In case Logical Library is missing:

```
***** ERROR: '' not a valid logical library *****
```

In case second parameter is different from the accepted values (6,8):

```
***** ERROR: 'value' not a valid value *****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
""Volser Reporting flag  was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] --modifyVolserReporting Library_1,6

Command Name:
`setMaximumVIOCartridges`

Description:
Update the number of Virtual I/O slots for the logical library.

Parameters:
The lower limit of the accepted values is the number of physical IO slots. If the library has two IO magazines, each with 18, the valid range of values will be : [36-255]

Logical library name, value [Total number of IO slots - 255]

Example of output:

In case value is different from the accepted values:

```
"***** ERROR: Wrong value. Parameters need to be inside quotation marks: [LL name], [Value]  *****
***** where Value = [32-255]  *****
```

In case any parameter is missing or is different from the accepted values:

```
***** ERROR: Missing parameters (need to be inside quotation marks ): [LL name], [Value] *****
***** where Value = [32-255]  *****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"The MAX VIO was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --setMaximumVIOCartridges Library_1,200

Command Name:
downloadLog

Description:
Downloads the zip file with all the logs.

Parameters:
-
Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
There was a problem dowloading logs. Try again
```

When the command is executed successfully a message indicating the completed percentage will be shown:

```
Downloading... [completed] / [totalSize] has been downloaded.
```

When the file has been downloaded:

```
The log file: "TS4500_FWLOGS_[yyyy-MM-dd_hh.mm.ss].zip has been downloaded.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –downloadLog

---

Command Name:
downloadPropertiesFile

Description:
Downloads the properties file (TSSC file) if exists in the card.

Parameters:
-
Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
There was a problem dowloading logs. Try again
```

When the command is executed successfully a message indicating the completed percentage will be shown:

```
Downloading... [completed] / [totalSize] has been downloaded.
```

When the file has been downloaded:

```
The file: ""LIBLG_01_VP_[yyyy-MM-dd_hh.mm.ss] has been downloaded.
```

If the file cannot be found:

```
File cannot be found
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --downloadPropertiesFile

Command Name:
`viewDataCartridges`

Description:
Shows all the data cartridges in the library

Parameters:
-

Example of output:

```
 Volume Serial,     Logical Library,Element Address,            Media Type,
Location(F,C,R),       Encryption,            Most Recent use
      M000NTJL,   Logical Library 1,          1046,                JAG,        Slot(F1,C5,R4,T1),
Not Encrypted,                    0
      M001NTJL,   Logical Library 1,          1294,                JAG,        Slot(F1,C6,R29,T0),
Not Encrypted,
```

*Sort by Volume serial (8 characters long).

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --viewDataCartridges

---

Command Name:
`viewCleaningCartridges`

Description:
Shows all the cleaning cartridges in the library

Parameters:
-

Example of output:

```
 Volume Serial,     Logical Library,Element Address,            Media Type,
Location(F,C,R),       Cleans remaining,            Most Recent use
      M000NTJL,   Logical Library 1,          1046,                JAG,        Slot(F1,C5,R4,T1),
50,                        31 December 1969 06:00:00
      M001NTJL,   Logical Library 1,          1294,                JAG,        Slot(F1,C6,R29,T0),
50,                        31 December 1969 06:00:00
```

*Sort by Volume serial (8 characters long).

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --viewCleaningCartridges

---

Command Name:

---

viewDriveSummary

Description:
Shows all the drives in the library
Parameters:
-
Example of output:

Location(F,C,R), State, Type, Contents, Firmware, Serial, WWWNN, Element Address, Logical Library

F1,C1,R3, Online, 3592, Empty, 0, 13A0001, , 257, GVGNFLOW

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewDriveSummary

---

Command Name:
viewDriveDetails

Description:
Shows details of the given drive

Parameters:
-f [frame] -c [column] -r [row]
Example of output:

Location(F,C,R): F1,C1,R3
State: Online
Type: 3592
Logical Library: GVGNFLOW
Control path: Enabled
Contents: Empty
Firmware:
WWNN:
Element Address: 257
Drive Display: ND
Port 0
Loop ID: 0
WWPN:
Port 1
Loop ID: 0
WWPN:

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewDriveDetails -f 1 -c 1 -r 3

Command Name:
viewNodeCards

Description:
Shows information on the node cards

Parameters:
-

Example of output:

Card name, CardPass, Location, Part number, Serial number, Firmware version
LCA 1, 0, Frame 1.Row 1, , , 1.1.0.0-00Q.00

*Sort by Card name

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewNodeCards

---

Command Name:
viewSystemSummary

Description:
Views the physical library system summary. It will display information about each frame in ascending order, and at the end a summary with total numbers on the whole library will be displayed.

Parameters:
-

Example of output:

```
                Frame:          1
                State:          Door closed
           Media Type:          3592
      Data Cartridges:          307
       Storage Slots:           660
               Drives:          10
  Upper IO cartridges:          9
   Uppper IO door slots:        16
Uppper IO door media type:      3592
  Lower IO cartridges:          7
  Lower IO door slots:          16
 Lower IO door media type:      3592
                MTM:            3584L25
                S/N:            FA002
------------------------------------------------------------------
                Frame:          2
             Accessors:         OK – Online
                State:          Door closed
           Media Type:          3592
      Data Cartridges:          332
       Storage Slots:           883
               Drives:          2
                MTM:            3584D25
                S/N:            FA106
------------------------------------------------------------------
    Total storage slots:        1543
 Total storage cartridges:      639
       Total IO slots:          16
    Total IO cartridges:        16
       Total Drives:            12
       Total Frames:            2
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewSystemSummary

---

Command Name:
viewSystemSummaryDetails

Description:
View physical library system summary for a particular frame

Parameters:
Frame number

Example of output:

In case the input frame is not available in the library, the following message will be displayed:

```
<Empty>
```

When the frame is found in the database the following information will be displayed:

Frame:	2
Accessors:	OK – Online
State:	Door closed
Media Type:	3592
Data Cartridges:	332
Storage Slots:	883
Licensed slots:	200
LTO Capacity:	
3590 capacity:	
Drives:	2
MTM:	3584D25
S/N:	FA106

Accessors information will be available depending on the type of frame.

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –
viewSystemSummaryDetails 2

Command Name:
`getVIOStatus`

Description:
Shows the status of the VIO flag on the library.

Parameters:
None

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"VIO flag is: [Enabled | Disabled]"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --getVIOStatus

---

Command Name:
`viewIoStation`

Description:
Shows all the cartridges in IO station

Parameters:
-

Example of output:

Volume Serial, Logical Library,Element Address, Media Type, Location(F,C,R), Encryption
M000NTJL, Logical Library 1, 1046, JAG, I/O Slot(F1,C5,R4,T1), Not Encrypted
M001NTJL, Logical Library 1, 1294, JAG, I/O Slot(F1,C6,R29,T0), Not Encrypted

*Sort by Volume serial (8 characters long).

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --viewIoStation

---

Command Name:
viewFibreChannel

Description:
Shows the Fibre Channel settings for both Ports in each drive.

Parameters:
-

Example of output:

Drive, Location(F,C,R), Logical Library, Type, Port, Link Status, Configured Link Speed, Configured Topology
30F000701, F1,C1,R1, Library1, 3592, 30F400701, Light Detected, 4 Gb/s, L Port
, , , , , 30F800701, No Light, Not Available, Not Available
30F000702, F1,C2,R1 Library2, 3592, 30F400702, Light Detected, 4 Gb/s, L Port
, , , , , 30F800702, No Light,Not Available, Not Available

*Sort by Drive (this columns shows drive's WWWNN)

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --viewFibreChannel

---

Command Name:
viewDriveVPD

Description:
Shows VPD information for all drives

Parameters:
-

Example of output:

Location(F,C,R),	Drive Type,	Firmware Version,	Machine Type,	Serial Number
F1,C2, R1	LTO 5	C7R2	3588	00078AE1DD
F1,C1,R2	LTO 5	C7R2	3588	00078AE1F3
F4,C4,R1	3592E04	2693	3592	0007859653

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --viewDriveVPD

Command Name:
viewLibraryVPD

Description:
Shows the library VPD information

Parameters:
-

Example of output:

```
Location,        Machine Type,     Model,       Serial Number,Media Type
 Frame 1,              3584,       L25,            13FA005,      3592
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] --viewLibraryVPD

Command Name:
modifyAdvancedEncSettings

Description:
Modifies the advanced encryption settings for a Logical Library. The advanced fields are: Advanced method, advanced policy, density code and key path.

Parameters and possible values:

Logical Library Name, Advanced Method [TRUE / FALSE], Advanced Policy*, Density code**, Key path***

*Advanced policy can take any of these possible values:

- No advanced setting = 0
- Don't encrypt if no policy = 1
- Encrypt if no policy = 2
- Policy required = 3
- Never encrypt (policy override) = 4
- Always encrypt (policy override) = 5
- Internal label – Selective Encryption = 6
- Internal label – Encrypt all = 7

**Density code can take any of these possible values:

- No advanced setting = 0
- Shows Encryption = 1
- Masks Encryption = 2

***Key path can take any of these possible values:

- No advanced setting = 1
- System = 2
- Application (IBM) = 3
- Application (T10) = 4
- Library = 6

Example:

Library1,TRUE,2,2,2

Example of output:

In case the input parameters have a wrong format an error message will be shown:

```
***** ERROR: Wrong number of parameters. *****
             Logical Library Name,
                   Advanced Method: [TRUE / FALSE],
                   Advanced Policy:
                   No advanced setting                          = 0
                   Don't Encrypt if No policy                   = 1
                   Encrypt if no policy                         = 2
                   Policy required                              = 3
                   Never encrypt ( policy override)             = 4
                   Always encrypt ( policy overrride)           = 5
                   Internal Label - Selective Encryption        = 6
                   Internal Label - Encrypt All                 = 7
            Density code:
```

```
                No advanced setting                              = 0
                Shows Encryption                                 = 1
                Masks Encryption                                 = 2

            Key Path:
                No advanced setting                              = 1
                System                                           = 2
                Application (IBM)                                = 3
                Application (T10)                                = 4
                Library                                          = 6

Advanced Policy [0-7], Density code [0-2], KeyPath[1,2,3,4,6].*****
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

In case any of the parameters have a value out of the accepted values a message indicating the error will be shown as long as the corresponding list of accepted values.

When the command is executed successfully then the message will be:

```
"The advanced encryption settings for Logical Library [NAME] were updated
successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –modifyAdvancedEncSettings LLName,Advanced Method [TRUE / FALSE], Advanced Policy[0-7], Density code[0-2], Key path[1,2,3,4,6]

Command Name:
viewAdvancedEncryptionSettings

Description:
Shows the advanced Encryption settings of the given logical library

Parameters:
Logical Library name

Example of output:

Name:	LibLog1
Advanced Method:	True
Advanced Policy:	Encrypt if no policy
Density Code:	Shows Encryption
Key Path:	System

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –
viewAdvancedEncryptionSettings LibLog1

Command Name:
`resetNodeCards`

Description:
Resets all specified node cards

Parameters:
A comma separated list of the nodes to be reset. Accepted values are:

```
ALL,ALLLCA,ALLACC,ALLSMC,ALLXYC,ALLPLUS,XYCPLUS,ACCPLUS,LCAPLUS,XYA,XYB,OPC1,SMC[2-16],ACCA,ACCB,LCA[1-31],LCAB[1-31]
```

Example of output:

In case the input parameters are not in the list , this message will be shown:

```
"There are no valid node cards to reset"
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"The command was executed successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --resetNodeCards LCA1

Command Name:
codeUpdate

Description:
Updates library firmware

Parameters:

Filename of Fw image.

Example of output:

In case the file doesn't exist , this message will be shown:
"The file cannot be found. The update was cancelled."

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

"Done"

If everything is OK the following messages will be shown:

```
Name: TS4500_1100-04Q.afwz
>>>Uploadling file..
.....................................................
Done.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –codeUpdate TS4500_1100-O4Q.afwz

Command Name:
`driveCodeUpdate`

Description:
Updates drive firmware

Parameters:

Filename of Fw image and list of drives to be updated. If the file is not in the same directory where CLI is running you need to include the whole path, i.e: C:\myPath\driveCode.afwz
The list of drives has this format:
F#C#R#,F#C#R#,... . The list must be preceded by "-l" (L in lowercase stands for Location)

There are "," to separate drives, but not to separate frame,column and row

--reset : Possible values are : IMMEDIATE, UNLOAD, MANUAL

Example of output:

In case the file doesn't exist , this message will be shown:

```
"The file cannot be found. The update was cancelled."
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"The code update has started
```

If everything is OK the following messages will be shown:

```
Name: LTO6_DAQM.fcp_fh.fmrz
>>>Uploadling file..
.....................................................
Done.
The code update has started.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –driveCodeUpdate LTO6_DAQM.fcp_fh.fmrz -l F1C4R2,F1C4R3 --reset IMMEDIATE

Command Name:
`moveToDrive`

Description:
Mounts a cartridge to the given drive

Parameters:

For the cartridge:

There is support for physical position
F1,C2,R2,T1
and VOLSER
A00045JA

For drive:
-f 1 -c 3 -r 1

Example of output:
In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The cartridge [volser] was mounted successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –moveToDrive F1,C5,R1,T1 -f 1 -c 4 -r 1

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –moveToDrive A00045JA -f 1 -c 4 -r 1

Command Name:
`moveFromDrive`

Description:
Ejects a cartridge from a drive

Parameters:

For drive:
-f 1 -c 3 -r 1

Example of output:
In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The cartridge was ejected successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –moveFromDrive -f 1 -c 4 -r 1

Command Name:
`destageDataCartridges`

Description:
This command is only for HD libraries. Destages cartridges based on a list of locations in a file.

Parameters:

file name

The file should contain a line for each cartridge in this format:
F1,C2,R4,T0
F2,C5,R2,T0
F1,C4,R8,T0
F1,C2,R6,T0

VOLSER is also accepted

A00045JA
A00046JA
A00047JA

we can also send the volser list in the command line

NOTE: The cartriges must be in TIER = 0

Example of output:
In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The cartridge [location] was moved successfully. Next element.."`

When all cartridges have been processed:
`"Done.”`

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –destageDataCartridges moves.txt

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] – destageDataCartridges A00000JA,A00001JA,A00002JA

Command Name:
`prestageDataCartridges`

Description:
This command is only for HD libraries. Prestages cartridges based on a list of locations in a file.

Parameters:

file name
The file should contain a line for each cartridge in this format:
F1,C2,R4,T3
F2,C5,R2,T3
F1,C4,R8,T4
F1,C2,R6,T4

VOLSER is also accepted

A00045JA
A00046JA
A00047JA

We can also send the volser list in the command line.

NOTE: Cartridges must be in TIER >= 3.
Example of output:
In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The cartridge [location] was moved successfully. Next element.."`

When all cartridges have been processed:
`"Done.”`

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –prestageDataCartridges moves.txt

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –prestageDataCartridges A00000JA,A00001JA,A00002JA

Command Name:
`moveFromAllDrives`

Description:
Ejects all cartridges

Parameters:

None

Example of output:
For each drive that is not empty one of these messages will be shown:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The drive " + location + " is empty now. Moving to next drive..."
```

When all drives have been emptied:

```
"Done. All drives are empty."
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –moveFromAllDrives

Command Name:
`resetDrive`

Description:
Power cycles a given drive

Parameters:

For drive:
-f 1 -c 3 -r 1

Example of output:
In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The drive was reset successfully"`

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –resetDrive -f 1 -c 4 -r 1

Command Name:
`cleanDrive`

Description:
Cleans a given drive

Parameters:

For drive:
-f 1 -c 3 -r 1

Example of output:
In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The drive was cleaned successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –cleanDrive -f 1 -c 4 -r 1

Command Name:
viewUsers

Description:
Displays a list of all the users in the library.

Parameters:

None

Example of output:

Name , Locked, State, Role, Email, Last login
jecervan, , Disconnected, Administrator, jecervan@mx1.ibm.com,

service, , Disconnected, Service, service@service.com,

temporal, , Disconnected, Administrator, ,
lunes, noviembre 25, 2013, 09:40:22 AM CST

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewUsers

Command Name:
`viewRoles`

Description:
Displays a list of all the defined roles in the library.

Parameters:

None

Example of output:

Role,	Mapped users
Administrator,	12
Superuser,	0
Service,	1
Monitor,	1

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewRoles

Command Name:
viewRolePermissions

Description:
Displays a list of all the permissions for a given role. The output can be display in the command window or save to a text file using ">". When user save the output the file can be use in "**setRolePermissions**" command. It will show a list of the web pages available in the library and its corresponding access level.

Parameters:
Role name

Example of output:

Action,	Access Level
Cartridges,	No Access
Cartridges by Logical Library,	No Access
Cleanning Cartridges,	No Access
Drives,	No Access
Drives by Logical Library,	No Access
Email Notifications,	Read Only
Email Recipients,	Read Only
Encryption Internal,	Read Only
Encryption Key Manager,	Read Only
Ethernet Ports,	No Access
Events,	No Access
Library Information,	No Access
Licensed Functions,	No Access
Logical Libraries,	Read Only
Management GUI Behavior,	No Access
Master Console,	No Access
Password Rules,	Read Only
Remote Authentication,	Read Only
Roles,	No Access
SNMP Destinations,	Read Only
SNMP Notifications,	Read Only
Scan Speed,	Modify
Secure Socket Layer,	Read Only
Service Port,	Modify
Syslogs Notifications,	Read Only
System,	Modify
System Date and Time,	Modify
Tasks,	Modify
Users,	No Access
VOLSER Ranges,	Read Only
VOLSER Ranges by Logical Library,	Read Only

How to run:

Saving output to a text file:
>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewRolePermissions [roleName] > rolePermissions.txt

Displaying output in command window
>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewRolePermissions [roleName]

Command Name:
`setRolePermissions`

Description:
This command is used with “viewRolePermissions” output. The user will be able to change the existing permissions for a given role using the “viewRolePermissions” output saved in a text file. The possible permissions are:
“Read only”: User will have only read access to the web page.
“Modify”: User can execute edit operation in the web page.
“No Access”: User won't be able to see the web page.

The output file from “viewRolepermisisons” can be edited to change any of the access levels for the listed pages, using the above values (Read only, modify, no access).
This new file will be used as a parameter for “setRolePermissions” command. The new values will be set for the given role.

Parameters:

fileName.txt
-role Role name

Example of output:

In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The permissions were updated successfully"`

How to run:

Example input file:
The file can contain all of the pages or just the ones that will change.

Action,	**Access Level**
Cartridges,	No Access
Cartridges by Logical Library,	No Access
Cleanning Cartridges,	No Access
Drives,	No Access
Drives by Logical Library,	No Access
Email Notifications,	Read Only
Email Recipients,	Read Only
Encryption Internal,	Read Only
Encryption Key Manager,	Read Only

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setRolePermissions rolePermissions.txt -role Service

Command Name:
`setNMADetection`

Description:
Sets ON/OFF the NMA detection flag in the library.

Parameters:
flag value:
TRUE for ON
FALSE for OFF

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The NMA detection flag was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setNMADetection TRUE

Command Name:
`setLibraryTime`

Description:
Sets the library date and time to that of the server on which the CLI is running

Parameters:
None

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The library date and time were updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setLibraryTime

Command Name:
`setDrivePortsId`

Description:
Sets the ports Ids for a given drive (Port 1 and Port 2).

Parameters:
`IDs [Port1ID,Port2ID]  -f [frame] -c [column] -r [row]`

Port 1 ID and Port 2 ID can take values between [0,127]

Example of output:

In case the parameters are incorrect:
When FCR are missing:
```
***** ERROR: Frame, Column and Row number must be included *****
```

When ids are missing or if the values are out of range:
```
***** ERROR: Wrong number of parameters. *****
[Port 1 ID, Port 2 ID]
 Accepted values are in the range [0,127]
```

In case there is a problem with the command execution:
```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The ports IDs were updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setDrivePortsID 18,21 -f 1 -c 1 -r 1

Command Name:
`modifyFibreChannelSettings`

Description:
Sets ports speed and topology for the given drive. Speed and topology will be the same for both ports.

Parameters:
`[Speed ,Topology] -f [frame] -c [column] -r [row]`

For Speed valid values are: Auto,1,2,4,8. The values are in Gb/s.
For Topology valid values are: Auto-L,L,N,Auto-N

Example of output:

In case the parameters are incorrect:
When FCR are missing:
```
***** ERROR: Frame, Column and Row number must be included *****
```

When parameters are missing or the values are invalid:
```
***** ERROR: Wrong number of parameters. *****
[Port Speed, Port Topology]
where: Port1 Speed, Port2 Speed (Gb/s): Auto,1, 2, 4 ,  8
Port1 Topology,Port2 Topology: Auto-L, L, N, Auto-N
```

In case there is a problem with the command execution:
```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The Fibre channel settings were updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --modifyFibreChannelSettings 4,N -f 1 -c 1 -r 1

Command Name:
`viewSnapshots`

Description:
View the list of available snapshots.

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Cannot retrieve data ***
```

When there is no snapshots available:

```
"There are no snapshots available."
```

When the command is executed successfully a list in ascending order (creation timestamp) will be shown:
1 – TS4500_Snapshot_01122013.zip
2 – TS4500_Snapshot_01012014.zip
3 – TS4500_Snapshot_01022014.zip

*These indexes will be used in “downloadSnapshot” command

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [username] -p [password] --viewSnapshots

---

Command Name:
`downloadSnapshot`

Description:
Downloads the selected snapshot. This command must be used after “viewSnapshots”

Parameters:
Index of the snapshot that will be downloaded
This index must be taken from the results of “viewSnapshots”

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
**** There was a problem. The file cannot be found or is empty.
```

When the command is executed successfully a message indicating the completed percentage will be shown:

```
Downloading... [completed] / [totalSize] has been downloaded.
```

When the file has been downloaded:

```
The snapshot file: "TS4500_FWLOGS_[yyyy-MM-dd_hh.mm.ss].zip has been downloaded.
```

* The file is downloaded to the same directory where CLI is running.
How to run:

```
>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --downloadSnapshot [Index]
```

Command Name:
`viewKeyLabelMapping`

Description:
View the list of available key labels.

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Cannot retrieve data ***
```

When the command is executed successfully a list in ascending order (creation order) will be shown:

Index,	Map From Key Label,	Key Mode ,	Map To Key Label
1,	Aedea,	Wrapped-Default,	0
2,	Aedea2,	Wrapped-Default,	0
3,	Aedea3,	Wrapped-Hash,	Aedea5

***The values in the Index column will be used in "editKeyLabelMapping" and "deleteKeyLabelMapping" commands**

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [username] -p [password] –viewKeyLabelMapping

Command Name:
`createKeyLabelMapping`

Description:
Create a key label mapping

Parameters:
keyLabelFrom: This is a string that cannot be repeated in the list of key labels.
KeyModeFrom: The possible values are: **`[Wrapped-Hash, Wrapped-Default, Wrapped-Clear,Direct-Default-Set,Direct-Specific].`** When using WRAPPED-DEFAULT keyLabelTo is disabled.
KeyLabelTo: This is a string.
The max length of keyLabelFrom and keyLabelTo is 50 characters.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When parameters are missing or the values are invalid:

```
***** ERROR: KLM parameters in wrong format. Required parameters should be separated by comma:  *****
*****        [KeyLabelFrom,KeyMode,KeyLabelTo]  *****
```

Invalid mode:

```
***** ERROR: KLM parameter: Key Mode  not allowed. Mode types accepted are:  *****
*****        [Wrapped-Hash, Wrapped-Default, Wrapped-Clear]  *****
```

When using Wrapped-Default mode:

```
***** NOTE: KeyLabelTo will be ignored.  *****
*****       When using WRAPPED-DEFAULT keyLabelTo is disabled  *****
```

If the keyLabelFrom already exists:

```
****ERROR: The name for Map From Key label already exists.
```

When the command is executed successfully:

```
The Key Label Mapping was created successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --createKeyLabelMapping START,Wrapped-Hash,END

Command Name:
`editKeyLabelMapping`

Description:
Edit a key label mapping. You need to execute "viewKeyLabelMapping" command before, to get the index of the key label mapping that will be edited.

Parameters:
Index: You need to execute "viewKeyLabelMapping" to get this value.
keyLabelFrom: This is a string that cannot be repeated in the list of key labels.
KeyModeFrom: The possible values are: **`[Wrapped-Hash, Wrapped-Default, Wrapped-Clear,Direct-Default-Set,Direct-Specific].`** When using WRAPPED-DEFAULT keyLabelTo is disabled.
KeyLabelTo: This is a string.
The max length of keyLabelFrom and keyLabelTo is 50 characters.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When parameters are missing or the values are invalid:

```
***** ERROR: KLM parameters in wrong format. Required parameters should be separated by comma:  *****
*****        [KeyLabelFrom,KeyMode,KeyLabelTo]  *****
```

When the index cannot be found

```
****ERROR: The key label was not found.
```

Invalid mode:

```
***** ERROR: KLM parameter: Key Mode  not allowed. Mode types accepted are:  *****
*****        [Wrapped-Hash, Wrapped-Default, Wrapped-Clear]  *****
```

When using Wrapped-Default mode:

```
***** NOTE: KeyLabelTo will be ignored.  *****
*****       When using WRAPPED-DEFAULT keyLabelTo is disabled  *****
```

If the keyLabelFrom already exists:

```
****ERROR: The name for Map From Key label already exists.
```

When the command is executed successfully:

```
The Key Label Mapping was created successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --editKeyLabelMapping 1,START,Wrapped-Hash,END2

Command Name:
`deleteKeyLabelMapping`

Description:
Deletes a key label mapping. You need to execute "viewKeyLabelMapping" command before, to get the index of the key label mapping that will be deleted

Parameters:
Index: You need to execute "viewKeyLabelMapping" to get this value.
.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the index cannot be found

```
****ERROR: The key label was not found.
```

When the command is executed successfully:

```
The Key Label Mapping was deleted successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --deleteKeyLabelMapping 1

Command Name:
createBEP

Description:
Creates a cartridges encryption policy

Parameters:
Volser Start,
Volser End,
Logical library name
Media type : Possible values are LTO, 3592
Key label mapping index 1 : You need to run “viewKeyLabelMapping” command first, to get the current list of valid indexes.
Key label mapping index 2 : This is an optional value.(Is user for JAG drives)
.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the parameters are incorrect or missing:

```
****ERROR: "***** ERROR: Starting range is greater than ending range

***** ERROR: [mediaType] not a valid media type [LTO, 3592]

***** ERROR: There are missing parameters:
*****        VOLSER start
"*****              VOLSER end
*****        LogLib name
*****        Media type (LTO,3592)
*****        KLM 1 index
*****        KLM 2 index (optional)
```

When the command is executed successfully:

```
The cartridge encryption policy was created successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –createBEP TTT200,TTT300,test1,3592,1

Command Name:
`modifyBEP`

Description:
Edits a cartridges encryption policy

Parameters:
Index: You need to run “viewBEP” command first, to get the current list of valid indexes.
Key label mapping index 1 : You need to run “viewKeyLabelMapping” command first, to get the current list of valid indexes. The parameter is --KLM1
Key label mapping index 2 : This is an optional value. The parameter is --KLM2
Bep enabled: Possible values are True , false. The parameter is --BEP

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the parameters are incorrect or missing:

```
****ERROR: There are missing parameters. At least one of these parameters must be
included:
 BEP:  [True/False]
KLM1: index taken from viewKeyLabelMapping
KLM2: index taken from viewKeyLabelMapping

****ERROR: Wrong value. BEP parameter valid values are:
BEP:  [True/False]
```

When the command is executed successfully:

```
The cartridge encryption policy was updated successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --editBEP 1 –KLM1 3 –BEP false –KLM2 1

Command Name:
`deleteBEP`

Description:
Deletes a cartridge encryption policy. You need to execute "viewBEP" command before, to get the index of the BEP that will be deleted

Parameters:
Index: You need to execute "viewBEP" to get this value.
.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the index cannot be found

```
****ERROR: The index was not found.
```

When the command is executed successfully:

```
The cartridge encryption policy was deleted successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --deleteBEP 1

Command Name:
viewBEP

Description:
Show a list of all the volser ranges showing the BEP settings.

Parameters:
None.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully:

```
 Index,Volser Start,  Volser End,   Logical Library,  Media Type,    KLM1,    KLM2,  BEP
     1,      TTT200,      TTT300,            test1,       3592,       1,       3,    0
     2,      CCC000,      DDD000,            test1,       3592,       1,       3,    1
     3,      TTT400,      TTT500,            test1,       3592,       1,       0,    1
     5,      TTT000,      TTT100,            test1,       3592,       1,       3,    1
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewBEP

Command Name:
`saveConfiguration`

Description:
Saves all databases to the user computer in a dbz file. The file will be save in the same directory where the CLI is running.

Parameters:
None.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
There was a problem dowloading configuration file. Try again
```

When the file has been downloaded:

```
The library DB backup process has finished successfully
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –saveConfiguration

---

Command Name:
`restoreConfiguration`

Description:
Restores all databases to the library.

Parameters:
File name: If the file is not in the same directory where CLI is running you must include the full path.

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 *****
```

When the command is executed successfully :

```
The library DB restore process has finished successfully.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] --restoreConfiguration TS4500_121214_LibName.dbz

---

Command Name:
`removeDataCartridges`

Description:
Remove the data cartridges (cartridges moved to the I/O station).
If VIO is enabled it will work no move the cartridge to the I/O station, it will only mark it as a cartridge ready for export. The cartridge will appear as unassigned.
If VIO is disabled, the cartridge will be move to the I/O station.

Parameters:
FileName.
Each line in the file will be interpreted as a remove operation.

The input file would take the following format:

From a storage location (even if tier = 0, you have to use the four values):

[F,C,R,T]

Or

VOLSER

--

The VOLSER list can be send in the command line directly.

Expected output:

In case the input file has a wrong format an error message will be shown:

```
***** ERROR: Wrong number of parameters. *****
[F,C,R,T]
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]  ***
**********************************************************************

Moving to next element...
```

When the command is executed successfully then the message will be:

```
"The cartridge [FCRT] has been removed successfully"
```

This command receives as input, a text file with one or more remove operations, once all of the operations are processed a message indicating the end of the whole process will be shown.
`"Done."`
Example:

fileName.txt will contain:

F1,C2,R8,T0

F1,C3,R18,T5

Or

A00045JA

A00047JA

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –removeDataCartridges fileName.txt

---

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –removeDataCartridges A00000JA,A00001JA,A00002JA

Command Name:
`setAutoEjectCleaningCarts`

Description:
Sets ENABLED/DISABLED the auto eject expired cleaning cartridges flag in the library.

Parameters:
flag value:
ENABLED for ON
DISABLED for OFF

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The auto eject cleaning cartridges flag was set successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setAutoEjectCleaningCarts ENABLED

Command Name:
`viewAccessor`

Description:
Shows accessor's status and usage statistics.

Parameters:
Nono

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully :

```
Availability

                    ,     Accessor A
           Accessor,    OK - Online
          Gripper 1,    OK - Online
          Gripper 2,    OK - Online


Usage Statistics

                          ,     Accessor A Component
                  Pivots,                       267
          Gripper 1 gets,                       104
          Gripper 1 puts,                       103
          Gripper 2 gets,                        46
          Gripper 2 puts,                        46
          Bar code scans,                      5724
       X travel (meters),                        86
       Y travel (meters),                       114
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –viewAccessor

Command Name:
batch

Description:
Perform actions specified within a file.

Parameters:
FileName.
Each line in the file will be interpreted as a new command.

The input file would take the following format:

--commandName [commandParameters]

Expected output:

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution – [Message]  ***
**********************************************************************
```

and the execution of the batch file will be stopped.

When the command is executed successfully the corresponding message for that command will be displayed.

Example:

fileName.txt will contain:

--viewLogicalLibraries

--viewIoStation

--viewAccessor

--viewRoles

--viewLogicalLibraryDetails GusVIO255

--bulkAssignDataCartridges bulk.txt,LTO_6

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] – batch fileName.txt

---

Command Name:
`setSSL`

Description:
Set the SSL flag to Enabled or Disabled

Parameters:
flag value:
ENABLED / DISABLED

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The SSL flag was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setSSL ENABLED

Command Name:
`setSSLForEKM`

Description:
Set the SSL flag to Enabled or Disabled for a given EKM server

Parameters:
flag value:
ENABLED / DISABLED
EKM: EKM server ip

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The SSL flag was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setSSLForEKM ENABLED –EKM [EKM server ip addreess]

Command Name:
`setScannerSpeed`

Description:
Sets scanner speed.

Parameters:
speed: 0,30,40,50,60,70,80,90,100

0 = Nominal speed

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The scanner speed was updated successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setScannerSpeed 100

Command Name:
downloadDrivesLog

Description:
Downloads the zip file with the selected drives's logs.

Parameters:
List of drives. In this format:
ALL : For all drives
F1C1R2, F3C4R4 (drives must be comma separated)

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
There was a problem dowloading logs. Try again
```

When the command is executed successfully a message indicating the completed percentage will be shown:

```
Downloading... [completed] / [totalSize] has been downloaded.
```

When the file has been downloaded:

```
The log file: "TS4500_DRIVELOGS_[yyyy-MM-dd_hh.mm.ss].zip has been downloaded.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –downloadDrivesLog ALL
>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –downloadDrivesLog F1C2R4,F1C3R4

Command Name:
`setSlotOnline`

Description:
Set Online a given slot

Parameters:
-f Frame index
-c Column index
-r Row index

when using r = 0 , all the rows for the given column will be set online

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
Storage Slot was set online
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –setSlotOnline -f 1 -c 3 -r 1

Command Name:
`setSlotOffline`

Description:
Set offline a given slot

Parameters:
-f Frame index
-c Column index
-r Row index

when using r = 0 , all the rows for the given column will be set offline

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
Storage Slot was set offline
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –setSlotOffline -f 1 -c 3 -r 1

Command Name:
`viewOfflineComponents`

Description:
View offline slots

Parameters:
None

Example of output:

Frame,	Column,	Row,	Status
1,	3,	1,	Offline
1,	3,	2,	Offline
1,	3,	3,	Offline

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –viewOfflineComponents

---

Parameter Name:
`ssl`

Description:
This parameter must be used to work with a library with SSL active

Parameters:

If you don't use -ssl with a library that has SSL you'll get the following error message:

***** ERROR: peer not authenticated *****
Response Body: Not available

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –[command] [parameters] -ssl

Command Name:
`getFWVersion`

Description:
This command returns the FW version installed in the library.

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
Firmware Version: 1.1.0.0-0AE.00
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –getFWVersion

---

Command Name:
`version`

Description:
This command returns the CLI version

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
CLI GEN 4 Version: 1.14
Build: 12/22/2014
```

How to run:

>> java -jar TS4500CLI.jar -version

Command Name:
`startLibraryVerify`

Description:
This command starts the library verify process for IO stations. To complete this process we need to run 3 different commands:
startLibraryVerify
continueLibraryVerify
continueCloseLibraryVerify

Once each command is done , a message with instructions will be shown to the user. When startLibraryVerify is done, CLI will ask the user to OPEN all IO doors before continue with the next command (continueLibraryVerify)

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
..Library verify - IO stations -  has started...
Library verify in
progress ..........................................................................
.....................................Done!
Next step:  OPEN all IO doors , and then use "continueLibraryVerify" command to continue
the test.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –startLibraryVerify

Command Name:
`continueLibraryVerify`

Description:
This command is the second step in the library verify process for IO stations. Previous command:
startLibraryVerify
Next command:
continueCloseLibraryVerify

Once each command is done , a message with instructions will be shown to the user. When continueLibraryVerify is done, CLI will ask the user to CLOSE all IO doors before continue with the next command (continueCloseLibraryVerify)

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
IO doors opened. Library verify continues now...
....Done!
Next step: CLOSE all IO doors, and then use "continueCloseLibraryVerify" command to
continue the test.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] --continueLibraryVerify

---

Command Name:
`continueCloseLibraryVerify`

Description:
This command is the last step in the library verify process for IO stations. Previous command: continueLibraryVerify

Once each command is done , a message with instructions will be shown to the user. When continueCloseLibraryVerify is done, will give you the test result.

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
IO doors closed. Library verify continues now...
The test has been completed successfully
```

When the finds a problem with IO doors :

```
IO doors closed. Library verify continues now...
****ERROR: The test has FAILED. Please, restart the test using "startLibraryVerify"
command.
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] --continueCloseLibraryVerify

---

IMPORTANT!!!!:

The following commands are NOT available to custumers. These are exlusive for manufacturing team and/or internal use. Must be careful when using any of these.

Command Name:
`testMoveSlot`

Description:
This command is used to test deep Cells.

Parameters:
[PUT/GET]
FCRTT = Frame Column Row Tier Type [2 for storage slots]
GRIPPER_1 / GRIPPER_2

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
The test was successful
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –testMoveSlot GET,F2C9R1T1T2,GRIPPER_1

Command Name:
`getFWVersion`

Description:
This command returns the FW version installed in the library.

Parameters:
-

Example of output:

In case that the command could not be executed, then a Error message will be shown:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully

```
Firmware Version: 1.1.0.0-0AE.00
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [user] -p [password] –getFWVersion

---

Command Name:
`enableSSH`

Description:
Enables SSH port.
NOTE: This command is only for internal use

Parameters:
None

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The message was sent successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –enableSSH

Command Name:
setVIOStatus

Description:
Sets the value of the VIO flag on the library.
Parameters:
ENABLED or DISABLED
Example of output:

In case the parameter is different from the accepted values a message will be shown:

```
*** ERROR: Parameter not valid. Value should be \"Enabled\" or \"Disabled\"
```

In case that the command could not be executed, then a Error message will be shown:

```
*** Error: There was a problem in the command execution - [Message]
**********************************************************************
```

When the command is executed successfully then the message will be:

```
"The VIO Status was set successfully"
```

How to run:
>> java -jar TS4500_CLI.jar -ip [LCC ip] -u superuser -p superuser –setVIOStatus ENABLED

Command Name:
`sendPMR`

Description:
This command is use by TSSC to send a PMR

Parameters:
id: Error id
pmr: PMR id

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The message was sent successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [ip] -u [user] -p [password] --sendPMR -id [id] -pmr [pmr]

Command Name:
`installLicensedFunction`

Description:
Install/Uninstall a licensed function with the given license key code
NOTE: This command is only for internal use

Parameters:
Flag: install / uninstall
License key code: A string of 12 characters. Example: 191001001610

Example of output:

In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The licensed feature was [installed/uninstalled] successfully"`

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –installLicensedFunction install, 191001001610

Command Name:
`sendMessage`

Description:
This command will send a debug message to the library (as in the Service web page).

Parameters:
All parameters are optional, when one is missing the default value will be send (00).
--targetNode
--targetTask
--sourceTask
--msgType
--control
--msgKey
--index
--flowControl
--dataSize
--data

For targetNode,targetTask,sourceTask,msgType the values can be hexadecimal (without 0x) or the name of the tasks/node as in the web page.
To get the messages updated, you can have the .h files with the defines in the same directory where CLI is running and these will be readed from there.

Example of output:

In case there is a problem with the command execution:
`***** ERROR: There was a problem in the comand execution. Return code = 1 ***`

When the command is executed successfully then the message will be:

`"The message was sent successfully"`

How to run:

>>java -jar TS4500CLI.jar -ip 10.1.1.1 -u [userName] -p [password] --sendMessage --targetNode 41 --targetTask 97 --sourceNode 41 --sourceTask 47 --msgType 850006 --dataSize 10 --data "01 03 01 01 02 01 03 02 01 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00"

>>java -jar TS4500CLI.jar -ip 10.1.1.1 -u [userName] -p [password] --sendMessage --targetNode NODE_LCA_A_1 --targetTask TID_WQM --sourceNode 41 --sourceTask 47 --msgType MT_MSP_PHYSICAL_MOVE_COMMAND --dataSize 10 --data "01 03 01 01 02 01 03 02 01 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00"

Command Name:
`startDiscoverHW`

Description:
Starts a discover HW operation

NOTE: This command is only for internal use

Parameters:
OVERWRITE: This flag is used to clear all configuration data and rebuild from scratch and discover data for selected frames.
Frames list [1-18]: A list of the selected frames that will be use for discover.

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"Discover HW has started"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –startDiscoverHW OVERWRITE,1,2,5,18
>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –startDiscoverHW 1,2,5,15,18

Command Name:
`startInventory`

Description:
Starts an inventory operation

NOTE: This command is only for internal use

Parameters:
--library : Possible values are SINGLE (single selection) NO (NO selection)
--audit: Possible values are FIRST (Scan tier 0 and tier 1) , ALL (Scan all tiers)
--frame: Possible values are ALL (entiry library) , # (frame index)
Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"Inventory has started"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –startInventory --library SINGLE –audit ALL –frame ALL

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –startInventory --library NO –audit FIRST –frame 5

Command Name:
`startCalibration`

Description:
Starts a calibration operation

NOTE: This command is only for internal use

Parameters:
It will receive the component that will be calibrate. Possible values are:
LIBRARY
FRAME,[frame index]
DRIVE,[drive location in this format: F1C2R3]
IO,[frameIndex],[IOIndex]
ACCESSOR,[1=Accessor A, 2=Accessor B]

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"Calibration has started"
```

How to run:

```
>java -jar TS4500CLI.jar -u admin -p admin -ip 9.11.59.80 --startCalibration LIBRARY
>java -jar TS4500CLI.jar -u admin -p admin -ip 9.11.59.80 --startCalibration DRIVE F1C2R3
>java -jar TS4500CLI.jar -u admin -p admin -ip 9.11.59.80 --startCalibration FRAME,1
>java -jar TS4500CLI.jar -u admin -p admin -ip 9.11.59.80 --startCalibration IO,1,2
>java -jar TS4500CLI.jar -u admin -p admin -ip 9.11.59.80 --startCalibration IO,1,2
```

Command Name:
`createLL`

Description:
Creates a new logical library

Parameters:
Name: Name of the new logical library
Media type: LTO or JAG

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The logical library was created successfully""
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –createLL NewLib,JAG

Command Name:
`deleteLL`

Description:
Deletes a given logical library

Parameters:
Name: Name of the logical library

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The logical library was deleted successfully""
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –deleteLL NewLib

---

Command Name:
`assignDriveToLL`

Description:
Assigns a drive to a given logical library

Parameters:
Name: Name of the logical library
Drive: F#C#R#

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The drive was assigned successfully""
```

How to run:

> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –assignDriveToLL CLILib,F1C2R1

Command Name:
`unassignDrive`

Description:
Unassigns a given drive

Parameters:
Drive: F#C#R#

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The drive was unassigned successfully""
```

How to run:

> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –unassignDrive F1C2R1

Command Name:
`setSleep`

Description:
`Sleeps thread for given seconds`

Parameters:
Seconds

Example of output:
`--`

How to run:

> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setSleep 5

Command Name:
`setSerialNumber`

Description:
Sets the serial number for the L frame

Parameters:
L frame index,
Number frames,
Serial number

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The serial number has been set correctly""
```

How to run:

> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setSerialNumber 1,2,FR0034

Command Name:
`setBaseWWNN`

Description:
Sets scanner speed.

Parameters:
Base WWNN in hexadecimal format (8 characters). The accepted values are in these ranges:
60440000 to 6047FFFF
630F0000 to 630F3FFF

Example of output:

In case there is a problem with the command execution:

```
***** ERROR: There was a problem in the comand execution. Return code = 1 ***
```

When the command is executed successfully then the message will be:

```
"The Base WWNN was set successfully"
```

How to run:

>> java -jar TS4500CLI.jar -ip [LCC ip] -u [userName] -p [password] –setBaseWWNN 60440000
