TCG Storage
Application Note: Encrypting Drives Compliant with Opal SSC

Specification Version 1.00 Final
Revision 1.00

19 February, 2010

Contact: admin@trustedcomputinggroup.org

Disclaimers, Notices, and License Terms

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

Without limitation, TCG disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and TCG disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

Contact the Trusted Computing Group at www.trustedcomputinggroup.org for information on specification licensing through membership agreements.

Any marks and brands contained herein are the property of their respective owners.

Revision History

Version 1.00	Date	Description
Rev 1.00	19 February, 2010	First publication

Table of Contents

1 Introduction 8

1.1 Purpose and Scope 8

1.2 Intended Audience 8

1.3 References to Other Documents 8

2 Use Scenarios: Desktop/Notebook PC Disk Encryption using the Locking SP 9

3 Recommended Implementation 10

3.1 Brief Description of the Sessions and Commands 10

3.1.1 Discovery 10
3.1.1.1 Discovering whether a storage device supports Opal SSC 10
3.1.1.1.1 Level 0 Discovery Request: 10
3.1.1.2 Exchange communication properties with the TPer: 10
3.1.2 Taking ownership of the storage device 10
3.1.3 Activating the Locking SP 11
3.1.4 Changing the Admin1 PIN in the Locking SP and adding users 11
3.1.5 Configuring Locking Objects (LBA ranges) 12
3.1.6 Unlocking ranges 13
3.1.7 Erasing a range 13
3.1.8 Enabling MBR Shadowing 13
3.1.9 Un-shadowing the MBR 14
3.1.10 Reverting the TPer 14
3.1.11 Reverting the Locking SP 14
3.1.12 Using the DataStore table 15

3.2 Command Tokenization 16

3.2.1 Discovery 16
3.2.1.1 Level 0 Discovery 16
3.2.1.1.1 Response 16
3.2.1.2 Exchange communication properties with the TPer 17
3.2.1.2.1 Host to TPer Properties invocation 18
3.2.1.2.2 TPer to Host Properties response 21
3.2.2 Common Commands and Responses 25
3.2.2.1 SyncSession Response 25
3.2.2.2 Set Results 26
3.2.2.3 Ending the Session 27
3.2.2.3.1 Send End of Session Token 27
3.2.2.3.2 End of Session Response 28
3.2.3 Taking ownership of the storage device 29
3.2.3.1 Open a session to the Admin SP as the Anybody authority 29
3.2.3.1.1 StartSession – Admin SP 29
3.2.3.1.2 SyncSession – Admin SP 30
3.2.3.2 Get the MSID's PIN value from the C_PIN table 30
3.2.3.3 Close the session 32
3.2.3.4 Open a session to the Admin SP as the SID authority using the <MSID_password> 32
3.2.3.5 Set the <new_SID_password> value in the SID's C_PIN credential PIN column 34
3.2.3.6 Close the session 35
3.2.4 Activating the Locking SP 35
3.2.4.1 Open a session to the Admin SP as the SID authority 35
3.2.4.2 Determine the life cycle state of the Locking SP 36

3.2.4.3 Activate the Locking SP by using the Activate method on the Locking SP object in the Admin SP 39
3.2.4.4 Close the session 40
3.2.5 Changing the Admin1 PIN in the Locking SP and adding users 40
3.2.5.1 Open a session to the Locking SP as Admin1 40
3.2.5.2 Set the <Admin1_password> value in Admin1's C_PIN credential PIN column 42
3.2.5.3 Enable the User1 authority 43
3.2.5.4 Change the password for User1 44
3.2.5.5 Enable the User2 authority 46
3.2.5.6 Change the password for User2 47
3.2.5.7 Close the session 48
3.2.6 Configuring Locking Objects (LBA ranges) 48
3.2.6.1 Open a session to the Locking SP as Admin1 48
3.2.6.2 Configure the range and enable read and write locking by changing RangeStart, RangeLength, ReadLockEnabled and WriteLockEnabled for Locking_Range1 50
3.2.6.3 Retrieve the UID of the range's media encryption key 51
3.2.6.4 Perform a Secure Erase of the range 53
3.2.6.5 Give access to multiple users to read-unlock the range (User1 and User2) 55
3.2.6.6 Give access to multiple users to write-unlock the range (User1 and User2) 57
3.2.6.7 Lock for read and write, by setting ReadLocked and WriteLocked for this range to TRUE 59
3.2.6.8 Close the session 60
3.2.7 Unlocking ranges 60
3.2.7.1 Open a session to the Locking SP as User1 60
3.2.7.2 Unlock a range by setting the Locked columns in the Locking table to FALSE 61
3.2.7.3 Close the session 63
3.2.8 Erasing a range 63
3.2.8.1 Open a session to the Locking SP as Admin1 63
3.2.8.2 Invoke the GenKey method on the media encryption key associated with one of the ranges 64
3.2.8.3 Close the session 66
3.2.9 Enabling MBR Shadowing 66
3.2.9.1 Open a session to the Locking SP as Admin1 66
3.2.9.2 Give access to User1 and User2 for setting the MBR Shadowing "Done" flag 67
3.2.9.3 Set the MBR table 69
3.2.9.4 Enable the MBR Shadowing feature 70
3.2.9.5 Close the session 72
3.2.10 Un-shadowing the MBR 72
3.2.10.1 Open a session to the Locking SP as User1 72
3.2.10.2 Un-shadow the MBR 73
3.2.10.3 Close the session 74
3.2.11 Reverting the TPer 74
3.2.11.1 Open a session to the Admin SP as SID 74
3.2.11.2 Revert the TPer 76
3.2.12 Reverting the Locking SP 78
3.2.12.1 Open a session to the Locking SP as Admin1 78
3.2.12.2 Revert the Locking SP 79
3.2.13 Using the DataStore table 81
3.2.13.1 Open a session to the Locking SP as Admin1 81
3.2.13.2 Give User1 write access to the DataStore table 82
3.2.13.3 Give User1 and User2 read access to the DataStore table 84
3.2.13.4 Close the session 85
3.2.13.5 Open a session to the Locking SP as User1 85
3.2.13.6 Write data to the DataStore table 87
3.2.13.7 Close the session 88
3.2.13.8 Open a session to the Locking SP as User2 88
3.2.13.9 Read data from the DataStore table 90
3.2.13.10 Close the session 92

Tables

Table 1 Level 0 Discovery Response 16
Table 2 Properties 19
Table 3 Properties Response 21
Table 4 SyncSession 25
Table 5 Set Method Results 27
Table 6 EndSession 27
Table 7 EndSession – Response 28
Table 8 StartSession – Admin SP 29
Table 9 Get – C_PIN_MSID 30
Table 10 Get – C_PIN_MSID – Response 31
Table 11 StartSession – Admin SP 32
Table 12 Set – C_PIN_SID 34
Table 13 StartSession – Admin SP 35
Table 14 Get – LockingSP 37
Table 15 Get – LockingSP – Response 38
Table 16 Activate – LockingSP 39
Table 17 Activate – LockingSP – Response 40
Table 18 StartSession – Locking SP 41
Table 19 Set – C_PIN_Admin1 42
Table 20 Set – User1 43
Table 21 Set – C_PIN_User1 45
Table 22 Set – User2 46
Table 23 Set – C_PIN_User2 47
Table 24 StartSession – Locking SP 48
Table 25 Set – Locking_Range1 50
Table 26 Get – Locking_Range1 51
Table 27 Get – Locking_Range1 – Response 53
Table 28 Genkey – K_AES_256_Range1_Key 54
Table 29 Genkey – K_AES_256_Range1_Key – Response 55
Table 30 Set – ACE_Locking_Range1_Set_RdLocked 55
Table 31 Set – ACE_Locking_Range1_Set_WrLocked 57
Table 32 Set – Locking_Range1 59
Table 33 StartSession – Locking SP 60
Table 34 Set – Locking_Range1 62
Table 35 StartSession – Admin SP 63
Table 36 Genkey – K_AES_256_Range1_Key 64
Table 37 Genkey – K_AES_256_Range1_Key – Response 65
Table 38 StartSession – Locking SP 66
Table 39 Set – ACE_MBRControl_Set_Done 68
Table 40 Set – MBR 69
Table 41 Set – MBRControl 71
Table 42 StartSession – Locking SP 72
Table 43 Set – MBRControl 73
Table 44 StartSession – Admin SP 75
Table 45 Revert – AdminSP 76
Table 46 Revert – AdminSP – Response 77
Table 47 StartSession – Locking SP 78
Table 48 RevertSP – ThisSP 79
Table 49 RevertSP – ThisSP – Response 80
Table 50 StartSession – Locking SP 81
Table 51 Set – ACE_DataStore_Set_All 82
Table 52 Set – ACE_DataStore_Get_All 84
Table 53 StartSession – Locking SP 86
Table 54 Set – DataStore 87

Table 55 StartSession – Locking SP 88
Table 56 Get – DataStore 90
Table 57 Get – DataStore – Response 91

1 Introduction

This section summarizes the purpose, scope, and intended audience for this document. The contents of this document are informative.

1.1 Purpose and Scope

The purpose of this document is to provide examples of the communication between a host and a storage device implementing the TCG Storage Security Subsystem Class: Opal (Opal SSC) [2] and the TCG Storage Architecture Core Specification (Core Spec) [1] to perform the scenarios listed in section 2.

1.2 Intended Audience

The intended audience for this document is implementers of systems using TCG Opal SSC-compliant storage devices.

1.3 References to Other Documents

[1] Trusted Computing Group (TCG), "TCG Storage Architecture Core Specification", Version 2.00
[2] Trusted Computing Group (TCG), "TCG Storage Security Subsystem Class: Opal", Version 1.00
[3] Trusted Computing Group (TCG), "TCG Storage Storage Interface Interactions Specification", Version 1.00

2 Use Scenarios: Desktop/Notebook PC Disk Encryption using the Locking SP

This document provides example communications with a device that complies with [2]. Examples are provided for the following scenarios:.

- Discovering whether a storage device supports Opal SSC
- Taking ownership of the storage device
- Activating the Locking SP
- Changing the Admin1 PIN in the Locking SP and adding users
- Configuring Locking Objects (LBA ranges)
- Unlocking ranges
- Erasing a range
- Enabling the MBR shadow
- Un-shadowing the MBR
- Reverting the TPer
- Reverting the Locking SP
- Using the DataStore table

3 Recommended Implementation

This section describes an example of the communications utilized in implementation of the use scenarios, using commands described by the TCG Storage Architecture Core Specification [1] and the Opal SSC [2].

3.1 Brief Description of the Sessions and Commands

3.1.1 Discovery

3.1.1.1 Discovering whether a storage device supports Opal SSC

This includes the sequence of operations that a host application should go through to ascertain whether a storage device supports the TCG Storage Opal SSC specification [2].

3.1.1.1.1 Level 0 Discovery Request:

```
IF_RECV with Protocol 01: Level 0 discovery (ComID 0x0001)
```

An Opal SSC compliant SD will return the following Level 0 response (additional descriptors may be present if the SD supports multiple SSCs):

- Level 0 Discovery Header
- TPer Feature Descriptor
- Locking Feature Descriptor
- Opal SSC Feature Descriptor

A device compliant with the Opal SSC will have LockingSupported = 1 and MediaEncryption = 1

3.1.1.2 Exchange communication properties with the TPer:

This includes the sequence of operations that a host application should go through to discover the storage device’s communication properties and inform the storage device of the host’s communication properties.

The Properties method may be used by the host to provide its communication properties to the TPer, and to retrieve the communication properties of the TPer. Invocation of the Properties method is optional. Communications may occur using just the minimum communications capability.

3.1.2 Taking ownership of the storage device

This introduces the steps the host follows to take ownership of the storage device (see 3.2.3). The host:

1. Open a session to the Admin SP as the Anybody authority
    a. StartSession
    b. SyncSession
2. Gets the MSID’s PIN value from the C_PIN table
    a. Get
    b. Get Result
3. Closes the session

    a. End of Session
    b. End of Session Response

4. Opens a session to the Admin SP as the SID authority using the <MSID_password>
    a. StartSession
    b. SyncSession
5. Sets the <new_SID_password> value in the SID's C_PIN credential PIN column
    a. Set
    b. Set Result
6. Closes the session
    a. End of Session
    b. End of Session Response

3.1.3 Activating the Locking SP

This section introduces the steps the host follows to activate the Locking SP (see 3.2.4). The host:

1. Opens a session to the Admin SP as the SID authority
    a. StartSession
    b. SyncSession
2. Determines the life cycle state of the Locking SP
    a. Get
    b. Get Result
3. Activates the Locking SP by using the Activate method on the Locking SP object in the Admin SP
    a. Activate
    b. Activate Result
4. Closes the session
    a. End of Session
    b. End of Session Response

3.1.4 Changing the Admin1 PIN in the Locking SP and adding users

This section introduces the steps the host follows to change the Admin1 PIN in the Locking SP, and one way the host could add users to the Locking SP (see 3.2.5). The host:

1. Opens a session to the Locking SP as Admin1
    a. StartSession
    b. SyncSession
2. Sets the <Admin1_password> value in Admin1's C_PIN credential PIN column
    a. Set
    b. Set Result
3. Enables the User1 authority
    a. Set

   b. Set Result

4. Changes the password for User1
   a. Set
   b. Set Result
5. Enables the User2 authority
   a. Set
   b. Set Result
6. Changes the password for User2
   a. Set
   b. Set Result
7. Closes the session
   a. End of Session
   b. End of Session Response

3.1.5 Configuring Locking Objects (LBA ranges)

This section introduces one way the host could configure the Locking table LBA ranges (see 3.2.6). The host:

1. Opens a session to the Locking SP as Admin1
   a. StartSession
   b. SyncSession
2. Configures the range and enables read and write locking by changing RangeStart, RangeLength, ReadLockEnabled and WriteLockEnabled for Locking_Range1
   a. Set
   b. Set Result
3. Retrieves the UID of the range’s media encryption key
   a. Get
   b. Get Result
4. Performs a Secure Erase of the range
   a. GenKey
   b. GenKey Result
5. Gives access to multiple users to read-unlock the range (User1 and User2)
   a. Set
   b. Set Result
6. Gives access to multiple users to write-unlock the range (User1 and User2)
   a. Set
   b. Set Result
7. Locks for read and write, by setting ReadLocked and WriteLocked for this range to TRUE
   a. Set

   b. Set Result

8. Closes the session
   a. End of Session
   b. End of Session Response

3.1.6 Unlocking ranges

This section introduces the steps the host follows to unlock LBA ranges (see 3.2.7). The host:

1. Opens a session to the Locking SP as User1
   a. StartSession
   b. SyncSession
2. Unlocks a range by setting the Locked columns in the Locking table to FALSE
   a. Set
   b. Set Result
3. Closes the session
   a. End of Session
   b. End of Session Response

3.1.7 Erasing a range

This section introduces the steps the host follows to erase LBA ranges (see 3.2.8). The host:

1. Opens a session to the Locking SP as Admin1
   a. StartSession
   b. SyncSession
2. Invokes the GenKey method on the media encryption key associated with one of the ranges
   a. GenKey
   b. GenKey Result
3. Closes the session
   a. End of Session
   b. End of Session Result

3.1.8 Enabling MBR Shadowing

This section introduces the steps the host follows for one possible configuration of enabling the MBR Shadowing feature (see 3.2.9). The host:

1. Opens a session to the Locking SP as Admin1
   a. StartSession
   b. SyncSession
2. Gives access to User1 and User2 for setting the MBR Shadowing “Done” flag:
   a. Set
   b. Set Result
3. Sets the MBR table
   a. Set

   b. Set Result

4. Enables the MBR Shadowing feature
   a. Set
   b. Set Result
5. Closes the session
   a. End of Session
   b. End of Session Response

3.1.9 Un-shadowing the MBR

This section introduces the steps the host follows to transition the MBR to the un-shadowed state (see 3.2.10). The host:

1. Opens a session to the Locking SP as User1
   a. StartSession
   b. SyncSession
2. Un-shadows the MBR
   a. Set
   b. Set Result
3. Closes the session
   a. End of Session
   b. End of Session Response

3.1.10 Reverting the TPer

This section introduces the steps the host follows to revert the TPer to its Original Manufacturing State (see 3.2.11). The host:

1. Opens a session to the Admin SP as SID
   a. StartSession
   b. SyncSession
2. Reverts the TPer
   a. Revert
   b. Revert Result

Note: The TPer aborts the session immediately after reporting the Revert result.

3.1.11 Reverting the Locking SP

This section introduces the steps the host follows to revert the Locking SP to its Original Manufacturing State (see 3.2.12). The host:

1. Opens a session to the Locking SP as Admin1
   a. StartSession
   b. SyncSession
2. Reverts the Locking SP
   a. RevertSP

  b. RevertSP Result

Note: The TPer aborts the session immediately after reporting the RevertSP result.

3.1.12 Using the DataStore table

This section introduces the steps the host follows for one possible configuration of using the DataStore table functionality (see 3.2.13). The host:

1. Opens a session to the Locking SP as Admin1
    a. StartSession
    b. SyncSession
2. Gives User1 write access to the DataStore table
    a. Set
    b. Set Result
3. Gives User1 and User2 read access to the DataStore table
    a. Set
    b. Set Result
4. Closes the session
    a. End of Session
    b. End of Session Response
5. Opens a session to the Locking SP as User1
    a. StartSession
    b. SyncSession
6. Writes data to the DataStore table
    a. Set
    b. Set Result
7. Closes the session
    a. End of Session
    b. End of Session Result
8. Opens a session to the Locking SP as User2
    a. StartSession
    b. SyncSession
9. Reads data from the DataStore table
    a. Get
    b. Get Result
10. Closes the session
    a. End of Session
    b. End of Session Result

3.2 Command Tokenization

This section provides the additional details regarding the commands described in section 3.1, as well as the tokenization of each command and the packaging of those commands in Subpackets, Packets and ComPackets.

The following details are common to all relevant commands as defined in this document, but may vary between implementations. In this document:

1. All commands use a reserved Extended ComID value of `0x07FE0000`
2. The host always uses the Host Session Number (HSN) `0x00000001`.
3. The TPer always uses the TPer Session Number (TSN) `0x00001001`.
4. Communications sent from the host to the TPer have a Packet.SeqNumber of 0's.
5. Communications sent from the TPer to the Host have a Packet.SeqNumber of 0's.

All transfers between the host and storage device are in 512 byte blocks. If the ComPacket does not end at a 512 byte boundary, bytes of 0x00 are appended after the ComPacket as pad up to the end of the block.

3.2.1 Discovery

3.2.1.1 Level 0 Discovery

The values in the Level 0 Discovery Response reported in this section are examples and vary between implementations and LBA Range locking states.

3.2.1.1.1 Response

```
0000 00000060 00000001 00000000 00000000
0010 00000000 00000000 00000000 00000000
0020 00000000 00000000 00000000 00000000
0030 0001100C 11000000 00000000 00000000
0040 0002100C 09000000 00000000 00000000
0050 02001010 07FE0001 00000000 00000000
0060 00000000 00000000 00000000 00000000
0070 00000000 00000000 00000000 00000000
                           . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 1 Level 0 Discovery Response

Bytes	**Purpose**	**Value**	**Notes**
Header			
00 00 00 60	Length of Parameter Data	96	This field is vendor unique - other values may be present
00 00 00 01	Data Structure Revision	1	
00 00 00 00 00 00 00 00	Reserved	0's	
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Vendor Specific	0's	This field is vendor unique - other values may be present
TPer Feature Descriptor			
00 01	Feature Code	1	

1	Version	1	
0	Reserved	0	
0C	Length	12	
11	Tper Features Supported	Streaming Supported and Sync Supported	Other features may be supported. Streaming and Sync must be supported
00 00 00 00 00 00 00 00 00 00 00	Reserved		
Locking Feature Descriptor			
00 02	Feature Code	2	
1	Version	1	
0	Reserved	0	
0C	Length	12	
09	Locking Features Supported	Media Encryption and Locking Supported	Other values are possible based on Locking SP configuration
00 00 00 00 00 00 00 00 00 00 00	Reserved	0’s	
Opal SSC Feature Descriptor			
02 00	Feature Code	2, 0	
1	Version	1	
0	Reserved	0	
10	Length	16	
07 FE	Base ComID	07FE	This field is vendor unique – other values may be present
00 01	Number of ComIDs	1	
00	Reserved (7 bits) | Range Crossing	0	Range Crossing value is vendor unique
00 00 00 00 00 00 00 00 00 00 00	Reserved	0’s	

3.2.1.2 Exchange communication properties with the TPer

In the following example, the host informs the TPer:

- Its receive buffer is 4KB in size (MaxComPacketSize)
- Its transmit buffer is 4KB in size (MaxResponseComPacketSize)
- It can receive Packets up to 4076 bytes in size (MaxPacketSize)
- It can accept individual tokens up to 4040 bytes in size (MaxIndTokenSize)
- It can accept a maximum of 1 Packet per ComPacket (MaxPackets)
- It can accept a maximum of 1 Subpacket per Packet (MaxSubpackets)
- It can accept a maximum of 1 method invocation per Subpacket (MaxMethods)

In the Properties response, the TPer informs the host:

- Its receive buffer is 8KB in size (MaxComPacketSize)
- Its transmit buffer is 8KB in size (MaxResponseComPacketSize)
- It can receive Packets up to 8172 bytes in size (MaxPacketSize)
- It can accept individual tokens up to 8136 bytes in size (MaxIndTokenSize)
- It can accept a maximum of 1 Packet per ComPacket (MaxPackets)
- It can accept a maximum of 1 Subpacket per Packet (MaxSubpackets)
- It can accept a maximum of 1 method invocation per Subpacket (MaxMethods)
- It does not accept continued tokens (ContinuedTokens)
- It does not support Packet sequence numbers (SequenceNumbers)
- It does not support the Packet ACK/NAK protocol (AckNak)
- It does not support the Asynchronous Communication Protocol (Asynchronous)
- It supports a maximum of 1 simultaneous session (MaxSessions)
- It supports a maximum of 2 authentications within a session (MaxAuthentications)
- It supports a maximum of 1 level of transactions within a session (MaxTransactionLimit)
- It supports a default session timeout of 2 minutes
- It will send ComPackets no larger than 4KB (HostProperties MaxComPacketSize)
- It will send Packets no larger than 4076 bytes (HostProperties MaxPacketSize)
- It will send individual tokens no larger than 4040 bytes (HostProperties MaxIndTokenSize)
- It will send no more than 1 Packet per ComPacket (HostProperties MaxPackets)
- It will send no more than 1 Subpacket per Packet (HostProperties MaxSubpackets)
- It will send no more than 1 method invocation per Subpacket (HostProperties MaxMethods)

Note that in this example, the TPer does not make use of the MaxResponseComPacketSize property for HostProperties, and therefore that property is not echoed back in the HostProperties parameter of the Properties response.

3.2.1.2.1 Host to TPer Properties invocation

```
session[0:0] -> SMUID.Properties[ HostProperties = [“MaxComPacketSize” = 4096,
“MaxResponseComPacketSize” = 4096, “MaxPacketSize” = 4076, “MaxIndTokenSize” = 4040,
“MaxPackets” = 1, “MaxSubpackets” = 1, “MaxMethods” = 1] ]
0000 00000000 07FE0000 00000000 00000000
0010 000000D0 00000000 00000000 00000000
0020 00000000 00000000 000000B8 00000000
0030 00000000 000000AB F8A80000 00000000
0040 00FFA800 00000000 00FF01F0 F200F0F2
0050 D0104D61 78436F6D 5061636B 65745369
0060 7A658210 00F3F2D0 184D6178 52657370
0070 6F6E7365 436F6D50 61636B65 7453697A
0080 65821000 F3F2AD4D 61785061 636B6574
0090 53697A65 820FECF3 F2AF4D61 78496E64
00A0 546F6B65 6E53697A 65820FC8 F3F2AA4D
00B0 61785061 636B6574 7301F3F2 AD4D6178
00C0 53756270 61636B65 747301F3 F2AA4D61
00D0 784D6574 686F6473 01F3F1F3 F1F9F000
00E0 0000F100 00000000 00000000 00000000
```

```
                       . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 2 Properties

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 D0	Length	208	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 B8	Length	184	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 AB	Length	171	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 01	Method UID	Properties Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostProperties”	
F0	Start List Token		Starts “HostProperties” list
F2	Start Name Token		name-value
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
4D 61 78 43 6F 6D 50 61 63 6B 65 74 53 69 7A 65		MaxComPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
10 00		4096	
F3	End Name Token		
F2	Start Name Token		name-value
D0 18	Medium Atom Token Header	Byte sequence; Length = 24	

4D 61 78 52 65 73 70 6F 6E 73 65 43 6F 6D 50 61 63 6B 65 74 53 69 7A 65		MaxResponseComPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
10 00		4096	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 50 61 63 6B 65 74 53 69 7A 65		MaxPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
0F EC		4076	
F3	End Name Token		
F2	Start Name Token		name-value
AF	Short Atom Token Header	Byte sequence; Length = 15	
4D 61 78 49 6E 64 54 6F 6B 65 6E 53 69 7A 65		MaxIndTokenSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
0F C8		4040	
F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 50 61 63 6B 65 74 73		MaxPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 53 75 62 70 61 63 6B 65 74 73		MaxSubPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 4D 65 74 68 6F 64 73		MaxMethods	
01	Tiny Atom Token	1	
F3	End Name Token		
F1	End List Token		Ends “HostProperties” list
F3	End Name Token		Ends “HostProperties” parameter
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

00	Pad		Included in Packet and ComPacket lengths

3.2.1.2.2 TPer to Host Properties response

```
session[0:0] <- SMUID.Properties[ Properties : [“MaxComPacketSize” = 8192,
“MaxResponseComPacketSize” = 8192, “MaxPacketSize” = 8172, “MaxIndTokenSize” = 8136,
“MaxPackets” = 1, “MaxSubpackets” = 1, “MaxMethods” = 1, “ContinuedTokens” = FALSE,
“SequenceNumbers” = FALSE, “AckNak” = FALSE, “Asynchronous” = FALSE, “MaxSessions” =
1, “MaxAuthentications” = 2, “MaxTransactionLimit” = 1, “DefSessionTimeout” =
120000], HostProperties = [“MaxComPacketSize” = 4096, “MaxPacketSize” = 4076,
“MaxIndTokenSize” = 4040, “MaxPackets” = 1, “MaxSubpackets” = 1, “MaxMethods” = 1] ]
0000 00000000 07FE0000 00000000 00000000
0010 000001D4 00000000 00000000 00000000
0020 00000000 00000000 000001BC 00000000
0030 00000000 000001B0 F8A80000 00000000
0040 00FFA800 00000000 00FF01F0 F0F2D010
0050 4D617843 6F6D5061 636B6574 53697A65
0060 822000F3 F2D0184D 61785265 73706F6E
0070 7365436F 6D506163 6B657453 697A6582
0080 2000F3F2 AD4D6178 5061636B 65745369
0090 7A65821F ECF3F2AF 4D617849 6E64546F
00A0 6B656E53 697A6582 1FC8F3F2 AA4D6178
00B0 5061636B 65747301 F3F2AD4D 61785375
00C0 62706163 6B657473 01F3F2AA 4D61784D
00D0 6574686F 647301F3 F2AF436F 6E74696E
00E0 75656454 6F6B656E 7300F3F2 AF536571
00F0 75656E63 654E756D 62657273 00F3F2A6
0100 41636B4E 616B00F3 F2AC4173 796E6368
0110 726F6E6F 757300F3 F2AB4D61 78536573
0120 73696F6E 7301F3F2 D0124D61 78417574
0130 68656E74 69636174 696F6E73 02F3F2D0
0140 134D6178 5472616E 73616374 696F6E4C
0150 696D6974 01F3F2D0 11446566 53657373
0160 696F6E54 696D656F 75748301 D4C0F3F1
0170 F200F0F2 D0104D61 78436F6D 5061636B
0180 65745369 7A658210 00F3F2AD 4D617850
0190 61636B65 7453697A 65820FEC F3F2AF4D
01A0 6178496E 64546F6B 656E5369 7A65820F
01B0 C8F3F2AA 4D617850 61636B65 747301F3
01C0 F2AD4D61 78537562 7061636B 65747301
01D0 F3F2AA4D 61784D65 74686F64 7301F3F1
01E0 F3F1F9F0 000000F1 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 3 Properties Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 01 D4	Length	468	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8

00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 01 BC	Length	444	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 01 B0	Length	432	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 01	Method UID	Properties Method UID	
F0	Start List Token		Begins parameter list
F0	Start List Token		Starts “Properties” list
F2	Start Name Token		name-value
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
4D 61 78 43 6F 6D 50 61 63 6B 65 74 53 69 7A 65		MaxComPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
20 00		8192	
F3	End Name Token		
F2	Start Name Token		name-value
D0 18	Medium Atom Token Header	Byte sequence; Length = 24	
4D 61 78 52 65 73 70 6F 6E 73 65 43 6F 6D 50 61 63 6B 65 74 53 69 7A 65		MaxResponseComPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
20 00		8192	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 50 61 63 6B 65 74 53 69 7A 65		MaxPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
1F EC		8172	
F3	End Name Token		
F2	Start Name Token		name-value
AF	Short Atom Token Header	Byte sequence; Length = 15	
4D 61 78 49 6E 64 54 6F 6B 65 6E 53 69 7A 65		MaxIndTokenSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
1F C8		8136	

F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 50 61 63 6B 65 74 73		MaxPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 53 75 62 70 61 63 6B 65 74 73		MaxSubPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 4D 65 74 68 6F 64 73		MaxMethods	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AF	Short Atom Token Header	Byte sequence; Length = 15	
43 6F 6E 74 69 6E 75 65 64 54 6F 6B 65 6E 73		ContinuedTokens	
00	Tiny Atom Token	0 = FALSE	
F3	End Name Token		
F2	Start Name Token		name-value
AF	Short Atom Token Header	Byte sequence; Length = 15	
53 65 71 75 65 6E 63 65 4E 75 6D 62 65 72 73		SequenceNumbers	
00	Tiny Atom Token	0 = FALSE	
F3	End Name Token		
F2	Start Name Token		name-value
A6	Short Atom Token Header	Byte sequence; Length = 6	
41 63 6B 4E 61 6B		AckNak	
00	Tiny Atom Token	0 = FALSE	
F3	End Name Token		
F2	Start Name Token		name-value
AC	Short Atom Token Header	Byte sequence; Length = 12	
41 73 79 6E 63 68 72 6F 6E 6F 75 73		Asynchronous	
00	Tiny Atom Token	0 = FALSE	
F3	End Name Token		
F2	Start Name Token		name-value
AB	Short Atom Token Header	Byte sequence; Length = 11	
4D 61 78 53 65 73 73 69 6F 6E 73		MaxSessions	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value

D0 12	Medium Atom Token Header	Byte sequence; Length = 18	
4D 61 78 41 75 74 68 65 6E 74 69 63 61 74 69 6F 6E 73		MaxAuthentications	
02	Tiny Atom Token	2	
F3	End Name Token		
F2	Start Name Token		name-value
D0 13	Medium Atom Token Header	Byte sequence; Length = 19	
4D 61 78 54 72 61 6E 73 61 63 74 69 6F 6E 4C 69 6D 69 74		MaxTransactionLimit	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
44 65 66 53 65 73 73 69 6F 6E 54 69 6D 65 6F 75 74		DefSessionTimeout	
83	Short Atom Token Header	Uinteger value; Length = 3	
01 D4 C0		120000	
F3	End Name Token		
F1	End List Token		Ends “Properties” list
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostProperties”	
F0	Start List Token		Starts “HostProperties” list
F2	Start Name Token		name-value
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
4D 61 78 43 6F 6D 50 61 63 6B 65 74 53 69 7A 65		MaxComPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
10 00		4096	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 50 61 63 6B 65 74 53 69 7A 65		MaxPacketSize	
82	Short Atom Token Header	Uinteger value; Length = 2	
0F EC		4076	
F3	End Name Token		
F2	Start Name Token		name-value
AF	Short Atom Token Header	Byte sequence; Length = 15	
4D 61 78 49 6E 64 54 6F 6B 65 6E 53 69 7A 65		MaxIndTokenSize	
82	Short Atom Token Header	Uinteger value; Length = 2	

0F C8		4040	
F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 50 61 63 6B 65 74 73		MaxPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AD	Short Atom Token Header	Byte sequence; Length = 13	
4D 61 78 53 75 62 70 61 63 6B 65 74 73		MaxSubPackets	
01	Tiny Atom Token	1	
F3	End Name Token		
F2	Start Name Token		name-value
AA	Short Atom Token Header	Byte sequence; Length = 10	
4D 61 78 4D 65 74 68 6F 64 73		MaxMethods	
01	Tiny Atom Token	1	
F3	End Name Token		
F1	End List Token		Ends “HostProperties” list
F3	End Name Token		Ends “HostProperties” parameter
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

3.2.2 Common Commands and Responses

The commands defined in this section are commonly used, and occur in most communications between the host and the TPer. The commands defined here are referenced from relevant sections, rather than repeated in each instance. The ability to reference a single source for these commands are based on the common elements described in 3.2.

3.2.2.1 SyncSession Response

```
session[0:0] <- SMUID.SyncSession[HostSessionID : 0x00000001, SPSessionID :
0x00001001]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00000000 00000000 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000025 F8A80000 00000000
0040 00FFA800 00000000 00FF03F0 84000000
0050 01840000 1001F1F9 F0000000 F1000000
0060 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 4 SyncSession

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 25	Length	37	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 03	Method UID	SyncSession Method UID	
F0	Start List Token		Begins parameter list
84	Short Atom Token Header	Uinteger value; Length = 4	
00 00 00 01	Required Parameter: HostSessionID	<1>	Echo Host Number
84	Short Atom Token Header	Uinteger value; Length = 4	
00 00 10 01	Required Parameter: SPSessionID	<1001>	Number assigned by storage device
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

3.2.2.2 Set Results

```
session[TSN:HSN] <- [ ]
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 5 Set Method Results

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Packet			
00 00 10 02 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

3.2.2.3 Ending the Session

3.2.2.3.1 Send End of Session Token

```
session[TSN:HSN] -> EOS
0000 00000000 07FE0000 00000000 00000000
0010 00000028 00001001 00000001 00000000
0020 00000000 00000000 00000010 00000000
0030 00000000 00000001 FA000000 00000000
0040 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 6 EndSession

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4

00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 10	Length	16	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 01	Length	1	uinteger_4
Data Payload			
FA	End of Session Token		Ends the Session
00 00 00	Pad		Included in ComPacket and Packet lengths

3.2.2.3.2 End of Session Response

```
session[TSN:HSN] <- EOS
0000 00000000 07FE0000 00000000 00000000
0010 00000028 00001001 00000001 00000000
0020 00000000 00000000 00000010 00000000
0030 00000000 00000001 FA000000 00000000
0040 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 7 EndSession – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 10	Length	16	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 01	Length	1	uinteger_4
Data Payload			
FA	End of Session Token		Ends the Session
00 00 00	Pad		Included in ComPacket and Packet lengths

3.2.3 Taking ownership of the storage device

3.2.3.1 Open a session to the Admin SP as the Anybody authority

3.2.3.1.1 StartSession – Admin SP

```
session[0:0] -> SMUID.StartSession[HostSessionID : 0x01, SPID : AdminSP_UID, Write :
True]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00000000 00000000 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000026 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000101F1 F9F00000 00F10000
0060 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 8 StartSession – Admin SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 26	Length	38	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	

00 00 02 05 00 00 00 01	Required Parameter: SPID	<AdminSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00	Pad		Included in Packet and ComPacket lengths

3.2.3.1.2 SyncSession – Admin SP

See 3.2.2.1.

3.2.3.2 Get the MSID's PIN value from the C_PIN table

```
session[TSN:HSN] -> C_PIN_MSID_UID.Get[Cellblock : [startColumn = PIN, endColumn =
PIN]]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000025 F8A80000 000B0000
0040 8402A800 00000600 000016F0 F0F20303
0050 F3F20403 F3F1F1F9 F0000000 F1000000
0060 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 9 Get – C_PIN_MSID

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 25	Length	37	uinteger_4

Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 0B 00 00 84 02	Invoking UID	C_PIN_MSID UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 16	Method UID	Get Method UID	
F0	Start List Token		Begins parameter list
F0	Start List Token		Begins cell block
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“startColumn”	
03	Tiny Atom Token: Value	<PIN>	
F3	End Name Token		
F2	Start Name Token		name-value
04	Tiny Atom Token: Name	“endColumn”	
03	Tiny Atom Token: Value	<PIN>	
F3	End Name Token		
F1	End List Token		Ends cell block
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- [ [PIN = <MSID_password>] ]
0000 00000000 07FE0000 00000000 00000000
0010 00000044 00001001 00000001 00000000
0020 00000000 00000000 0000002C 00000000
0030 00000000 0000001D F0F0F203 AF3C4D53
0040 49445F70 61737377 6F72643E F3F1F1F9
0050 F0000000 F1000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 10 Get – C_PIN_MSID – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4

00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1D	Length	29	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F0	Start List Token		Start of first row of results
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“PIN”	
AF	Short Atom Token Header	Byte sequence; Length = 15	
3C 4D 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<MSID_password>	Example password value
F3	End Name Token		
F1	End List Token		End of first row of results
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

3.2.3.3 Close the session

See 3.2.2.3.

3.2.3.4 Open a session to the Admin SP as the SID authority using the <MSID_password>

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : AdminSP_UID, Write :
True, HostChallenge = <MSID_password>, HostSigningAuthority = SID_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000045 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000101F2 00AF3C4D 5349445F
0060 70617373 776F7264 3EF3F203 A8000000
0070 09000000 06F3F1F9 F0000000 F1000000
0080 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 11 StartSession – Admin SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4

00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 45	Length	69	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 01	Required Parameter: SPID	<AdminSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
AF	Short Atom Token Header	Byte sequence; Length = 15	
3C 4D 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<MSID_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 00 00 06	Value	<SID_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.3.5 Set the <new_SID_password> value in the SID’s C_PIN credential PIN column

```
session[TSN:HSN] -> C_PIN_SID_UID.Set[Values = [PIN = <new_SID_password>]]
0000 00000000 07FE0000 00000000 00000000
0010 0000005C 00001001 00000001 00000000
0020 00000000 00000000 00000044 00000000
0030 00000000 00000037 F8A80000 000B0000
0040 0001A800 00000600 000017F0 F201F0F2
0050 03D0123C 6E65775F 5349445F 70617373
0060 776F7264 3EF3F1F3 F1F9F000 0000F100
0070 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 12 Set – C_PIN_SID

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
		Packet	
00 00 10 02 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 37	Length	55	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 0B 00 00 00 01	Invoking UID	C_PIN table SID row UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“PIN”	
D0 12	Medium Atom Token Header	Byte sequence; Length = 18	

3C 6E 65 77 5F 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<new_SID_password>	Example password value
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.3.6 Close the session

See 3.2.2.3.

3.2.4 Activating the Locking SP

3.2.4.1 Open a session to the Admin SP as the SID authority

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : AdminSP_UID, Write :
TRUE, HostChallenge = <new_SID_password>, HostSigningAuthority = SID_UID]
0000 00000000 07FE0000 00000000 00000000
0010 00000070 00000000 00000000 00000000
0020 00000000 00000000 00000058 00000000
0030 00000000 00000049 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000101F2 00D0123C 6E65775F
0060 5349445F 70617373 776F7264 3EF3F203
0070 A8000000 09000000 06F3F1F9 F0000000
0080 F1000000 00000000 00000000 00000000
0090 00000000 00000000 00000000 00000000
                                ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 13 StartSession – Admin SP

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 70	Length	112	uinteger_4
		Packet	
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2

00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 58	Length	88	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 49	Length	73	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 01	Required Parameter: SPID	<AdminSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 12	Medium Atom Token Header	Byte sequence; Length = 18	
3C 6E 65 77 5F 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<new_SID_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 00 00 06	Value	<SID_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.4.2 Determine the life cycle state of the Locking SP

Note: In this example, the Locking SP is in the Manufactured-Inactive state. Other TPers may have a Locking SP in the Manufactured state, in which case invoking the Activate method is not necessary.

```
session[TSN:HSN] -> LockingSP_UID.Get[Cellblock : [startColumn = LifeCycle,
endColumn = LifeCycle]]
```

```
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000025 F8A80000 02050000
0040 0002A800 00000600 000016F0 F0F20306
0050 F3F20406 F3F1F1F9 F0000000 F1000000
0060 00000000 00000000 00000000 00000000
                          . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 14 Get – LockingSP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 25	Length	37	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Invoking UID	LockingSP UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 16	Method UID	Get Method UID	
F0	Start List Token		Begins parameter list
F0	Start List Token		Begins cell block
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“startColumn”	
06	Tiny Atom Token: Value	<LifeCycle>	
F3	End Name Token		
F2	Start Name Token		name-value
04	Tiny Atom Token: Name	“endColumn”	
06	Tiny Atom Token: Value	<LifeCycle>	
F3	End Name Token		
F1	End List Token		Ends cell block
F1	End List Token		Ends parameter list

F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- [ [LifeCycle = Manufactured-Inactive] ]
0000 00000000 07FE0000 00000000 00000000
0010 00000034 00001001 00000001 00000000
0020 00000000 00000000 0000001C 00000000
0030 00000000 0000000E F0F0F206 08F3F1F1
0040 F9F00000 00F10000 00000000 00000000
0050 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 15 Get – LockingSP – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 1C	Length	28	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 0E	Length	14	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F0	Start List Token		Start of first row of results
F2	Start Name Token		name-value
06	Tiny Atom Token: Name	“LifeCycle”	
08	Tiny Atom Token: Value	Manufactured-Inactive	
F3	End Name Token		
F1	End List Token		End of first row of results
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

00 00	Pad		Included in ComPacket and Packet lengths

3.2.4.3 Activate the Locking SP by using the Activate method on the Locking SP object in the Admin SP

Note: After the Locking SP is activated, the PIN for Admin1 in the Locking SP will be set to the current SID PIN, which is “<new_SID_password>”.

```
session[TSN:HSN] -> LockingSP_UID.Activate[]
0000 00000000 07FE0000 00000000 00000000
0010 00000040 00001001 00000001 00000000
0020 00000000 00000000 00000028 00000000
0030 00000000 0000001B F8A80000 02050000
0040 0002A800 00000600 000203F0 F1F9F000
0050 0000F100 00000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 16 Activate – LockingSP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1B	Length	27	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Invoking UID	LockingSP_UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 02 03	Method UID	Activate UID	
F0	Start List Token		Begins parameter list
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method

F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- []
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                        ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 17 Activate – LockingSP – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

3.2.4.4 Close the session

See 3.2.2.3.

3.2.5 Changing the Admin1 PIN in the Locking SP and adding users

3.2.5.1 Open a session to the Locking SP as Admin1

Note: When the Locking SP was activated, the current SID PIN was copied into the Admin1 PIN.

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <new_SID_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 00000070 00000000 00000000 00000000
0020 00000000 00000000 00000058 00000000
0030 00000000 00000049 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0123C 6E65775F
0060 5349445F 70617373 776F7264 3EF3F203
0070 A8000000 09000100 01F3F1F9 F0000000
0080 F1000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 18 StartSession – Locking SP

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 70	Length	112	uinteger_4
	Packet		
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 58	Length	88	uinteger_4
	Data SubPacket		
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 49	Length	73	uinteger_4
	Data Payload		
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value

00	Tiny Atom Token: Name	"HostChallenge"	
D0 12	Medium Atom Token Header	Byte sequence; Length = 18	
3C 6E 65 77 5F 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<new_SID_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	"HostSigningAuthority"	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1 00 00 00	Method Status List Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.5.2 Set the <Admin1_password> value in Admin1's C_PIN credential PIN column

```
session[TSN:HSN] -> C_PIN_Admin1_UID.Set[Values = [PIN = <Admin1_password>]]
0000 00000000 07FE0000 00000000 00000000
0010 0000005C 00001001 00000001 00000000
0020 00000000 00000000 00000044 00000000
0030 00000000 00000036 F8A80000 000B0001
0040 0001A800 00000600 000017F0 F201F0F2
0050 03D0113C 41646D69 6E315F70 61737377
0060 6F72643E F3F1F3F1 F9F00000 00F10000
0070 00000000 00000000 00000000 00000000
                        ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 19 Set – C_PIN_Admin1

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Packet			
00 00 10 02 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6

00 00	Kind	0's	uinteger_2
00 00 00 36	Length	54	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 0B 00 01 00 01	Invoking UID	C_PIN_Admin1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	"Values"	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	"PIN"	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.5.3 Enable the User1 authority

```
session[TSN:HSN] -> User1_UID.Set[Values = [Enabled = TRUE]]
0000 00000000 07FE0000 00000000 00000000
0010 00000048 00001001 00000001 00000000
0020 00000000 00000000 00000030 00000000
0030 00000000 00000024 F8A80000 00090003
0040 0001A800 00000600 000017F0 F201F0F2
0050 0501F3F1 F3F1F9F0 000000F1 00000000
0060 00000000 00000000 00000000 00000000
                        ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 20 Set – User1

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4

07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 48	Length	72	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 30	Length	48	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 24	Length	36	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Invoking UID	Locking SP Authority Table User1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
05	Tiny Atom Token: Name	“Enabled”	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.5.4 Change the password for User1

```
session[TSN:HSN] -> C_PIN_User1_UID.Set[Values = [PIN = <User1_password>]]
0000 00000000 07FE0000 00000000 00000000
0010 0000005C 00001001 00000001 00000000
0020 00000000 00000000 00000044 00000000
0030 00000000 00000035 F8A80000 000B0003
0040 0001A800 00000600 000017F0 F201F0F2
0050 03D0103C 55736572 315F7061 7373776F
```

```
0060 72643EF3 F1F3F1F9 F0000000 F1000000
0070 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 21 Set – C_PIN_User1

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 35	Length	53	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 0B 00 03 00 01	Invoking UID	C_PIN table CPIN_User1 row UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“PIN”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<User1_password>	Example password value
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.5.5 Enable the User2 authority

```
session[TSN:HSN] -> User2_UID.Set[Values = [Enabled = TRUE]]
0000 00000000 07FE0000 00000000 00000000
0010 00000048 00001001 00000001 00000000
0020 00000000 00000000 00000030 00000000
0030 00000000 00000024 F8A80000 00090003
0040 0002A800 00000600 000017F0 F201F0F2
0050 0501F3F1 F3F1F9F0 000000F1 00000000
0060 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 22 Set – User2

Bytes	**Purpose**	**Value**	**Notes**
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 48	Length	72	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 30	Length	48	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 24	Length	36	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Invoking UID	Locking SP Authority Table User2 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	

F0	Start List Token		
F2	Start Name Token		name-value
05	Tiny Atom Token: Name	“Enabled”	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.5.6 Change the password for User2

```
session[TSN:HSN] -> C_PIN_User2_UID.Set[Values = [PIN = <User2_password>]]
0000 00000000 07FE0000 00000000 00000000
0010 0000005C 00001001 00000001 00000000
0020 00000000 00000000 00000044 00000000
0030 00000000 00000035 F8A80000 000B0003
0040 0002A800 00000600 000017F0 F201F0F2
0050 03D0103C 55736572 325F7061 7373776F
0060 72643EF3 F1F3F1F9 F0000000 F1000000
0070 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 23 Set – C_PIN_User2

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 35	Length	53	uinteger_4
Data Payload			

F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 0B 00 03 00 02	Invoking UID	C_PIN table CPIN_User2 row UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“PIN”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 32 5F 70 61 73 73 77 6F 72 64 3E	Value	<User2_password>	Example password value
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.5.7 Close the session

See 3.2.2.3.

3.2.6 Configuring Locking Objects (LBA ranges)

3.2.6.1 Open a session to the Locking SP as Admin1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <Admin1_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000048 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0113C 41646D69
0060 6E315F70 61737377 6F72643E F3F203A8
0070 00000009 00010001 F3F1F9F0 000000F1
0080 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 24 StartSession – Locking SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 48	Length	72	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list

F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

SyncSession response – see 3.2.2.1.

3.2.6.2 Configure the range and enable read and write locking by changing RangeStart, RangeLength, ReadLockEnabled and WriteLockEnabled for Locking_Range1

```
session[TSN:HSN] -> Locking_Range1_UID.Set[Values = [RangeStart = 1000, RangeLength
= 1501, ReadLockEnabled = TRUE, WriteLockEnabled = TRUE]]
0000 00000000 07FE0000 00000000 00000000
0010 00000058 00001001 00000001 00000000
0020 00000000 00000000 00000040 00000000
0030 00000000 00000034 F8A80000 08020003
0040 0001A800 00000600 000017F0 F201F0F2
0050 038203E8 F3F20482 05DDF3F2 0501F3F2
0060 0601F3F1 F3F1F9F0 000000F1 00000000
0070 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 25 Set – Locking_Range1

Bytes	**Purpose**	**Value**	**Notes**
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 58	Length	88	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 34	Length	52	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 02 00 03 00 01	Invoking UID	Locking_Range1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		

F2	Start Name Token		name-value
03	Tiny Atom Token: Name	"RangeStart"	
82	Short Atom Token Header	Byte sequence; Length = 2	
03 E8	Value	<1000>	
F3	End Name Token		
F2	Start Name Token		name-value
04	Tiny Atom Token: Name	"RangeLength"	
82	Short Atom Token Header	Byte sequence; Length = 2	
05 DD	Value	<1501>	
F3	End Name Token		
F2	Start Name Token		name-value
05	Tiny Atom Token: Name	"ReadLockEnabled"	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F2	Start Name Token		name-value
06	Tiny Atom Token: Name	"WriteLockEnabled"	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.6.3 Retrieve the UID of the range's media encryption key

Note: In this example, the TPer utilizes AES-256, and therefore returns the UID for K_AES_256_Range1_Key. Some TPers utilize AES-128, and therefore would return the UID for K_AES_128_Range1_Key.

```
session[TSN:HSN] -> Locking_Range1_UID.Get[Cellblock : [startColumn = ActiveKey,
endColumn = ActiveKey] ]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000025 F8A80000 08020003
0040 0001A800 00000600 000016F0 F0F2030A
0050 F3F2040A F3F1F1F9 F0000000 F1000000
0060 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 26 Get – Locking_Range1

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4

00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 25	Length	37	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 02 00 03 00 01	Invoking UID	Locking_Range1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 16	Method UID	Get Method UID	
F0	Start List Token		Begins parameter list
F0	Start List Token		Begins cell block
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“startColumn”	
0A	Tiny Atom Token: Value	<ActiveKey>	
F3	End Name Token		
F2	Start Name Token		name-value
04	Tiny Atom Token: Name	“endColumn”	
0A	Tiny Atom Token: Value	<ActiveKey>	
F3	End Name Token		
F1	End List Token		Ends cell block
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- [ [ActiveKey = K_AES_256_Range1_Key_UID] ]
0000 00000000 07FE0000 00000000 00000000
0010 0000003C 00001001 00000001 00000000
0020 00000000 00000000 00000024 00000000
0030 00000000 00000016 F0F0F20A A8000008
0040 06000300 01F3F1F1 F9F00000 00F10000
0050 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
```

```
01F0 00000000 00000000 00000000 00000000
```

Table 27 Get – Locking_Range1 – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 3C	Length	60	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 24	Length	36	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 16	Length	22	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F0	Start List Token		Start of first row of results
F2	Start Name Token		name-value
0A	Tiny Atom Token: Name	<ActiveKey>	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 06 00 03 00 01	Value	K_AES_256_Range1_Key UID	In this example, the TPer utilizes AES-256. Other TPers may utilize AES-128.
F3	End Name Token		
F1	End List Token		End of first row of results
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		
00 00	Pad		Included in ComPacket and Packet lengths

3.2.6.4 Perform a Secure Erase of the range

```
session[TSN:HSN] -> K_AES_256_Range1_Key_UID.Genkey[]
0000 00000000 07FE0000 00000000 00000000
0010 00000040 00001001 00000001 00000000
0020 00000000 00000000 00000028 00000000
0030 00000000 0000001B F8A80000 08060003
```

```
0040 0001A800 00000600 000010F0 F1F9F000
0050 0000F100 00000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                            . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 28 Genkey – K_AES_256_Range1_Key

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1B	Length	27	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 06 00 03 00 01	Invoking UID	K_AES_256_Range1_Key UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 10	Method UID	Genkey Method UID	
F0	Start List Token		Begins parameter list
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

```
Results
session[TSN:HSN] <- []
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                            . . .
```

```
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 29 Genkey – K_AES_256_Range1_Key – Response

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
	Packet		
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
	Data SubPacket		
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
	Data Payload		
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

3.2.6.5 Give access to multiple users to read-unlock the range (User1 and User2)

```
session[TSN:HSN] -> ACE_Locking_Range1_Set_RdLocked_UID.Set[Values = [BooleanExpr =
[User1_UID | User2_UID]]]
0000 00000000 07FE0000 00000000 00000000
0010 00000074 00001001 00000001 00000000
0020 00000000 00000000 0000005C 00000000
0030 00000000 0000004D F8A80000 00080003
0040 E001A800 00000600 000017F0 F201F0F2
0050 03F0F2A4 00000C05 A8000000 09000300
0060 01F3F2A4 00000C05 A8000000 09000300
0070 02F3F2A4 0000040E 01F3F1F3 F1F3F1F9
0080 F0000000 F1000000 00000000 00000000
0090 00000000 00000000 00000000 00000000
                    ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 30 Set – ACE_Locking_Range1_Set_RdLocked

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4

07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 74	Length	116	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 4D	Length	77	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 08 00 03 E0 01	Invoking UID	ACE_Locking_Range1_Set_RdLocked UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“BooleanExpr”	
F0	Start List Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Value	<User2 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 04 0E	Name	“Half-UID – boolean_ACE”	
01	Tiny Atom Token: Value	<1> (OR)	

F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.6.6 Give access to multiple users to write-unlock the range (User1 and User2)

```
session[TSN:HSN] -> ACE_Locking_Range1_Set_WrLocked_UID.Set[Values = [BooleanExpr =
[User1_UID | User2_UID]]]
0000 00000000 07FE0000 00000000 00000000
0010 00000074 00001001 00000001 00000000
0020 00000000 00000000 0000005C 00000000
0030 00000000 0000004D F8A80000 00080003
0040 E801A800 00000600 000017F0 F201F0F2
0050 03F0F2A4 00000C05 A8000000 09000300
0060 01F3F2A4 00000C05 A8000000 09000300
0070 02F3F2A4 0000040E 01F3F1F3 F1F3F1F9
0080 F0000000 F1000000 00000000 00000000
0090 00000000 00000000 00000000 00000000
                    ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 31 Set – ACE_Locking_Range1_Set_WrLocked

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 74	Length	116	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6

00 00	Kind	0's	uinteger_2
00 00 00 4D	Length	77	uinteger_4
	Data Payload		
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 08 00 03 E8 01	Invoking UID	ACE_Locking_Range1_ Set_WrLocked UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“BooleanExpr”	
F0	Start List Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Value	<User2 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 04 0E	Name	“Half-UID – boolean_ACE”	
01	Tiny Atom Token: Value	<1> (OR)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.6.7 Lock for read and write, by setting ReadLocked and WriteLocked for this range to TRUE

```
session[TSN:HSN] -> Locking_Range1_UID.Set[Values = [ReadLocked = TRUE, WriteLocked
= TRUE]]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000028 F8A80000 08020003
0040 0001A800 00000600 000017F0 F201F0F2
0050 0701F3F2 0801F3F1 F3F1F9F0 000000F1
0060 00000000 00000000 00000000 00000000
                    ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 32 Set – Locking_Range1

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 28	Length	40	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 02 00 03 00 01	Invoking UID	Locking_Range1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	"Values"	
F0	Start List Token		
F2	Start Name Token		name-value
07	Tiny Atom Token: Name	"ReadLocked"	

01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F2	Start Name Token		name-value
08	Tiny Atom Token: Name	“WriteLocked”	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.6.8 Close the session

See 3.2.2.3.

3.2.7 Unlocking ranges

3.2.7.1 Open a session to the Locking SP as User1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <User1_password>, HostSigningAuthority = User1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000047 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0103C 55736572
0060 315F7061 7373776F 72643EF3 F203A800
0070 00000900 030001F3 F1F9F000 0000F100
0080 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 33 StartSession – Locking SP

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
	Packet		
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2

00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 47	Length	71	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<User1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.7.2 Unlock a range by setting the Locked columns in the Locking table to FALSE

```
session[TSN:HSN] -> Locking_Range1_UID.Set[Values = [ReadLocked = FALSE, WriteLocked
= FALSE]]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
```

```
0020 00000000 00000000 00000034 00000000
0030 00000000 00000028 F8A80000 08020003
0040 0001A800 00000600 000017F0 F201F0F2
0050 0700F3F2 0800F3F1 F3F1F9F0 000000F1
0060 00000000 00000000 00000000 00000000
                          . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 34 Set – Locking_Range1

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 28	Length	40	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 02 00 03 00 01	Invoking UID	Locking_Range1 UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
07	Tiny Atom Token: Name	“ReadLocked”	
00	Tiny Atom Token: Value	<0> (False)	
F3	End Name Token		
F2	Start Name Token		name-value
08	Tiny Atom Token: Name	“WriteLocked”	
00	Tiny Atom Token: Value	<0> (False)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		

F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.7.3 Close the session

See 3.2.2.3.

3.2.8 Erasing a range

3.2.8.1 Open a session to the Locking SP as Admin1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <Admin1_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000048 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0113C 41646D69
0060 6E315F70 61737377 6F72643E F3F203A8
0070 00000009 00030001 F3F1F9F0 000000F1
0080 00000000 00000000 00000000 00000000
                             ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 35 StartSession – Admin SP

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
		Packet	
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 48	Length	72	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	

00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

SyncSession response – see 3.2.2.1.

3.2.8.2 Invoke the GenKey method on the media encryption key associated with one of the ranges

Note: In this example, the TPer utilizes AES-256, and therefore GenKey is invoked on the UID for K_AES_256_Range1_Key. To retrieve the UID of the key associated with Locking_Range1, see 3.2.6.3.

```
session[TSN:HSN] -> K_AES_256_Range1_Key_UID.Genkey[]
0000 00000000 07FE0000 00000000 00000000
0010 00000040 00001001 00000001 00000000
0020 00000000 00000000 00000028 00000000
0030 00000000 0000001B F8A80000 08060003
0040 0001A800 00000600 000010F0 F1F9F000
0050 0000F100 00000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 36 Genkey – K_AES_256_Range1_Key

Bytes	**Purpose**	**Value**	**Notes**
ComPacket			

00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1B	Length	27	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 06 00 03 00 01	Invoking UID	K_AES_256_Range1_Key UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 10	Method UID	Genkey Method UID	
F0	Start List Token		Begins parameter list
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- []
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                    ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 37 Genkey – K_AES_256_Range1_Key – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4

00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

3.2.8.3 Close the session

See 3.2.2.3.

3.2.9 Enabling MBR Shadowing

3.2.9.1 Open a session to the Locking SP as Admin1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <Admin1_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000048 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0113C 41646D69
0060 6E315F70 61737377 6F72643E F3F203A8
0070 00000009 00010001 F3F1F9F0 000000F1
0080 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 38 StartSession – Locking SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4

00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 48	Length	72	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

SyncSession response – see 3.2.2.1.

3.2.9.2 Give access to User1 and User2 for setting the MBR Shadowing “Done” flag

```
session[TSN:HSN] -> ACE_MBRControl_Set_Done_UID.Set[Values = [BooleanExpr =
[User1_UID | User2_UID]]]
0000 00000000 07FE0000 00000000 00000000
0010 00000074 00001001 00000001 00000000
0020 00000000 00000000 0000005C 00000000
```

```
0030 00000000 0000004D F8A80000 00080003
0040 F801A800 00000600 000017F0 F201F0F2
0050 03F0F2A4 00000C05 A8000000 09000300
0060 01F3F2A4 00000C05 A8000000 09000300
0070 02F3F2A4 0000040E 01F3F1F3 F1F3F1F9
0080 F0000000 F1000000 00000000 00000000
0090 00000000 00000000 00000000 00000000
                            . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 39 Set – ACE_MBRControl_Set_Done

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 74	Length	116	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 4D	Length	77	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 08 00 03 F8 01	Invoking UID	ACE_MBRControl_Set_ Done UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“BooleanExpr”	
F0	Start List Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1 UID>	

F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Value	<User2 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 04 0E	Name	“Half-UID – boolean_ACE”	
01	Tiny Atom Token: Value	<1> (OR)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.9.3 Set the MBR table

Note: ComPackets are limited in size, and therefore multiple invocations of `Set` may be necessary for writing all data to the MBR table.

```
session[TSN:HSN] -> MBR_UID.Set[Where = 0, Values = “<Master_Boot_Record_shadow>”]
0000 00000000 07FE0000 00000000 00000000
0010 00000064 00001001 00000001 00000000
0020 00000000 00000000 0000004C 00000000
0030 00000000 0000003F F8A80000 08040000
0040 0000A800 00000600 000017F0 F20000F3
0050 F201D01B 3C4D6173 7465725F 426F6F74
0060 5F526563 6F72645F 73686164 6F773EF3
0070 F1F9F000 0000F100 00000000 00000000
0080 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 40 Set – MBR

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4

00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 64	Length	100	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 3F	Length	63	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 04 00 00 00 00	Invoking UID	MBR UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	"Where"	
00	Tiny Atom Token: Value	<0>	
F3	End Name Token		
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	"Values"	
D0 1B	Medium Atom Token Header	Byte Sequence; Length = 27	
3C 4D 61 73 74 65 72 5F 42 6F 6F 74 5F 52 65 63 6F 72 64 5F 73 68 61 64 6F 77 3E	Value	<Master_Boot_Record_shadow>	Example data
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.9.4 Enable the MBR Shadowing feature

```
session[TSN:HSN] -> MBRControl_UID.Set[Values = [Enable = TRUE]]
0000 00000000 07FE0000 00000000 00000000
```

```
0010 00000048 00001001 00000001 00000000
0020 00000000 00000000 00000030 00000000
0030 00000000 00000024 F8A80000 08030000
0040 0001A800 00000600 000017F0 F201F0F2
0050 0101F3F1 F3F1F9F0 000000F1 00000000
0060 00000000 00000000 00000000 00000000
                          . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 41 Set – MBRControl

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 48	Length	72	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 30	Length	48	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 24	Length	36	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 03 00 00 00 01	Invoking UID	MBRControl UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Enable”	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.9.5 Close the session

See 3.2.2.3.

3.2.10 Un-shadowing the MBR

3.2.10.1 Open a session to the Locking SP as User1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <User1_password>, HostSigningAuthority = User1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000047 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0103C 55736572
0060 315F7061 7373776F 72643EF3 F203A800
0070 00000900 030001F3 F1F9F000 0000F100
0080 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 42 StartSession – Locking SP

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
		Packet	
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 47	Length	71	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	

F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<User1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.10.2 Un-shadow the MBR

```
session[TSN:HSN] -> MBRControl_UID.Set[Values = [Done = TRUE]]
00 00 00 00 07 FE 00 00 00 00 00 00 00 00 00 00
00 00 00 48 00 00 10 01 00 00 00 01 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 30 00 00 00 00
00 00 00 00 00 00 00 24 F8 A8 00 00 08 03 00 00
00 01 A8 00 00 00 06 00 00 00 17 F0 F2 01 F0 F2
02 01 F3 F1 F3 F1 F9 F0 00 00 00 F1 00 00 00 00
```

Table 43 Set – MBRControl

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 48	Length	72	uinteger_4
	Packet		
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8

00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 30	Length	48	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 24	Length	36	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 08 03 00 00 00 01	Invoking UID	MBRControl UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
02	Tiny Atom Token: Name	“Done”	
01	Tiny Atom Token: Value	<1> (True)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

Results

See 3.2.2.2.

3.2.10.3 Close the session

See 3.2.2.3.

3.2.11 Reverting the TPer

3.2.11.1 Open a session to the Admin SP as SID

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : AdminSP_UID, Write :
TRUE, HostChallenge = <new_SID_password>, HostSigningAuthority = SID_UID]
0000 00000000 07FE0000 00000000 00000000
0010 00000070 00000000 00000000 00000000
0020 00000000 00000000 00000058 00000000
0030 00000000 00000049 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000101F2 00D0123C 6E65775F
0060 5349445F 70617373 776F7264 3EF3F203
0070 A8000000 09000000 06F3F1F9 F0000000
0080 F1000000 00000000 00000000 00000000
```

```
0090 00000000 00000000 00000000 00000000
                          . . .
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 44 StartSession – Admin SP

Bytes	Purpose	Value	Notes
	ComPacket		
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 70	Length	112	uinteger_4
	Packet		
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 58	Length	88	uinteger_4
	Data SubPacket		
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 49	Length	73	uinteger_4
	Data Payload		
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 01	Required Parameter: SPID	<AdminSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 12	Medium Atom Token Header	Byte sequence; Length = 18	
3C 6E 65 77 5F 53 49 44 5F 70 61 73 73 77 6F 72 64 3E	Value	<new_SID_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value

03	Tiny Atom Token: Name	"HostSigningAuthority"	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 00 00 06	Value	<SID_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.11.2 Revert the TPer

```
session[TSN:HSN] -> AdminSP_UID.Revert[]
0000 00000000 07FE0000 00000000 00000000
0010 00000040 00001001 00000001 00000000
0020 00000000 00000000 00000028 00000000
0030 00000000 0000001B F8A80000 02050000
0040 0001A800 00000600 000202F0 F1F9F000
0050 0000F100 00000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                            ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 45 Revert – AdminSP

Bytes	**Purpose**	**Value**	**Notes**
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1B	Length	27	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 01	Invoking UID	AdminSP UID	

A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 02 02	Method UID	Revert Method UID	
F0	Start List Token		Begins parameter list
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- []
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 46 Revert – AdminSP – Response

Bytes	**Purpose**	**Value**	**Notes**
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

Note: The TPer aborts the session immediately after reporting the Revert result.

3.2.12 Reverting the Locking SP

3.2.12.1 Open a session to the Locking SP as Admin1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <Admin1_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000048 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0113C 41646D69
0060 6E315F70 61737377 6F72643E F3F203A8
0070 00000009 00010001 F3F1F9F0 000000F1
0080 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 47 StartSession – Locking SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0's	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 48	Length	72	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	

00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

SyncSession response – see 3.2.2.1.

3.2.12.2 Revert the Locking SP

Note: If the optional parameter, “KeepGlobalRangeKey”, is provided with a value of TRUE, then the media encryption key associated with Locking_GlobalRange is preserved. In this example, “KeepGlobalRangeKey” is not provided, and therefore all user data is cryptographically erased.

```
session[TSN:HSN] -> ThisSP.RevertSP[]
0000 00000000 07FE0000 00000000 00000000
0010 00000040 00001001 00000001 00000000
0020 00000000 00000000 00000028 00000000
0030 00000000 0000001B F8A80000 00000000
0040 0001A800 00000600 000011F0 F1F9F000
0050 0000F100 00000000 00000000 00000000
0060 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 48 RevertSP – ThisSP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 40	Length	64	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2

00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 28	Length	40	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 1B	Length	27	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 01	Invoking UID	This SP UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 11	Method UID	RevertSP Method UID	
F0	Start List Token		Begins parameter list
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- []
0000 00000000 07FE0000 00000000 00000000
0010 0000002C 00001001 00000001 00000000
0020 00000000 00000000 00000014 00000000
0030 00000000 00000008 F0F1F9F0 000000F1
0040 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 49 RevertSP – ThisSP – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 2C	Length	44	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 14	Length	20	uinteger_4
Data SubPacket			

00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 08	Length	8	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

Note: The TPer aborts the session immediately after reporting the RevertSP result.

3.2.13 Using the DataStore table

3.2.13.1 Open a session to the Locking SP as Admin1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <Admin1_password>, HostSigningAuthority = Admin1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000048 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0113C 41646D69
0060 6E315F70 61737377 6F72643E F3F203A8
0070 00000009 00010001 F3F1F9F0 000000F1
0080 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 50 StartSession – Locking SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 48	Length	72	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	

00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 11	Medium Atom Token Header	Byte sequence; Length = 17	
3C 41 64 6D 69 6E 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<Admin1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 01 00 01	Value	<Admin1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

SyncSession response – see 3.2.2.1.

3.2.13.2 Give User1 write access to the DataStore table

```
session[TSN:HSN] -> ACE_DataStore_Set_All_UID.Set[Values = [BooleanExpr =
[User1_UID]]]
0000 00000000 07FE0000 00000000 00000000
0010 0000005C 00001001 00000001 00000000
0020 00000000 00000000 00000044 00000000
0030 00000000 00000035 F8A80000 00080003
0040 FC01A800 00000600 000017F0 F201F0F2
0050 03F0F2A4 00000C05 A8000000 09000300
0060 01F3F1F3 F1F3F1F9 F0000000 F1000000
0070 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 51 Set – ACE_DataStore_Set_All

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4

07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
Packet			
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 44	Length	68	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 35	Length	53	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 08 00 03 FC 01	Invoking UID	ACE_DataStore_Set_all UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	
F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“BooleanExpr”	
F0	Start List Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1 UID>	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.13.3 Give User1 and User2 read access to the DataStore table

```
session[TSN:HSN] -> ACE_DataStore_Get_All_UID.Set[Values = [BooleanExpr = [User1_UID
| User2_UID]]]
0000 00000000 07FE0000 00000000 00000000
0010 00000074 00001001 00000001 00000000
0020 00000000 00000000 0000005C 00000000
0030 00000000 0000004D F8A80000 00080003
0040 FC00A800 00000600 000017F0 F201F0F2
0050 03F0F2A4 00000C05 A8000000 09000300
0060 01F3F2A4 00000C05 A8000000 09000300
0060 02F3F2A4 0000040E 01F3F1F3 F1F3F1F9
0070 F0000000 F1000000 00000000 00000000
0080 00000000 00000000 00000000 00000000
                     ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 52 Set – ACE_DataStore_Get_All

Bytes	Purpose	Value	Notes
		ComPacket	
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 74	Length	116	uinteger_4
		Packet	
00 00 10 01 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 5C	Length	92	uinteger_4
		Data SubPacket	
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 4D	Length	77	uinteger_4
		Data Payload	
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 08 00 03 FC 00	Invoking UID	ACE_DataStore_Get_all UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“Values”	

F0	Start List Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“BooleanExpr”	
F0	Start List Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 0C 05	Name	“Half-UID – Authority_object_ref”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Value	<User2 UID>	
F3	End Name Token		
F2	Start Name Token		name-value
A4	Short Atom Token Header	Byte sequence; Length = 4	
00 00 04 0E	Name	“Half-UID – boolean_ACE”	
01	Tiny Atom Token: Value	<1> (OR)	
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.13.4 Close the session

See 3.2.2.3.

3.2.13.5 Open a session to the Locking SP as User1

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <User1_password>, HostSigningAuthority = User1_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000047 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0103C 55736572
0060 315F7061 7373776F 72643EF3 F203A800
```

```
0070 00000900 030001F3 F1F9F000 0000F100
0080 00000000 00000000 00000000 00000000
                                  ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 53 StartSession – Locking SP

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 47	Length	71	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 31 5F 70 61 73 73 77 6F 72 64 3E	Value	<User1_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	

A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 01	Value	<User1_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.13.6 Write data to the DataStore table

```
session[TSN:HSN] -> DataStore_UID.Set[Where = 0, Values =
“<data_to_be_stored_in_DataStore_table>”]
0000 00000000 07FE0000 00000000 00000000
0010 00000070 00001001 00000001 00000000
0020 00000000 00000000 00000058 00000000
0030 00000000 0000004A F8A80000 10010000
0040 0000A800 00000600 000017F0 F20000F3
0050 F201D026 3C646174 615F746F 5F62655F
0060 73746F72 65645F69 6E5F4461 74615374
0070 6F72655F 7461626C 653EF3F1 F9F00000
0080 00F10000 00000000 00000000 00000000
0090 00000000 00000000 00000000 00000000
                          ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 54 Set – DataStore

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 70	Length	112	uinteger_4
Packet			
00 00 10 02 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 58	Length	88	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 4A	Length	74	uinteger_4
Data Payload			
F8	Call Token		Begins method

A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 10 01 00 00 00 00	Invoking UID	DataStore UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 17	Method UID	Set Method UID	
F0	Start List Token		Begins parameter list
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	"Where"	
00	Tiny Atom Token: Value	<0>	
F3	End Name Token		
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	"Values"	
D0 26	Medium Atom Token Header	Byte sequence; Length = 38	
3C 64 61 74 61 5F 74 6F 5F 62 65 5F 73 74 6F 72 65 64 5F 69 6E 5F 44 61 74 61 53 74 6F 72 65 5F 74 61 62 6C 65 3E	Value	<data_to_be_stored_in_DataStore_table>	Example data
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00	Pad		Included in ComPacket and Packet lengths

Results

See 3.2.2.2.

3.2.13.7 Close the session

See 3.2.2.3.

3.2.13.8 Open a session to the Locking SP as User2

```
session[0:0] -> SMUID.StartSession[HostSessionID : HSN, SPID : LockingSP_UID, Write
: TRUE, HostChallenge = <User2_password>, HostSigningAuthority = User2_UID]
0000 00000000 07FE0000 00000000 00000000
0010 0000006C 00000000 00000000 00000000
0020 00000000 00000000 00000054 00000000
0030 00000000 00000047 F8A80000 00000000
0040 00FFA800 00000000 00FF02F0 01A80000
0050 02050000 000201F2 00D0103C 55736572
0060 325F7061 7373776F 72643EF3 F203A800
0070 00000900 030002F3 F1F9F000 0000F100
0080 00000000 00000000 00000000 00000000
                         ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 55 StartSession – Locking SP

Bytes	Purpose	Value	Notes
		ComPacket	

00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 6C	Length	108	uinteger_4
Packet			
00 00 00 00 00 00 00 00	Session	0's	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 47	Length	71	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 00 FF	Invoking UID	Session Manager Reserved UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 00 00 00 FF 02	Method UID	StartSession Method UID	
F0	Start List Token		Begins parameter list
01	Tiny Atom Token: Required Parameter: HostSessionID	<1>	Host Supplied Number
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 02 05 00 00 00 02	Required Parameter: SPID	<LockingSP UID>	UID of SP to which session is being opened
01	Tiny Atom Token: Required Parameter: Write	<1>	Read/Write Session
F2	Start Name Token		name-value
00	Tiny Atom Token: Name	“HostChallenge”	
D0 10	Medium Atom Token Header	Byte sequence; Length = 16	
3C 55 73 65 72 32 5F 70 61 73 73 77 6F 72 64 3E	Value	<User2_password>	Example password value
F3	End Name Token		
F2	Start Name Token		name-value
03	Tiny Atom Token: Name	“HostSigningAuthority”	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 09 00 03 00 02	Value	<User2_UID>	
F3	End Name Token		
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		

00	Pad		Included in ComPacket and Packet lengths

SyncSession response – see 3.2.2.1.

3.2.13.9 Read data from the DataStore table

```
session[TSN:HSN] -> DataStore_UID.Get[Cellblock : [startRow = 0, endRow = 37]]
0000 00000000 07FE0000 00000000 00000000
0010 0000004C 00001001 00000001 00000000
0020 00000000 00000000 00000034 00000000
0030 00000000 00000025 F8A80000 10010000
0040 0000A800 00000600 000016F0 F0F20100
0050 F3F20225 F3F1F1F9 F0000000 F1000000
0060 00000000 00000000 00000000 00000000
                       ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 56 Get – DataStore

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 4C	Length	76	uinteger_4
Packet			
00 00 10 03 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 34	Length	52	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 25	Length	37	uinteger_4
Data Payload			
F8	Call Token		Begins method
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 10 01 00 00 00 00	Invoking UID	DataStore UID	
A8	Short Atom Token Header	Byte sequence; Length = 8	
00 00 00 06 00 00 00 16	Method UID	Get Method UID	
F0	Start List Token		Begins parameter list
F0	Start List Token		Begins cell block
F2	Start Name Token		name-value
01	Tiny Atom Token: Name	“startRow”	
00	Tiny Atom Token: Value	<0>	

F3	End Name Token		
F2	Start Name Token		name-value
02	Tiny Atom Token: Name	“endRow”	
25	Tiny Atom Token: Value	<37>	
F3	End Name Token		
F1	End List Token		Ends cell block
F1	End List Token		Ends parameter list
F9	End of Data Token		Ends method
F0 00 00 00 F1	Method Status List		
00 00 00	Pad		Included in ComPacket and Packet lengths

Results

```
session[TSN:HSN] <- [“<data_to_be_stored_in_DataStore_table>”]
0000 00000000 07FE0000 00000000 00000000
0010 00000054 00001001 00000001 00000000
0020 00000000 00000000 0000003C 00000000
0030 00000000 00000030 F0D0273C 64617461
0040 5F746F5F 62655F73 746F7265 645F696E
0050 5F446174 6153746F 72655F74 61626C65
0060 3EF1F9F0 000000F1 00000000 00000000
0070 00000000 00000000 00000000 00000000
                                    ...
01E0 00000000 00000000 00000000 00000000
01F0 00000000 00000000 00000000 00000000
```

Table 57 Get – DataStore – Response

Bytes	Purpose	Value	Notes
ComPacket			
00 00 00 00	Reserved	0's	uinteger_4
07 FE 00 00	Extended ComID	07FE, 0	uinteger_4
00 00 00 00	OutstandingData	0's	uinteger_4
00 00 00 00	MinTransfer	0's	uinteger_4
00 00 00 54	Length	84	uinteger_4
Packet			
00 00 10 03 00 00 00 01	Session	1001:1	uinteger_8
00 00 00 00	SeqNumber	0	uinteger_4
00 00	Reserved	0’s	uinteger_2
00 00	AckType	0's	uinteger_2
00 00 00 00	Acknowledgement	0's	uinteger_4
00 00 00 3C	Length	60	uinteger_4
Data SubPacket			
00 00 00 00 00 00	Reserved	0's	uinteger_6
00 00	Kind	0's	uinteger_2
00 00 00 30	Length	48	uinteger_4
Data Payload			
F0	Start List Token		Start of results list
D0 26	Medium Atom Token Header	Byte sequence; Length = 38	

3C 64 61 74 61 5F 74 6F 5F 62 65 5F 73 74 6F 72 65 64 5F 69 6E 5F 44 61 74 61 53 74 6F 72 65 5F 74 61 62 6C 65 3E	Data	<data_to_be_stored_in_DataStore_table>	Example data
F1	End List Token		End of results list
F9	End of Data Token		
F0 00 00 00 F1	Method Status List		

3.2.13.10 Close the session

See 3.2.2.3.