

## KMIP Tape Library Profile Version 1.0

~~Committee Specification Draft 01 /  
Public Review Draft 01~~

**COMMENT CHANGES DRAFT**

09 January 2014

### Specification URIs

#### This version:

<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/csprd01/kmip-tape-lib-profile-v1.0-csprd01.doc> (Authoritative)  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/csprd01/kmip-tape-lib-profile-v1.0-csprd01.html>  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/csprd01/kmip-tape-lib-profile-v1.0-csprd01.pdf>

#### Previous version:

N/A

#### Latest version:

~~<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.doc>  
(Authoritative)  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.html>  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.pdf>  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.doc>  
(Authoritative)  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.html>  
<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.pdf>~~

### Technical Committee:

OASIS Key Management Interoperability Protocol (KMIP) TC

### Chairs:

Robert Griffin ([robert.griffin@rsa.com](mailto:robert.griffin@rsa.com)), EMC Corporation  
Subhash Sankuratripati ([Subhash.Sankuratripati@netapp.com](mailto:Subhash.Sankuratripati@netapp.com)), NetApp

### Editors:

Tim Hudson ([tjh@cryptsoft.com](mailto:tjh@cryptsoft.com)), Cryptsoft  
Stan Feather ([stan.feather@hp.com](mailto:stan.feather@hp.com)), Hewlett-Packard  
Rod Wideman ([rod.wideman@quantum.com](mailto:rod.wideman@quantum.com)), Quantum

### Related work:

This specification is related to:

- *Key Management Interoperability Protocol Profiles Version 1.0*. 01 October 2010. OASIS Standard. <http://docs.oasis-open.org/kmip/profiles/v1.0/os/kmip-profiles-1.0-os.html>.
- *Key Management Interoperability Protocol Specification Version 1.1*. 24 January 2013. OASIS Standard. <http://docs.oasis-open.org/kmip/spec/v1.1/os/kmip-spec-v1.1-os.html>.
- *Key Management Interoperability Protocol Specification Version 1.2*. Latest version. <http://docs.oasis-open.org/kmip/spec/v1.2/kmip-spec-v1.2.html>.

**Abstract:**

Describes a profile for Tape Libraries as KMIP clients interacting with KMIP servers.

**Status:**

This document was last revised or approved by the OASIS Key Management Interoperability Protocol (KMIP) TC on the above date. The level of approval is also listed above. Check the "Latest version" location noted above for possible later revisions of this document.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "Send A Comment" button on the Technical Committee's web page at <http://www.oasis-open.org/committees/kmip>/<http://www.oasis-open.org/committees/kmip/>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (<http://www.oasis-open.org/committees/kmip/ipr.php>/<http://www.oasis-open.org/committees/kmip/ipr.php>).

**Citation format:**

When referencing this specification the following citation format should be used:

**[kmip-tape-lib-v1.0]**

*KMIP Tape Library Profile Version 1.0*. Edited by Tim Hudson, Stan Feather, and Rod Wideman. 09 January 2014. OASIS Committee Specification Draft 01 / Public Review Draft 01.

<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/csprd01/kmip-tape-lib-profile-v1.0-csprd01.html>. Latest version: <http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.html>/<http://docs.oasis-open.org/kmip/kmip-tape-lib-profile/v1.0/kmip-tape-lib-profile-v1.0.html>.

---

## Notices

Copyright © OASIS Open 2014. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full [Policy](#) may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

# Table of Contents

1	Introduction	65
1.1	Terminology	65
1.2	Normative References	65
1.3	Non-Normative References	76
2	Tape Library Profile	87
2.1	Authentication Suite	87
2.2	Baseline Tape Library	87
2.3	Using Application Specific Information for Key Identifiers	108
2.4	Using Alternative Name for tape media barcode	1140
3	Tape Library Profile Test Cases	1244
3.1	Mandatory Test Cases	1244
3.1.1	TL-M-1-10 - Configuration	1244
3.1.2	TL-M-2-10 - Write with new (created) key	1342
3.1.3	TL-M-3-10 - Read an encrypted tape	1644
3.1.4	TL-M-1-11 - Configuration	2423
3.1.5	TL-M-2-11 - Write with new (created) key	2524
3.1.6	TL-M-3-11 - Read an encrypted tape	2826
3.1.7	TL-M-1-12 - Configuration	3635
3.1.8	TL-M-2-12 - Write with new (created) key	3736
3.1.9	TL-M-3-12 - Read an encrypted tape	4038
4	Conformance	4948
4.1	Conformance Statement	4948
4.2	Permitted Test Case Variations	5048
4.2.1	Variable Items	5048
4.2.2	Variable behavior	5149
Appendix A.	Acknowledgments	5351
Appendix B.	KMIP Specification Cross Reference	5654
Appendix C.	Revision History	6159
1	Introduction	6
1.1	Terminology	6
1.2	Normative References	6
2	Tape Library Profile	7
2.1	Authentication Suite	8
2.2	Baseline Tape Library - Client	8
2.3	Baseline Tape Library - Server	8
2.4	Using Application Specific Information for Key Identifiers	10
2.5	Using Alternative Name for tape media barcode	11
3	Tape Library Profile Test Cases	11
3.1	Mandatory Test Cases KMIP v1.0	12
3.1.1	TL-M-1-10 - Configuration	12
3.1.2	TL-M-2-10 - Write with new (created) key	13
3.1.3	TL-M-3-10 - Read an encrypted tape	16
3.2	Mandatory Test Cases KMIP v1.1	24

3.2.1	TL-M-1-11 - Configuration .....	24
3.2.2	TL-M-2-11 - Write with new (created) key .....	25
3.2.3	TL-M-3-11 - Read an encrypted tape .....	28
3.3	Mandatory Test Cases KMIP v1.2 .....	36
3.3.1	TL-M-1-12 - Configuration .....	36
3.3.2	TL-M-2-12 - Write with new (created) key .....	37
3.3.3	TL-M-3-12 - Read an encrypted tape .....	40
4	Conformance .....	49
4.1	Tape Library Client KMIP v1.0 Conformance .....	49
4.2	Tape Library Client KMIP v1.1 Conformance .....	49
4.3	Tape Library Client KMIP v1.2 Conformance .....	49
4.4	Tape Library Server KMIP v1.0 Conformance .....	49
4.5	Tape Library Server KMIP v1.1 Conformance .....	49
4.6	Tape Library Server KMIP v1.2 Conformance .....	50
4.7	Permitted Test Case Variations .....	50
4.7.1	Variable Items .....	50
4.7.2	Variable behavior .....	51
Appendix A.	Acknowledgments .....	53
Appendix B.	KMIP Specification Cross Reference .....	56
Appendix C.	Revision History .....	61

# 1 Introduction

For normative definition of the elements of KMIP see the [KMIP Specification](#) [KMIP-SPEC] and the [KMIP Profiles](#) [KMIP-PROF].

~~Illustrative guidance for the implementation of KMIP clients and servers is provided in the [KMIP Usage Guide](#) [KMIP-UG].~~

This profile defines the necessary KMIP functionality that a Tape Library operating as a KMIP client SHALL use and a KMIP server conforming to this profile SHALL support in order to interoperate in conformance with this profile.

## 1.1 Terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

Key Associated Data (KAD)	Part of the tape format. May be segmented into authenticated and unauthenticated fields. KAD usage is detailed in the SCSI SSC-3 standard from the T10 organization available as ANSI INCITS 335-2000.
Hexadecimal Numeric Characters	Case-sensitive, printable, single byte ASCII characters representing the numbers 0 through 9 and uppercase alpha A through F. (US-ASCII characters 30h-39h and 41h-46h). Each byte (single 8-bit numeric value) is represented as two hexadecimal numeric characters with the high-nibble represented by the first (left-most) hexadecimal numeric character and the low-nibble represented by the second (right-most) hexadecimal numeric character.
N(a)	The maximum number of bytes in the tape authenticated KAD field. For LTO4, N(a) is 12 bytes. For LTO5, N(a) is 60 bytes. For LTO6, N(a) is 60 bytes.
N(u)	The maximum number of bytes in the tape unauthenticated KAD field. For LTO4, N(u) is 32 bytes. For LTO5, N(u) is 32 bytes. For LTO6, N(u) is 32 bytes.
N(k)	The maximum number of bytes in the tape format KAD fields – i.e. N(a) + N(u). For LTO4, N(k) is 44 bytes. For LTO5, N(k) is 92 bytes. For LTO6, N(k) is 92 bytes.

## 1.2 Normative References

- [RFC2119] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, BCP 14, RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>.

- 18 ~~[RFC2246]~~ — ~~T. Dierks and C. Allen, *The TLS Protocol, Version 1.0*, IETF RFC 2246, Jan~~  
 19 ~~1999, <http://www.ietf.org/rfc/rfc2246.txt>~~
- 20 ~~[KMIP-ENCODE]~~ ~~*KMIP Additional Message Encodings Version 1.0.*~~  
 21 ~~[URL](#)~~  
 22 ~~Candidate OASIS Standard 01. **DD MMM YYYY**~~
- 23 [KMIP-SPEC] One or more of [KMIP-SPEC-1\_0], [KMIP-SPEC-1\_1], [KMIP-SPEC-1\_2]  
 24 [KMIP-SPEC-1\_0] Key Management Interoperability Protocol Specification Version 1.0  
 25 <http://docs.oasis-open.org/kmip/spec/v1.0/os/kmip-spec-1.0-os.doc>  
 26 OASIS Standard, October 2010.
- 27 [KMIP-SPEC-1\_1] *Key Management Interoperability Protocol Specification Version 1.1.*  
 28 <http://docs.oasis-open.org/kmip/spec/v1.1/os/kmip-spec-v1.1-os.doc>  
 29 OASIS Standard. 24 January 2013.
- 30 [KMIP-SPEC-1\_2] *Key Management Interoperability Protocol Specification Version 1.2.*  
 31 ~~[URL](#)~~  
 32 Candidate OASIS Standard 01. **DD MMM YYYY**.
- 33 [KMIP-PROF] One or more of [KMIP-PROF-1\_0], [KMIP-PROF-1\_1], [KMIP-PROF-1\_2]  
 34 [KMIP-PROF-1\_0] *Key Management Interoperability Protocol ~~Usage Guide Profiles~~ Version 1.0.*  
 35 <http://docs.oasis-open.org/kmip/profiles/v1.0/os/kmip-profiles-1.0-os.doc>  
 36 OASIS Standard. 1 October 2010.
- 37 [KMIP-PROF-1\_1] *~~Key~~0Key Management Interoperability Protocol ~~Usage Guide Profiles~~ Version*  
 38 ~~1.1.1.~~  
 39 <http://docs.oasis-open.org/kmip/profiles/v1.1/os/kmip-profiles-v1.1-os.doc>  
 40 OASIS Standard 01. 24 January 2013.
- 41 [KMIP-PROF-1\_2] *~~Key~~0Key Management Interoperability Protocol ~~Usage Guide Profiles~~ Version 1.2.*  
 42 ~~[URL](#)~~  
 43 Candidate OASIS Standard 01. **DD MMM YYYY**.

### 44 **1.3 Non-Normative References**

- 45 ~~[KMIP-UG]~~ — ~~One or more of [KMIP-UG-1\_0], [KMIP-UG-1\_1], [KMIP-UG-1\_2]~~  
 46 ~~[KMIP-UG-1\_0]~~ — ~~*Key Management Interoperability Protocol Usage Guide Version 1.0.*~~  
 47 ~~<http://docs.oasis-open.org/kmip/ug/v1.1/kmip-ug-v1.1-cnd01.doc>~~  
 48 ~~Committee Note Draft, 1 December 2011-~~
- 49 ~~[KMIP-UG-1\_1]~~ — ~~*Key Management Interoperability Protocol Usage Guide Version 1.1.*~~  
 50 ~~<http://docs.oasis-open.org/kmip/ug/v1.1/cn01/kmip-ug-v1.1-cn01.doc>~~  
 51 ~~Committee Note 01, 27 July 2012-~~
- 52 ~~[KMIP-UG-1\_2]~~ — ~~*Key Management Interoperability Protocol Usage Guide Version 1.2.*~~  
 53 ~~[URL](#)~~  
 54 ~~Committee Note Draft, **DD MMM YYYY**-~~
- 55 ~~[KMIP-TC-1\_1]~~ — ~~*Key Management Interoperability Protocol Test Cases Version 1.1.*~~  
 56 ~~[http://docs.oasis-open.org/kmip/testcases/v1.1/cn01/kmip-testcases-v1.1-](http://docs.oasis-open.org/kmip/testcases/v1.1/cn01/kmip-testcases-v1.1-cn01.doc)~~  
 57 ~~[cn01.doc](#), Committee Note 01, 27 July 2012-~~
- 58 ~~[KMIP-TC-1\_2]~~ — ~~*Key Management Interoperability Protocol Test Cases Version 1.2.*~~  
 59 ~~[URL](#), Committee Note Draft, **DD MMM YYYY**-~~
- 60 ~~[KMIP-UC]~~ — ~~*Key Management Interoperability Protocol Use Cases Version 1.0.*~~  
 61 ~~<http://docs.oasis-open.org/kmip/usecases/v1.0/cs01/kmip-usecases-1.0-cs->~~  
 62 ~~[01.doc](#), Committee Specification, 15 June 2010-~~

---

## 2 Tape Library Profile

63

64 The Tape Library Profile specifies the behavior of a tape library operating as a KMIP client interacting with  
65 a KMIP server.

### 2.1 Authentication Suite

66

67 Implementations conformant to this profile SHALL support at least one of the Authentication Suites  
68 defined within [KMIP-PROF]. ~~The establishment of the trust relationship between the KMIP client and the~~  
69 ~~KMIP server is the same as the defined base profiles for the version of the profile supported.~~

### 2.2 Baseline Tape Library - Client

70

71 KMIP clients conformant to this profile under [KMIP-SPEC-1\_0]:

72 1. SHALL conform to the [KMIP-SPEC-1\_0]

73 ~~KMIP clients conformant to this profile under [KMIP-SPEC-1\_1]:~~

74 ~~4.2. SHALL conform to the Baseline Client profile in Clause (section 5.12) of [KMIP-PROF] and [KMIP-~~  
75 ~~SPEC-1\_1]~~

76 ~~KMIP clients conformant to this profile under [KMIP-SPEC-1\_2]:~~

77 3. SHALL conform to the *Baseline Client* (section 5.2) of [KMIP-PROF-1\_2]

78 ~~KMIP clients conformant to this profile~~

79 ~~2.4.~~ SHOULD support *Application Specific Information* [KMIP-SPEC] with Application Data provided  
80 by the client in accordance with section ~~2.4.2-3~~

81 ~~3.5.~~ SHOULD NOT use a *Custom Attribute* [KMIP-SPEC] that duplicates information that is already in  
82 standard *Attributes* [KMIP-SPEC]

83 ~~4.6.~~ MAY use x-Barcode as a *Custom Attribute* [KMIP-SPEC] of type Text String to store the barcode

84 ~~5.7.~~ MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause  
85 within this section 2.2

86 ~~6.8.~~ MAY support extensions outside the scope of this standard (e.g., vendor extensions,  
87 conformance clauses) that do not conflict with any KMIP requirements

88

89 KMIP clients conformant to this profile under [KMIP-SPEC-1\_2]:

90 ~~4.9.~~ SHALL support the following *Attributes* [KMIP-SPEC]

91 a. *Alternative Name* [KMIP-SPEC-1\_2]

92 ~~2.10.~~ SHALL support the following *Message Encoding* [KMIP-SPEC-1\_2]:

93 a. *Alternative Name Type Enumeration* [KMIP-SPEC-1\_2] value:

94 i. Uninterpreted Text String

95 ~~3.11.~~ SHALL store the media barcode information in an *Alternative Name* [KMIP-SPEC-1\_2]  
96 *Attribute* [KMIP-SPEC-1\_2] in accordance with section ~~02-4~~

97

### 2.3 Baseline Tape Library - Server

98

99 KMIP servers conformant to this profile under [KMIP-SPEC-1\_0]:

100 1. SHALL conform to the [KMIP-Baseline-Server-SPEC-1\_0]

101 ~~KMIP servers conformant to this~~ profile ~~in~~ under [KMIP-SPEC-1\_1]:



102 | ~~4-2. SHALL conform to the *Baseline Client Clause* (section 5.12) of [KMIP-PROF] and [KMIP-SPEC-~~  
103 | ~~1-1]~~  
104 | ~~KMIP servers conformant to this profile under [KMIP-SPEC-1 2]:~~  
105 | ~~3. SHALL conform to the *Baseline Client* (section 5.2) of [KMIP-PROF-1 2]~~  
106 | ~~KMIP servers conformant to this profile:~~  
107 | ~~2-4. SHALL support the following *Objects* [KMIP-SPEC]~~  
108 |     ~~a. *Symmetric Key* [KMIP-SPEC]~~  
109 | ~~3-5. SHALL support the following *Attributes* [KMIP-SPEC]:~~  
110 |     ~~a. *Name* [KMIP-SPEC]~~  
111 |     ~~b. *Cryptographic Algorithm* [KMIP-SPEC]~~  
112 |     ~~c. *Custom Attribute* [KMIP SPEC]~~  
113 |     ~~d. *Application Specific Information* [KMIP SPEC]~~  
114 | ~~4-6. SHALL support the following *Client-to-Server Operations* [KMIP-SPEC]:~~  
115 |     ~~a. *Create* [KMIP-SPEC]~~  
116 | ~~5-7. SHALL support the following *Message Contents* [KMIP-SPEC]:~~  
117 |     ~~a. *Batch Order Option* [KMIP-SPEC] value:~~  
118 |         ~~i. True~~  
119 |     ~~b. *Batch Count* [KMIP-SPEC] value:~~  
120 |         ~~i. 1 to 32~~  
121 | ~~6-8. SHALL support the following *Message Encoding* [KMIP-SPEC]:~~  
122 |     ~~a. *Cryptographic Algorithm Enumeration* [KMIP-SPEC] value:~~  
123 |         ~~i. AES~~  
124 |     ~~b. *Object Type Enumeration* [KMIP-SPEC] value:~~  
125 |         ~~i. Symmetric Key~~  
126 |     ~~c. *Key Format Type Enumeration* [KMIP-SPEC] value:~~  
127 |         ~~i. Raw~~  
128 |     ~~d. *Cryptographic Length* [KMIP-SPEC] value :~~  
129 |         ~~i. 256-bit~~  
130 |     ~~e. *Name Type Enumeration* [KMIP-SPEC] value:~~  
131 |         ~~i. Uninterpreted Text String~~  
132 | ~~7-9. SHALL support *Custom Attribute* [KMIP-SPEC] with the following data types and properties:~~  
133 |     ~~a. Text String~~  
134 |     ~~b. Integer~~  
135 |     ~~c. Date Time~~  
136 | ~~8-10. \_\_\_\_\_ SHALL support a minimum length of 256 characters for *Custom Attribute* [KMIP-SPEC]~~  
137 | ~~and *Name* [KMIP-SPEC] values where the attribute type is of variable length~~  
138 | ~~9-11. \_\_\_\_\_ SHALL support a minimum of 30 *Custom Attribute* [KMIP-SPEC] per managed object~~  
139 | ~~10-12. \_\_\_\_\_ SHALL support a minimum of 64 characters in *Custom Attribute* [KMIP-SPEC] names~~  
140 | ~~11. SHALL support the *Tape Library Profile Test Cases*, returning results in accordance with the test~~  
141 | ~~cases~~  
142 | ~~1-13. \_\_\_\_\_ MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other~~  
143 | ~~clause within this section 2.2~~  
144 | ~~12-14. \_\_\_\_\_ MAY support extensions outside the scope of this standard (e.g., vendor extensions,~~  
145 | ~~conformance clauses) that do not conflict with any KMIP requirements~~

146  
147 KMIP servers conformant to this profile under [KMIP-SPEC-1\_2]:

148 | ~~4.15.~~ SHALL support the following *Attributes* [KMIP-SPEC]

149 | a. *Alternative Name* [KMIP-SPEC-1\_2]

150 | ~~2.16.~~ SHALL support the following *Message Encoding* [KMIP-SPEC-1\_2]:

151 | a. *Alternative Name Type Enumeration* [KMIP-SPEC-1\_2] value:

152 | i. Uninterpreted Text String

## 153 | ~~2.32.4~~ Using Application Specific Information for Key Identifiers

154 | This information applies to Tape Libraries that use the *Application Specific Information* [KMIP-SPEC]  
155 | attribute to store key identifiers. KMIP clients are ~~NOT~~not required to use *Application Specific Information*  
156 | [KMIP-SPEC] however KMIP servers are required to support KMIP clients that use *Application Specific*  
157 | *Information* [KMIP-SPEC] and KMIP clients that do not use *Application Specific Information* [KMIP-  
158 | SPEC].

159 | The *Application Specific Information* [KMIP-SPEC] MAY be used to store data that is specific to the  
160 | application (Tape Library) using the object.

161 | The following Application Namespaces SHOULD be used in the Application Namespace field of the  
162 | *Application Specific Information* [KMIP-SPEC]:

- 163 | • LIBRARY-LTO, LIBRARY-LTO4, LIBRARY-LTO5, and LIBRARY-LTO6

164 | For backwards compatibility with deployed Tape Library implementations, servers MAY support  
165 | VENDOR-LIBRARY-LTO as an Application Namespace, where VENDOR is an ASCII string that SHALL  
166 | NOT be further interpreted and SHALL be handled by the server as if the Application Namespace was set  
167 | to LIBRARY-LTO.

168 | *Application Specific Information* [KMIP-SPEC] supports key identifiers being created either on the server  
169 | or on the client (Tape Library), but not both. This profile specifies key identifiers created by the client.

170 | The *Application Specific Information* [KMIP-SPEC] method of key identification relies on the ability to  
171 | uniquely identify a key based only on its Application Data (preferably), or (alternatively) on some  
172 | combination of Application Data and *Custom Attributes* [KMIP-SPEC], which the key creator guarantees  
173 | to be unique within the Application Namespace.

174 | Key identifiers stored in the KMIP server's *Application Specific Information* [KMIP-SPEC] are in ASCII  
175 | format. Key identifiers stored in the KMIP client's tape format KAD fields are numeric format. The  
176 | specific algorithm for converting between text and numeric formats is specified below.

177 | All information contained in the tape format's KAD fields is converted to an ASCII string consisting of  
178 | hexadecimal numeric character pairs as follows:

- 179 | 1. The unauthenticated KAD is converted to text;
- 180 | 2. The authenticated KAD is converted to text and;
- 181 | 3. The converted authenticated KAD text is concatenated to the end of the converted  
182 | unauthenticated KAD text.

183 | If the implementation uses client-created key identifiers, then the client generates a new identifier in  
184 | ASCII format that SHALL be unique within the chosen namespace. The source material for generating the  
185 | string is dependent on client policy. The numeric representation of this identifier SHALL be no larger than  
186 | the N(k) bytes of the KAD for the tape media being used.

187 | For KMIP clients and servers conforming to this profile, *Application Specific Information* [KMIP-SPEC]  
188 | SHALL be created by the Tape Library KMIP client based on the tape format's KAD fields as follows:

- 189 | 1. Define an empty output buffer sufficient to contain a string with a maximum length of 2\*N(k)  
190 | bytes.

- 191 2. Copy the tape format's unauthenticated KAD (if present) to the output buffer, converting each  
192 byte value to exactly two Hexadecimal Numeric Characters. The first byte (i.e., byte 0) of the  
193 output buffer is the first byte of unauthenticated KAD.
- 194 3. Concatenate the tape format's authenticated KAD to the output buffer, converting each byte value  
195 to exactly two Hexadecimal Numeric Characters.

196

197 Note: the contents of the unauthenticated KAD and authenticated KAD fields may be less than the  
198 maximum permitted lengths; the implementation provides the correct length values to use in the algorithm  
199 rather than using fixed maximum length fields.

200 If *Application Specific Information* [KMIP-SPEC] is supported, then it SHALL be used by the client for  
201 locating the object for the purpose of encrypting and decrypting data on tape. The *Application Specific*  
202 *Information* [KMIP-SPEC] value SHALL solely be used for this purpose.

203

204 ~~Refer to [KMIP-UG] for further non-normative details of conversion to and from KAD values.~~

## 205 **2.42.5 Using Alternative Name for tape media barcode**

206 The Tape Library client SHALL assign a text (i.e., human-readable) representation of the media barcode  
207 to the *Alternative Name* [KMIP-SPEC-1\_2] of the object. This SHALL occur on first use of the object for  
208 encryption, which normally is when the library requests the server to create the object.

209

210 The relationship between key identifiers in *Application Specific Information* [KMIP-SPEC] and *Alternative*  
211 *Name* [KMIP-SPEC-1\_2] is as follows:

- 212 a) The values for both are provided by the client
- 213 b) The identifier in *Alternative Name* [KMIP-SPEC-1\_2] (i.e., the barcode) SHALL be used by the  
214 server administrator for finding keys associated with specific tape media (e.g., a server  
215 administrator may want to find the key(s) associated with a missing tape cartridge, where the  
216 barcode of that tape cartridge is known).
- 217 c) The *Alternative Name* [KMIP-SPEC-1\_2] SHALL NOT be used by a client for locating the object  
218 to encrypt or decrypt data, since the value (barcode) is ~~NOT~~not required to be unique and  
219 therefore- does not ensure retrieval of the correct key.

220 ~~Refer to [KMIP-UG-1\_2] for further non-normative details of *Alternative Name* [KMIP-SPEC-1\_2] usage.~~

221

### 3 Tape Library Profile Test Cases

222  
223

~~This section documents the test cases that a client or server conformant to the Tape Library Profile SHALL support.~~

224  
225  
226  
227

~~The test cases define a number of request-response pairs for KMIP operations. Each test case is provided in the XML format specified in [KMIP-ENCODE] intended to be both human-readable and usable by automated tools. The time sequence (starting from 0) for each request-response pair is noted and line numbers are provided for ease of cross-reference for a given test sequence.~~

228  
229

~~Each test case has a unique label (the section name) which includes indication of mandatory (-M-) or optional (-O-) status and the protocol version major and minor numbers as part of the identifier.~~

230  
231

~~The test cases may depend on a specific configuration of a KMIP client and server being configured in a manner consistent with the test case assumptions.~~

232  
233  
234

~~Where possible the flow of unique identifiers between tests, the date-time values, and other dynamic items are indicated using symbolic identifiers – in actual request and response messages these dynamic values will be filled in with valid values.~~

235  
236  
237

~~Note: the values for the returned items and the *Custom Attributes* custom attributes are illustrative. Actual values from a real client system will vary. ~~These test cases demonstrate usage of all of the mandatory types that are required to be supported in a server implementation, as specified in section 4.7.~~~~

238

#### 3.1 Mandatory Test Cases KMIP v1.0

239  
240

~~This section documents the test cases that a client or server conformant to the Tape Library Profile SHALL support for each version of the KMIP protocol (major and minor).~~

241

##### 3.1.1 TL-M-1-10 - Configuration

242  
243  
244

Determine server configuration details including operations supported (only the mandatory operations are listed in the response example), objects supported (only the mandatory objects types are listed in the response example), optional server information, and optional list of application name spaces.

0001	# TIME 0
0002	<RequestMessage>
0003	<RequestHeader>
0004	<ProtocolVersion>
0005	<ProtocolVersionMajor type="Integer" value="1"/>
0006	<ProtocolVersionMinor type="Integer" value="0"/>
0007	</ProtocolVersion>
0008	<BatchCount type="Integer" value="1"/>
0009	</RequestHeader>
0010	<BatchItem>
0011	<Operation type="Enumeration" value="Query"/>
0012	<UniqueBatchItemID type="ByteString" value="01"/>
0013	<RequestPayload>
0014	<QueryFunction type="Enumeration" value="QueryOperations"/>
0015	<QueryFunction type="Enumeration"
0016	value="QueryServerInformation"/>
0017	<QueryFunction type="Enumeration"
0018	value="QueryApplicationNamespaces"/>
0019	</RequestPayload>
0020	</BatchItem>
0021	</RequestMessage>
0022	<ResponseMessage>
0023	<ResponseHeader>

```

0022     <ProtocolVersion>
0023         <ProtocolVersionMajor type="Integer" value="1"/>
0024         <ProtocolVersionMinor type="Integer" value="0"/>
0025     </ProtocolVersion>
0026     <TimeStamp type="DateTime" value="2012-10-05T21:35:17+00:00"/>
0027     <BatchCount type="Integer" value="1"/>
0028 </ResponseHeader>
0029 <BatchItem>
0030     <Operation type="Enumeration" value="Query"/>
0031     <UniqueBatchItemID type="ByteString" value="01"/>
0032     <ResultStatus type="Enumeration" value="Success"/>
0033     <ResponsePayload>
0034         <Operation type="Enumeration" value="Query"/>
0035         <Operation type="Enumeration" value="Locate"/>
0036         <Operation type="Enumeration" value="Destroy"/>
0037         <Operation type="Enumeration" value="Get"/>
0038         <Operation type="Enumeration" value="Create"/>
0039         <Operation type="Enumeration" value="Register"/>
0040         <Operation type="Enumeration" value="GetAttributes"/>
0041         <Operation type="Enumeration" value="GetAttributeList"/>
0042         <Operation type="Enumeration" value="AddAttribute"/>
0043         <Operation type="Enumeration" value="ModifyAttribute"/>
0044         <Operation type="Enumeration" value="DeleteAttribute"/>
0045         <Operation type="Enumeration" value="Activate"/>
0046         <Operation type="Enumeration" value="Revoke"/>
0047         <Operation type="Enumeration" value="Check"/>
0048         <ObjectType type="Enumeration" value="SymmetricKey"/>
0049         <ObjectType type="Enumeration" value="Template"/>
0050         <VendorIdentification type="TextString" value="server-
0051         vendor.com"/>
0051     <ServerInformation>
0052     </ServerInformation>
0053     </ResponsePayload>
0054 </BatchItem>
0055 </ResponseMessage>

```

245

246 **3.1.2 TL-M-2-10 - Write with new (created) key**

247 This case may occur when the Write operation starts with the first block on a tape. The implementation  
248 may choose which Write operations qualify for creation of a new key. Regardless of the initiating  
249 circumstances, the Tape Library requests the server to create a new AES-256 symmetric key with  
250 appropriate identifying information which is unique within the Application Namespace.

251 Additional custom attributes MAY be specified in order to:

- 252 - ensure uniqueness of the key identifier when later Locating the key via ASI
- 253 - provide human-readable information (such as the tape Barcode value)
- 254 - provide information to support client-specific purposes

```

0001 # TIME 0
0001 <RequestMessage>
0002     <RequestHeader>
0003         <ProtocolVersion>
0004             <ProtocolVersionMajor type="Integer" value="1"/>
0005             <ProtocolVersionMinor type="Integer" value="0"/>
0006         </ProtocolVersion>
0007         <BatchOrderOption type="Boolean" value="true"/>
0008         <BatchCount type="Integer" value="2"/>

```

```

0009 </RequestHeader>
0010 <BatchItem>
0011   <Operation type="Enumeration" value="Create"/>
0012   <UniqueBatchItemID type="ByteString" value="01"/>
0013   <RequestPayload>
0014     <ObjectType type="Enumeration" value="SymmetricKey"/>
0015     <TemplateAttribute>
0016       <Attribute>
0017         <AttributeName type="TextString" value="x-ID"/>
0018         <AttributeValue type="TextString" value="TL-M-2-10"/>
0019       </Attribute>
0020       <Attribute>
0021         <AttributeName type="TextString" value="Name"/>
0022         <AttributeValue>
0023           <NameValue type="TextString"
0024 value="AAAAAAAA1A1AA12345678901234567"/>
0025           <NameType type="Enumeration"
0026 value="UninterpretedTextString"/>
0027         </AttributeValue>
0028       </Attribute>
0029         <AttributeName type="TextString" value="Cryptographic
0030 Algorithm"/>
0031         <AttributeValue type="Enumeration" value="AES"/>
0032       </Attribute>
0033         <AttributeName type="TextString" value="Cryptographic
0034 Length"/>
0035         <AttributeValue type="Integer" value="256"/>
0036       </Attribute>
0037         <AttributeName type="TextString" value="Cryptographic
0038 Usage Mask"/>
0039         <AttributeValue type="Integer" value="Decrypt Encrypt"/>
0040       </Attribute>
0041         <AttributeName type="TextString" value="Application
0042 Specific Information"/>
0043         <AttributeValue>
0044           <ApplicationNamespace type="TextString" value="LIBRARY-
0045 LTO"/>
0046           <ApplicationData type="TextString"
0047 value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
0048 000"/>
0049         </AttributeValue>
0050       </Attribute>
0051         <AttributeName type="TextString" value="x-Barcode"/>
0052         <AttributeValue type="TextString" value="XXA012A1"/>
0053       </Attribute>
0054         <AttributeName type="TextString" value="x-
0055 VendorAttribute1"/>
0056         <AttributeValue type="TextString" value="XXA012A1"/>
0057       </Attribute>
0058         <AttributeName type="TextString" value="x-
0059 VendorAttribute2"/>

```





0108	</BatchItem>
0109	</ResponseMessage>

255

### 256 3.1.3 TL-M-3-10 - Read an encrypted tape

257 The Tape Library constructs an identifier string based on the method in 2.3, then requests the server to  
 258 Locate that string via ASI. A Get is then requested based on the key's unique identifier. The Tape Library  
 259 MAY update attributes associated with the Symmetric Key Managed Object. The following test case  
 260 shows extensive use of custom attributes. Custom attributes are not required if the Application Name is  
 261 unique within the Application Namespace. An implementation may also use custom attributes for vendor-  
 262 unique purposes, or to improve usability.

263 The test case destroys the key created in the previous test case to clean up after the test. Tape Library  
 264 implementations ~~MAY~~may elect to not perform this step.

265

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="0"/>
0006     </ProtocolVersion>
0007     <BatchOrderOption type="Boolean" value="true"/>
0008     <BatchCount type="Integer" value="2"/>
0009   </RequestHeader>
0010   <BatchItem>
0011     <Operation type="Enumeration" value="Locate"/>
0012     <UniqueBatchItemID type="ByteString" value="01"/>
0013     <RequestPayload>
0014       <Attribute>
0015         <AttributeName type="TextString" value="Object Type"/>
0016         <AttributeValue type="Enumeration" value="SymmetricKey"/>
0017       </Attribute>
0018       <Attribute>
0019         <AttributeName type="TextString" value="Application Specific
Information"/>
0020         <AttributeValue>
0021           <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0022           <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0023         </AttributeValue>
0024       </Attribute>
0025     </RequestPayload>
0026   </BatchItem>
0027   <BatchItem>
0028     <Operation type="Enumeration" value="Get"/>
0029     <UniqueBatchItemID type="ByteString" value="02"/>
0030     <RequestPayload>
0031   </RequestPayload>
0032   </BatchItem>
0033 </RequestMessage>
0034 <ResponseMessage>
0035   <ResponseHeader>
0036     <ProtocolVersion>

```





```

Information"/>
0090     <AttributeValue>
0091     <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0092     <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0093     </AttributeValue>
0094     </Attribute>
0095     </RequestPayload>
0096 </BatchItem>
0097 <BatchItem>
0098   <Operation type="Enumeration" value="GetAttributeList"/>
0099   <UniqueBatchItemID type="ByteString" value="02"/>
0100   <RequestPayload>
0101   </RequestPayload>
0102 </BatchItem>
0103 </RequestMessage>
0104 <ResponseMessage>
0105   <ResponseHeader>
0106     <ProtocolVersion>
0107       <ProtocolVersionMajor type="Integer" value="1"/>
0108       <ProtocolVersionMinor type="Integer" value="0"/>
0109     </ProtocolVersion>
0110     <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0111     <BatchCount type="Integer" value="2"/>
0112   </ResponseHeader>
0113   <BatchItem>
0114     <Operation type="Enumeration" value="Locate"/>
0115     <UniqueBatchItemID type="ByteString" value="01"/>
0116     <ResultStatus type="Enumeration" value="Success"/>
0117     <ResponsePayload>
0118       <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER 0"/>
0119     </ResponsePayload>
0120   </BatchItem>
0121   <BatchItem>
0122     <Operation type="Enumeration" value="GetAttributeList"/>
0123     <UniqueBatchItemID type="ByteString" value="02"/>
0124     <ResultStatus type="Enumeration" value="Success"/>
0125     <ResponsePayload>
0126       <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0127       <AttributeName type="TextString" value="x-ID"/>
0128       <AttributeName type="TextString" value="x-Barcode"/>
0129       <AttributeName type="TextString" value="x-VendorAttribute1"/>
0130       <AttributeName type="TextString" value="x-VendorAttribute2"/>
0131       <AttributeName type="TextString" value="x-VendorAttribute3"/>
0132       <AttributeName type="TextString" value="Unique Identifier"/>
0133       <AttributeName type="TextString" value="Object Type"/>
0134       <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0135       <AttributeName type="TextString" value="Cryptographic
Length"/>
0136       <AttributeName type="TextString" value="Application Specific
Information"/>
0137       <AttributeName type="TextString" value="Cryptographic Usage
Mask"/>

```

0138	<AttributeName type="TextString" value="Digest"/>
0139	<AttributeName type="TextString" value="Initial Date"/>
0140	<AttributeName type="TextString" value="Last Change Date"/>
0141	<AttributeName type="TextString" value="Lease Time"/>
0142	<AttributeName type="TextString" value="Name"/>
0143	<AttributeName type="TextString" value="State"/>
0144	</ResponsePayload>
0145	</BatchItem>
0146	</ResponseMessage>
	# TIME 2
0147	<RequestMessage>
0148	<RequestHeader>
0149	<ProtocolVersion>
0150	<ProtocolVersionMajor type="Integer" value="1"/>
0151	<ProtocolVersionMinor type="Integer" value="0"/>
0152	</ProtocolVersion>
0153	<BatchOrderOption type="Boolean" value="true"/>
0154	<BatchCount type="Integer" value="2"/>
0155	</RequestHeader>
0156	<BatchItem>
0157	<Operation type="Enumeration" value="Locate"/>
0158	<UniqueBatchItemID type="ByteString" value="01"/>
0159	<RequestPayload>
0160	<Attribute>
0161	<AttributeName type="TextString" value="Object Type"/>
0162	<AttributeValue type="Enumeration" value="SymmetricKey"/>
0163	</Attribute>
0164	<Attribute>
0165	<AttributeName type="TextString" value="Application Specific Information"/>
0166	<AttributeValue>
0167	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0168	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0169	</AttributeValue>
0170	</Attribute>
0171	</RequestPayload>
0172	</BatchItem>
0173	<BatchItem>
0174	<Operation type="Enumeration" value="GetAttributes"/>
0175	<UniqueBatchItemID type="ByteString" value="02"/>
0176	<RequestPayload>
0177	</RequestPayload>
0178	</BatchItem>
0179	</RequestMessage>
0180	<ResponseMessage>
0181	<ResponseHeader>
0182	<ProtocolVersion>
0183	<ProtocolVersionMajor type="Integer" value="1"/>
0184	<ProtocolVersionMinor type="Integer" value="0"/>
0185	</ProtocolVersion>
0186	<TimeStamp type="DateTime" value="2012-10-05T22:00:32+00:00"/>
0187	<BatchCount type="Integer" value="2"/>
0188	</ResponseHeader>
0189	<BatchItem>
0190	<Operation type="Enumeration" value="Locate"/>

```

0191     <UniqueBatchItemID type="ByteString" value="01"/>
0192     <ResultStatus type="Enumeration" value="Success"/>
0193     <ResponsePayload>
0194         <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0195     </ResponsePayload>
0196 </BatchItem>
0197 </BatchItem>
0198     <Operation type="Enumeration" value="GetAttributes"/>
0199     <UniqueBatchItemID type="ByteString" value="02"/>
0200     <ResultStatus type="Enumeration" value="Success"/>
0201     <ResponsePayload>
0202         <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0203         <Attribute>
0204             <AttributeName type="TextString" value="x-ID"/>
0205             <AttributeValue type="TextString" value="TL-M-2-10"/>
0206         </Attribute>
0207         <Attribute>
0208             <AttributeName type="TextString" value="x-Barcode"/>
0209             <AttributeValue type="TextString" value="XXA012A1"/>
0210         </Attribute>
0211         <Attribute>
0212             <AttributeName type="TextString" value="x-
VendorAttribute1"/>
0213             <AttributeValue type="TextString" value="XXA012A1"/>
0214         </Attribute>
0215         <Attribute>
0216             <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0217             <AttributeValue type="Integer" value="0"/>
0218         </Attribute>
0219         <Attribute>
0220             <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0221             <AttributeValue type="DateTime" value="2012-10-
05T22:08:19+00:00"/>
0222         </Attribute>
0223         <Attribute>
0224             <AttributeName type="TextString" value="Unique Identifier"/>
0225             <AttributeValue type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0226         </Attribute>
0227         <Attribute>
0228             <AttributeName type="TextString" value="Object Type"/>
0229             <AttributeValue type="Enumeration" value="SymmetricKey"/>
0230         </Attribute>
0231         <Attribute>
0232             <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0233             <AttributeValue type="Enumeration" value="AES"/>
0234         </Attribute>
0235         <Attribute>
0236             <AttributeName type="TextString" value="Cryptographic
Length"/>
0237             <AttributeValue type="Integer" value="256"/>
0238         </Attribute>
0239     </Attribute>

```

0240	<AttributeName type="TextString" value="Application Specific Information"/>
0241	<AttributeValue>
0242	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0243	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0244	</AttributeValue>
0245	</Attribute>
0246	<Attribute>
0247	<AttributeName type="TextString" value="Cryptographic Usage Mask"/>
0248	<AttributeValue type="Integer" value="Decrypt Encrypt"/>
0249	</Attribute>
0250	<Attribute>
0251	<AttributeName type="TextString" value="Digest"/>
0252	<AttributeValue>
0253	<HashingAlgorithm type="Enumeration" value="SHA_256"/>
0254	<DigestValue type="ByteString" value="0bd3d7ada745e4eb34cc9d26cc84d4852a15906b2f77f80d01fc31c28108c49b"/>
0255	</AttributeValue>
0256	</Attribute>
0257	<Attribute>
0258	<AttributeName type="TextString" value="Initial Date"/>
0259	<AttributeValue type="DateTime" value="2012-10-05T21:41:46+00:00"/>
0260	</Attribute>
0261	<Attribute>
0262	<AttributeName type="TextString" value="Last Change Date"/>
0263	<AttributeValue type="DateTime" value="2012-10-05T21:41:46+00:00"/>
0264	</Attribute>
0265	<Attribute>
0266	<AttributeName type="TextString" value="Lease Time"/>
0267	<AttributeValue type="Interval" value="3600"/>
0268	</Attribute>
0269	<Attribute>
0270	<AttributeName type="TextString" value="Name"/>
0271	<AttributeValue>
0272	<NameValue type="TextString" value="AAAAAAA1A1AA12345678901234567"/>
0273	<NameType type="Enumeration" value="UninterpretedTextString"/>
0274	</AttributeValue>
0275	</Attribute>
0276	<Attribute>
0277	<AttributeName type="TextString" value="State"/>
0278	<AttributeValue type="Enumeration" value="PreActive"/>
0279	</Attribute>
0280	</ResponsePayload>
0281	</BatchItem>
0282	</ResponseMessage>
	# TIME 3
0283	<RequestMessage>
0284	<RequestHeader>
0285	<ProtocolVersion>

0286	<ProtocolVersionMajor type="Integer" value="1"/>
0287	<ProtocolVersionMinor type="Integer" value="0"/>
0288	</ProtocolVersion>
0289	<BatchOrderOption type="Boolean" value="true"/>
0290	<BatchCount type="Integer" value="3"/>
0291	</RequestHeader>
0292	<BatchItem>
0293	<Operation type="Enumeration" value="Locate"/>
0294	<UniqueBatchItemID type="ByteString" value="01"/>
0295	<RequestPayload>
0296	<Attribute>
0297	<AttributeName type="TextString" value="Object Type"/>
0298	<AttributeValue type="Enumeration" value="SymmetricKey"/>
0299	</Attribute>
0300	<Attribute>
0301	<AttributeName type="TextString" value="Application Specific Information"/>
0302	<AttributeValue>
0303	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0304	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0305	</AttributeValue>
0306	</Attribute>
0307	</RequestPayload>
0308	</BatchItem>
0309	<BatchItem>
0310	<Operation type="Enumeration" value="ModifyAttribute"/>
0311	<UniqueBatchItemID type="ByteString" value="02"/>
0312	<RequestPayload>
0313	<Attribute>
0314	<AttributeName type="TextString" value="x-VendorAttribute2"/>
0315	<AttributeValue type="DateTime" value="2012-10-05T22:27:06+00:00"/>
0316	</Attribute>
0317	</RequestPayload>
0318	</BatchItem>
0319	<BatchItem>
0320	<Operation type="Enumeration" value="ModifyAttribute"/>
0321	<UniqueBatchItemID type="ByteString" value="03"/>
0322	<RequestPayload>
0323	<Attribute>
0324	<AttributeName type="TextString" value="x-VendorAttribute3"/>
0325	<AttributeValue type="Integer" value="1"/>
0326	</Attribute>
0327	</RequestPayload>
0328	</BatchItem>
0329	</RequestMessage>
0330	<ResponseMessage>
0331	<ResponseHeader>
0332	<ProtocolVersion>
0333	<ProtocolVersionMajor type="Integer" value="1"/>
0334	<ProtocolVersionMinor type="Integer" value="0"/>
0335	</ProtocolVersion>
0336	<TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>

0337	<BatchCount type="Integer" value="3"/>
0338	</ResponseHeader>
0339	<BatchItem>
0340	<Operation type="Enumeration" value="Locate"/>
0341	<UniqueBatchItemID type="ByteString" value="01"/>
0342	<ResultStatus type="Enumeration" value="Success"/>
0343	<ResponsePayload>
0344	<UniqueIdentifier type="TextString"
	value="\$UNIQUE_IDENTIFIER_0"/>
0345	</ResponsePayload>
0346	</BatchItem>
0347	<BatchItem>
0348	<Operation type="Enumeration" value="ModifyAttribute"/>
0349	<UniqueBatchItemID type="ByteString" value="02"/>
0350	<ResultStatus type="Enumeration" value="Success"/>
0351	<ResponsePayload>
0352	<UniqueIdentifier type="TextString"
	value="\$UNIQUE_IDENTIFIER_0"/>
0353	<Attribute>
0354	<AttributeName type="TextString" value="x-
	VendorAttribute2"/>
0355	<AttributeValue type="DateTime" value="2012-10-
	05T22:27:06+00:00"/>
0356	</Attribute>
0357	</ResponsePayload>
0358	</BatchItem>
0359	<BatchItem>
0360	<Operation type="Enumeration" value="ModifyAttribute"/>
0361	<UniqueBatchItemID type="ByteString" value="03"/>
0362	<ResultStatus type="Enumeration" value="Success"/>
0363	<ResponsePayload>
0364	<UniqueIdentifier type="TextString"
	value="\$UNIQUE_IDENTIFIER_0"/>
0365	<Attribute>
0366	<AttributeName type="TextString" value="x-
	VendorAttribute3"/>
0367	<AttributeValue type="Integer" value="1"/>
0368	</Attribute>
0369	</ResponsePayload>
0370	</BatchItem>
0371	</ResponseMessage>
	<i># TIME 4</i>
0372	<RequestMessage>
0373	<RequestHeader>
0374	<ProtocolVersion>
0375	<ProtocolVersionMajor type="Integer" value="1"/>
0376	<ProtocolVersionMinor type="Integer" value="0"/>
0377	</ProtocolVersion>
0378	<BatchCount type="Integer" value="1"/>
0379	</RequestHeader>
0380	<BatchItem>
0381	<Operation type="Enumeration" value="Destroy"/>
0382	<RequestPayload>
0383	<UniqueIdentifier type="TextString"
	value="\$UNIQUE_IDENTIFIER_0"/>
0384	</RequestPayload>
0385	</BatchItem>
0386	</RequestMessage>

```

0387 <ResponseMessage>
0388   <ResponseHeader>
0389     <ProtocolVersion>
0390       <ProtocolVersionMajor type="Integer" value="1"/>
0391       <ProtocolVersionMinor type="Integer" value="0"/>
0392     </ProtocolVersion>
0393     <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0394     <BatchCount type="Integer" value="1"/>
0395   </ResponseHeader>
0396   <BatchItem>
0397     <Operation type="Enumeration" value="Destroy"/>
0398     <ResultStatus type="Enumeration" value="Success"/>
0399     <ResponsePayload>
0400       <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0401     </ResponsePayload>
0402   </BatchItem>
0403 </ResponseMessage>

```

266

## 267 **3.2 Mandatory Test Cases KMIP v1.1**

### 268 **3.1.43.2.1 TL-M-1-11 - Configuration**

269 Determine server configuration details including operations supported (only the mandatory operations are  
270 listed in the response example), objects supported (only the mandatory objects types are listed in the  
271 response example), optional server information, and optional list of application name spaces.

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="1"/>
0006     </ProtocolVersion>
0007     <BatchCount type="Integer" value="1"/>
0008   </RequestHeader>
0009   <BatchItem>
0010     <Operation type="Enumeration" value="Query"/>
0011     <UniqueBatchItemID type="ByteString" value="01"/>
0012     <RequestPayload>
0013       <QueryFunction type="Enumeration" value="QueryOperations"/>
0014       <QueryFunction type="Enumeration" value="QueryObjects"/>
0015       <QueryFunction type="Enumeration"
value="QueryServerInformation"/>
0016       <QueryFunction type="Enumeration"
value="QueryApplicationNamespaces"/>
0017     </RequestPayload>
0018   </BatchItem>
0019 </RequestMessage>
0020 <ResponseMessage>
0021   <ResponseHeader>
0022     <ProtocolVersion>
0023       <ProtocolVersionMajor type="Integer" value="1"/>
0024       <ProtocolVersionMinor type="Integer" value="1"/>
0025     </ProtocolVersion>
0026     <TimeStamp type="DateTime" value="2012-10-05T21:35:17+00:00"/>
0027     <BatchCount type="Integer" value="1"/>

```



```

0028 </ResponseHeader>
0029 <BatchItem>
0030   <Operation type="Enumeration" value="Query"/>
0031   <UniqueBatchItemID type="ByteString" value="01"/>
0032   <ResultStatus type="Enumeration" value="Success"/>
0033   <ResponsePayload>
0034     <Operation type="Enumeration" value="Query"/>
0035     <Operation type="Enumeration" value="Locate"/>
0036     <Operation type="Enumeration" value="Destroy"/>
0037     <Operation type="Enumeration" value="Get"/>
0038     <Operation type="Enumeration" value="Create"/>
0039     <Operation type="Enumeration" value="Register"/>
0040     <Operation type="Enumeration" value="GetAttributes"/>
0041     <Operation type="Enumeration" value="GetAttributeList"/>
0042     <Operation type="Enumeration" value="AddAttribute"/>
0043     <Operation type="Enumeration" value="ModifyAttribute"/>
0044     <Operation type="Enumeration" value="DeleteAttribute"/>
0045     <Operation type="Enumeration" value="Activate"/>
0046     <Operation type="Enumeration" value="Revoke"/>
0047     <Operation type="Enumeration" value="Check"/>
0048     <ObjectType type="Enumeration" value="SymmetricKey"/>
0049     <ObjectType type="Enumeration" value="Template"/>
0050     <VendorIdentification type="TextString" value="server-
vendor.com"/>
0051     <ServerInformation>
0052     </ServerInformation>
0053   </ResponsePayload>
0054 </BatchItem>
0055 </ResponseMessage>

```

272

### 273 3.1.53.2.2 TL-M-2-11 - Write with new (created) key

274 This case may occur when the Write operation starts with the first block on a tape. The implementation  
275 may choose which Write operations qualify for creation of a new key. Regardless of the initiating  
276 circumstances, the Tape Library requests the server to create a new AES-256 symmetric key with  
277 appropriate identifying information which is unique within the Application Namespace.

278 Additional custom attributes MAY be specified in order to:

- 279 - ensure uniqueness of the key identifier when later Locating the key via ASI
- 280 - provide human-readable information (such as the tape Barcode value)
- 281 - provide information to support client-specific purposes

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="1"/>
0006     </ProtocolVersion>
0007     <BatchOrderOption type="Boolean" value="true"/>
0008     <BatchCount type="Integer" value="2"/>
0009   </RequestHeader>
0010   <BatchItem>
0011     <Operation type="Enumeration" value="Create"/>
0012     <UniqueBatchItemID type="ByteString" value="01"/>
0013     <RequestPayload>
0014       <ObjectType type="Enumeration" value="SymmetricKey"/>

```

```

0015     <TemplateAttribute>
0016     <Attribute>
0017         <AttributeName type="TextString" value="x-ID"/>
0018         <AttributeValue type="TextString" value="TL-M-2-11"/>
0019     </Attribute>
0020     <Attribute>
0021         <AttributeName type="TextString" value="Name"/>
0022         <AttributeValue>
0023             <NameValue type="TextString"
value="AAAAAAA1A1AA12345678901234567"/>
0024             <NameType type="Enumeration"
value="UninterpretedTextString"/>
0025         </AttributeValue>
0026     </Attribute>
0027     <Attribute>
0028         <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0029         <AttributeValue type="Enumeration" value="AES"/>
0030     </Attribute>
0031     <Attribute>
0032         <AttributeName type="TextString" value="Cryptographic
Length"/>
0033         <AttributeValue type="Integer" value="256"/>
0034     </Attribute>
0035     <Attribute>
0036         <AttributeName type="TextString" value="Cryptographic
Usage Mask"/>
0037         <AttributeValue type="Integer" value="Decrypt Encrypt"/>
0038     </Attribute>
0039     <Attribute>
0040         <AttributeName type="TextString" value="Application
Specific Information"/>
0041         <AttributeValue>
0042             <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0043             <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0044         </AttributeValue>
0045     </Attribute>
0046     <Attribute>
0047         <AttributeName type="TextString" value="x-Barcode"/>
0048         <AttributeValue type="TextString" value="XXA012A1"/>
0049     </Attribute>
0050     <Attribute>
0051         <AttributeName type="TextString" value="x-
VendorAttribute1"/>
0052         <AttributeValue type="TextString" value="XXA012A1"/>
0053     </Attribute>
0054     <Attribute>
0055         <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0056         <AttributeValue type="Integer" value="0"/>
0057     </Attribute>
0058     <Attribute>
0059         <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0060         <AttributeValue type="DateTime" value="2012-10-

```



283 **3.1.63.2.3 TL-M-3-11 - Read an encrypted tape**

284 The Tape Library constructs an identifier string based on the method in 2.3, then requests the server to  
 285 Locate that string via ASI. A Get is then requested based on the key's unique identifier. The Tape Library  
 286 MAY update attributes associated with the Symmetric Key Managed Object. The following test case  
 287 shows extensive use of custom attributes. Custom attributes are not required if the Application Name is  
 288 unique within the Application Namespace. An implementation may also use custom attributes for vendor-  
 289 unique purposes, or to improve usability.

290 The test case destroys the key created in the previous test case to clean up after the test. Tape Library  
 291 implementations MAY may elect to not perform this step.

292

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="1"/>
0006     </ProtocolVersion>
0007     <BatchOrderOption type="Boolean" value="true"/>
0008     <BatchCount type="Integer" value="2"/>
0009   </RequestHeader>
0010   <BatchItem>
0011     <Operation type="Enumeration" value="Locate"/>
0012     <UniqueBatchItemID type="ByteString" value="01"/>
0013     <RequestPayload>
0014       <Attribute>
0015         <AttributeName type="TextString" value="Object Type"/>
0016         <AttributeValue type="Enumeration" value="SymmetricKey"/>
0017       </Attribute>
0018       <Attribute>
0019         <AttributeName type="TextString" value="Application Specific
Information"/>
0020         <AttributeValue>
0021           <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0022           <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0023         </AttributeValue>
0024       </Attribute>
0025     </RequestPayload>
0026   </BatchItem>
0027   <BatchItem>
0028     <Operation type="Enumeration" value="Get"/>
0029     <UniqueBatchItemID type="ByteString" value="02"/>
0030     <RequestPayload>
0031     </RequestPayload>
0032   </BatchItem>
0033 </RequestMessage>
0034 <ResponseMessage>
0035   <ResponseHeader>
0036     <ProtocolVersion>
0037       <ProtocolVersionMajor type="Integer" value="1"/>
0038       <ProtocolVersionMinor type="Integer" value="1"/>
0039     </ProtocolVersion>
0040     <TimeStamp type="DateTime" value="2012-10-05T22:00:32+00:00"/>
0041     <BatchCount type="Integer" value="2"/>

```



```

0093     value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
0094     000"/>
0095         </AttributeValue>
0096     </Attribute>
0097 </RequestPayload>
0098 </BatchItem>
0099 <BatchItem>
0100     <Operation type="Enumeration" value="GetAttributeList"/>
0101     <UniqueBatchItemID type="ByteString" value="02"/>
0102     <RequestPayload>
0103     </RequestPayload>
0104 </BatchItem>
0105 </RequestMessage>
0106 <ResponseMessage>
0107     <ResponseHeader>
0108         <ProtocolVersion>
0109             <ProtocolVersionMajor type="Integer" value="1"/>
0110             <ProtocolVersionMinor type="Integer" value="1"/>
0111         </ProtocolVersion>
0112         <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0113         <BatchCount type="Integer" value="2"/>
0114     </ResponseHeader>
0115     <BatchItem>
0116         <Operation type="Enumeration" value="Locate"/>
0117         <UniqueBatchItemID type="ByteString" value="01"/>
0118         <ResponseStatus type="Enumeration" value="Success"/>
0119         <ResponsePayload>
0120             <UniqueIdentifier type="TextString"
0121             value="$UNIQUE_IDENTIFIER 0"/>
0122         </ResponsePayload>
0123     </BatchItem>
0124     <BatchItem>
0125         <Operation type="Enumeration" value="GetAttributeList"/>
0126         <UniqueBatchItemID type="ByteString" value="02"/>
0127         <ResponseStatus type="Enumeration" value="Success"/>
0128         <ResponsePayload>
0129             <UniqueIdentifier type="TextString"
0130             value="$UNIQUE_IDENTIFIER_0"/>
0131             <AttributeName type="TextString" value="x-ID"/>
0132             <AttributeName type="TextString" value="x-Barcode"/>
0133             <AttributeName type="TextString" value="x-VendorAttribute1"/>
0134             <AttributeName type="TextString" value="x-VendorAttribute2"/>
0135             <AttributeName type="TextString" value="x-VendorAttribute3"/>
0136             <AttributeName type="TextString" value="Unique Identifier"/>
0137             <AttributeName type="TextString" value="Object Type"/>
0138             <AttributeName type="TextString" value="Cryptographic
0139             Algorithm"/>
0140             <AttributeName type="TextString" value="Cryptographic
0141             Length"/>
0142             <AttributeName type="TextString" value="Application Specific
0143             Information"/>
0144             <AttributeName type="TextString" value="Cryptographic Usage
0145             Mask"/>
0146             <AttributeName type="TextString" value="Digest"/>
0147             <AttributeName type="TextString" value="Initial Date"/>
0148             <AttributeName type="TextString" value="Last Change Date"/>
0149             <AttributeName type="TextString" value="Lease Time"/>
0150             <AttributeName type="TextString" value="Name"/>

```

0143	<AttributeName type="TextString" value="State"/>
0144	</ResponsePayload>
0145	</BatchItem>
0146	</ResponseMessage>
	# TIME 2
0147	<RequestMessage>
0148	<RequestHeader>
0149	<ProtocolVersion>
0150	<ProtocolVersionMajor type="Integer" value="1"/>
0151	<ProtocolVersionMinor type="Integer" value="1"/>
0152	</ProtocolVersion>
0153	<BatchOrderOption type="Boolean" value="true"/>
0154	<BatchCount type="Integer" value="2"/>
0155	</RequestHeader>
0156	<BatchItem>
0157	<Operation type="Enumeration" value="Locate"/>
0158	<UniqueBatchItemID type="ByteString" value="01"/>
0159	<RequestPayload>
0160	<Attribute>
0161	<AttributeName type="TextString" value="Object Type"/>
0162	<AttributeValue type="Enumeration" value="SymmetricKey"/>
0163	</Attribute>
0164	<Attribute>
0165	<AttributeName type="TextString" value="Application Specific Information"/>
0166	<AttributeValue>
0167	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0168	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0169	</AttributeValue>
0170	</Attribute>
0171	</RequestPayload>
0172	</BatchItem>
0173	<BatchItem>
0174	<Operation type="Enumeration" value="GetAttributes"/>
0175	<UniqueBatchItemID type="ByteString" value="02"/>
0176	<RequestPayload>
0177	</RequestPayload>
0178	</BatchItem>
0179	</RequestMessage>
0180	<ResponseMessage>
0181	<ResponseHeader>
0182	<ProtocolVersion>
0183	<ProtocolVersionMajor type="Integer" value="1"/>
0184	<ProtocolVersionMinor type="Integer" value="1"/>
0185	</ProtocolVersion>
0186	<TimeStamp type="DateTime" value="2012-10-05T22:00:32+00:00"/>
0187	<BatchCount type="Integer" value="2"/>
0188	</ResponseHeader>
0189	<BatchItem>
0190	<Operation type="Enumeration" value="Locate"/>
0191	<UniqueBatchItemID type="ByteString" value="01"/>
0192	<ResultStatus type="Enumeration" value="Success"/>
0193	<ResponsePayload>
0194	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER 0"/>

```

0195     </ResponsePayload>
0196 </BatchItem>
0197 <BatchItem>
0198   <Operation type="Enumeration" value="GetAttributes"/>
0199   <UniqueBatchItemID type="ByteString" value="02"/>
0200   <ResultStatus type="Enumeration" value="Success"/>
0201   <ResponsePayload>
0202     <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0203     <Attribute>
0204       <AttributeName type="TextString" value="x-ID"/>
0205       <AttributeValue type="TextString" value="TL-M-2-11"/>
0206     </Attribute>
0207     <Attribute>
0208       <AttributeName type="TextString" value="x-
VendorAttribute1"/>
0209       <AttributeValue type="TextString" value="XXA012A1"/>
0210     </Attribute>
0211     <Attribute>
0212       <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0213       <AttributeValue type="Integer" value="0"/>
0214     </Attribute>
0215     <Attribute>
0216       <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0217       <AttributeValue type="DateTime" value="2012-10-
05T22:08:19+00:00"/>
0218     </Attribute>
0219     <Attribute>
0220       <AttributeName type="TextString" value="Unique Identifier"/>
0221       <AttributeValue type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0222     </Attribute>
0223     <Attribute>
0224       <AttributeName type="TextString" value="Object Type"/>
0225       <AttributeValue type="Enumeration" value="SymmetricKey"/>
0226     </Attribute>
0227     <Attribute>
0228       <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0229       <AttributeValue type="Enumeration" value="AES"/>
0230     </Attribute>
0231     <Attribute>
0232       <AttributeName type="TextString" value="Cryptographic
Length"/>
0233       <AttributeValue type="Integer" value="256"/>
0234     </Attribute>
0235     <Attribute>
0236       <AttributeName type="TextString" value="Application Specific
Information"/>
0237       <AttributeValue>
0238         <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0239         <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF123400
00"/>
0240       </AttributeValue>

```



0241	</Attribute>
0242	<Attribute>
0243	<AttributeName type="TextString" value="Cryptographic Usage Mask"/>
0244	<AttributeValue type="Integer" value="Decrypt Encrypt"/>
0245	</Attribute>
0246	<Attribute>
0247	<AttributeName type="TextString" value="Digest"/>
0248	<AttributeValue>
0249	<HashingAlgorithm type="Enumeration" value="SHA_256"/>
0250	<DigestValue type="ByteString" value="0bd3d7ada745e4eb34cc9d26cc84d4852a15906b2f77f80d01fc31c28108c49b"/>
0251	<KeyFormatType type="Enumeration" value="Raw"/>
0252	</AttributeValue>
0253	</Attribute>
0254	<Attribute>
0255	<AttributeName type="TextString" value="Fresh"/>
0256	<AttributeValue type="Boolean" value="false"/>
0257	</Attribute>
0258	<Attribute>
0259	<AttributeName type="TextString" value="Initial Date"/>
0260	<AttributeValue type="DateTime" value="2012-10-05T21:41:46+00:00"/>
0261	</Attribute>
0262	<Attribute>
0263	<AttributeName type="TextString" value="Last Change Date"/>
0264	<AttributeValue type="DateTime" value="2012-10-05T21:41:46+00:00"/>
0265	</Attribute>
0266	<Attribute>
0267	<AttributeName type="TextString" value="Lease Time"/>
0268	<AttributeValue type="Interval" value="3600"/>
0269	</Attribute>
0270	<Attribute>
0271	<AttributeName type="TextString" value="Name"/>
0272	<AttributeValue>
0273	<NameValue type="TextString" value="AAAAAAA1A1AA12345678901234567"/>
0274	<NameType type="Enumeration" value="UninterpretedTextString"/>
0275	</AttributeValue>
0276	</Attribute>
0277	<Attribute>
0278	<AttributeName type="TextString" value="State"/>
0279	<AttributeValue type="Enumeration" value="PreActive"/>
0280	</Attribute>
0281	</ResponsePayload>
0282	</BatchItem>
0283	</ResponseMessage>
0284	# TIME 3 <RequestMessage>
0285	<RequestHeader>
0286	<ProtocolVersion>
0287	<ProtocolVersionMajor type="Integer" value="1"/>
0288	<ProtocolVersionMinor type="Integer" value="1"/>
0289	</ProtocolVersion>
0290	<BatchOrderOption type="Boolean" value="true"/>

0291	<BatchCount type="Integer" value="3"/>
0292	</RequestHeader>
0293	<BatchItem>
0294	<Operation type="Enumeration" value="Locate"/>
0295	<UniqueBatchItemID type="ByteString" value="01"/>
0296	<RequestPayload>
0297	<Attribute>
0298	<AttributeName type="TextString" value="Object Type"/>
0299	<AttributeValue type="Enumeration" value="SymmetricKey"/>
0300	</Attribute>
0301	<Attribute>
0302	<AttributeName type="TextString" value="Application Specific Information"/>
0303	<AttributeValue>
0304	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0305	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0306	</AttributeValue>
0307	</Attribute>
0308	</RequestPayload>
0309	</BatchItem>
0310	<BatchItem>
0311	<Operation type="Enumeration" value="ModifyAttribute"/>
0312	<UniqueBatchItemID type="ByteString" value="02"/>
0313	<RequestPayload>
0314	<Attribute>
0315	<AttributeName type="TextString" value="x-VendorAttribute2"/>
0316	<AttributeValue type="DateTime" value="2012-10-05T22:27:06+00:00"/>
0317	</Attribute>
0318	</RequestPayload>
0319	</BatchItem>
0320	<BatchItem>
0321	<Operation type="Enumeration" value="ModifyAttribute"/>
0322	<UniqueBatchItemID type="ByteString" value="03"/>
0323	<RequestPayload>
0324	<Attribute>
0325	<AttributeName type="TextString" value="x-VendorAttribute3"/>
0326	<AttributeValue type="Integer" value="1"/>
0327	</Attribute>
0328	</RequestPayload>
0329	</BatchItem>
0330	</RequestMessage>
0331	<ResponseMessage>
0332	<ResponseHeader>
0333	<ProtocolVersion>
0334	<ProtocolVersionMajor type="Integer" value="1"/>
0335	<ProtocolVersionMinor type="Integer" value="1"/>
0336	</ProtocolVersion>
0337	<TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0338	<BatchCount type="Integer" value="3"/>
0339	</ResponseHeader>
0340	<BatchItem>
0341	<Operation type="Enumeration" value="Locate"/>

0342	<UniqueBatchItemID type="ByteString" value="01"/>
0343	<ResultStatus type="Enumeration" value="Success"/>
0344	<ResponsePayload>
0345	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0346	</ResponsePayload>
0347	</BatchItem>
0348	<BatchItem>
0349	<Operation type="Enumeration" value="ModifyAttribute"/>
0350	<UniqueBatchItemID type="ByteString" value="02"/>
0351	<ResultStatus type="Enumeration" value="Success"/>
0352	<ResponsePayload>
0353	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0354	<Attribute>
0355	<AttributeName type="TextString" value="x- VendorAttribute2"/>
0356	<AttributeValue type="DateTime" value="2012-10- 05T22:27:06+00:00"/>
0357	</Attribute>
0358	</ResponsePayload>
0359	</BatchItem>
0360	<BatchItem>
0361	<Operation type="Enumeration" value="ModifyAttribute"/>
0362	<UniqueBatchItemID type="ByteString" value="03"/>
0363	<ResultStatus type="Enumeration" value="Success"/>
0364	<ResponsePayload>
0365	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0366	<Attribute>
0367	<AttributeName type="TextString" value="x- VendorAttribute3"/>
0368	<AttributeValue type="Integer" value="1"/>
0369	</Attribute>
0370	</ResponsePayload>
0371	</BatchItem>
0372	</ResponseMessage>
	<i># TIME 4</i>
0373	<RequestMessage>
0374	<RequestHeader>
0375	<ProtocolVersion>
0376	<ProtocolVersionMajor type="Integer" value="1"/>
0377	<ProtocolVersionMinor type="Integer" value="1"/>
0378	</ProtocolVersion>
0379	<BatchCount type="Integer" value="1"/>
0380	</RequestHeader>
0381	<BatchItem>
0382	<Operation type="Enumeration" value="Destroy"/>
0383	<RequestPayload>
0384	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0385	</RequestPayload>
0386	</BatchItem>
0387	</RequestMessage>
0388	<ResponseMessage>
0389	<ResponseHeader>
0390	<ProtocolVersion>
0391	<ProtocolVersionMajor type="Integer" value="1"/>

```

0392     <ProtocolVersionMinor type="Integer" value="1"/>
0393     </ProtocolVersion>
0394     <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0395     <BatchCount type="Integer" value="1"/>
0396     </ResponseHeader>
0397     <BatchItem>
0398         <Operation type="Enumeration" value="Destroy"/>
0399         <ResultStatus type="Enumeration" value="Success"/>
0400         <ResponsePayload>
0401             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0402         </ResponsePayload>
0403     </BatchItem>
0404 </ResponseMessage>

```

293

294 **3.3 Mandatory Test Cases KMIP v1.2**

295 **3.1.73.3.1 TL-M-1-12 - Configuration**

296 Determine server configuration details including operations supported (only the mandatory operations are  
297 listed in the response example), objects supported (only the mandatory objects types are listed in the  
298 response example), optional server information, and optional list of application name spaces.

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="2"/>
0006     </ProtocolVersion>
0007     <BatchCount type="Integer" value="1"/>
0008   </RequestHeader>
0009   <BatchItem>
0010     <Operation type="Enumeration" value="Query"/>
0011     <UniqueBatchItemID type="ByteString" value="01"/>
0012     <RequestPayload>
0013       <QueryFunction type="Enumeration" value="QueryOperations"/>
0014       <QueryFunction type="Enumeration" value="QueryObjects"/>
0015       <QueryFunction type="Enumeration"
value="QueryServerInformation"/>
0016       <QueryFunction type="Enumeration"
value="QueryApplicationNamespaces"/>
0017     </RequestPayload>
0018   </BatchItem>
0019 </RequestMessage>
0020 <ResponseMessage>
0021   <ResponseHeader>
0022     <ProtocolVersion>
0023       <ProtocolVersionMajor type="Integer" value="1"/>
0024       <ProtocolVersionMinor type="Integer" value="2"/>
0025     </ProtocolVersion>
0026     <TimeStamp type="DateTime" value="2012-10-05T21:35:17+00:00"/>
0027     <BatchCount type="Integer" value="1"/>
0028   </ResponseHeader>
0029   <BatchItem>
0030     <Operation type="Enumeration" value="Query"/>
0031     <UniqueBatchItemID type="ByteString" value="01"/>

```

```

0032 <ResultStatus type="Enumeration" value="Success"/>
0033 <ResponsePayload>
0034   <Operation type="Enumeration" value="Query"/>
0035   <Operation type="Enumeration" value="Locate"/>
0036   <Operation type="Enumeration" value="Destroy"/>
0037   <Operation type="Enumeration" value="Get"/>
0038   <Operation type="Enumeration" value="Create"/>
0039   <Operation type="Enumeration" value="Register"/>
0040   <Operation type="Enumeration" value="GetAttributes"/>
0041   <Operation type="Enumeration" value="GetAttributeList"/>
0042   <Operation type="Enumeration" value="AddAttribute"/>
0043   <Operation type="Enumeration" value="ModifyAttribute"/>
0044   <Operation type="Enumeration" value="DeleteAttribute"/>
0045   <Operation type="Enumeration" value="Activate"/>
0046   <Operation type="Enumeration" value="Revoke"/>
0047   <Operation type="Enumeration" value="Check"/>
0048   <ObjectType type="Enumeration" value="SymmetricKey"/>
0049   <ObjectType type="Enumeration" value="Template"/>
0050   <VendorIdentification type="TextString" value="server-
vendor.com"/>
0051   <ServerInformation>
0052   </ServerInformation>
0053 </ResponsePayload>
0054 </BatchItem>
0055 </ResponseMessage>

```

299

### 300 **3.1.83.3.2 TL-M-2-12 - Write with new (created) key**

301 This case may occur when the Write operation starts with the first block on a tape. The implementation  
302 may choose which Write operations qualify for creation of a new key. Regardless of the initiating  
303 circumstances, the Tape Library requests the server to create a new AES-256 symmetric key with  
304 appropriate identifying information which is unique within the Application Namespace.

305 Additional custom attributes MAY be specified in order to:

- 306 - ensure uniqueness of the key identifier when later Locating the key via ASI
- 307 - provide human-readable information (such as the tape Barcode value)
- 308 - provide information to support client-specific purposes

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="2"/>
0006     </ProtocolVersion>
0007     <BatchOrderOption type="Boolean" value="true"/>
0008     <BatchCount type="Integer" value="2"/>
0009   </RequestHeader>
0010   <BatchItem>
0011     <Operation type="Enumeration" value="Create"/>
0012     <UniqueBatchItemID type="ByteString" value="01"/>
0013     <RequestPayload>
0014       <ObjectType type="Enumeration" value="SymmetricKey"/>
0015       <TemplateAttribute>
0016         <Attribute>
0017           <AttributeName type="TextString" value="x-ID"/>
0018           <AttributeValue type="TextString" value="TL-M-2-12"/>

```

```

0019     </Attribute>
0020     <Attribute>
0021         <AttributeName type="TextString" value="Alternative
Name" />
0022         <AttributeValue>
0023             <AlternativeNameValue type="TextString"
value="XXA012A1" />
0024             <AlternativeNameType type="Enumeration"
value="UninterpretedTextString" />
0025         </AttributeValue>
0026     </Attribute>
0027     <Attribute>
0028         <AttributeName type="TextString" value="Name" />
0029         <AttributeValue>
0030             <NameValue type="TextString"
value="AAAAAAAAA1A1AA12345678901234567" />
0031             <NameType type="Enumeration"
value="UninterpretedTextString" />
0032         </AttributeValue>
0033     </Attribute>
0034     <Attribute>
0035         <AttributeName type="TextString" value="Cryptographic
Algorithm" />
0036         <AttributeValue type="Enumeration" value="AES" />
0037     </Attribute>
0038     <Attribute>
0039         <AttributeName type="TextString" value="Cryptographic
Length" />
0040         <AttributeValue type="Integer" value="256" />
0041     </Attribute>
0042     <Attribute>
0043         <AttributeName type="TextString" value="Cryptographic
Usage Mask" />
0044         <AttributeValue type="Integer" value="Decrypt Encrypt" />
0045     </Attribute>
0046     <Attribute>
0047         <AttributeName type="TextString" value="Application
Specific Information" />
0048         <AttributeValue>
0049             <ApplicationNamespace type="TextString" value="LIBRARY-
LTO" />
0050             <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000" />
0051         </AttributeValue>
0052     </Attribute>
0053     <Attribute>
0054         <AttributeName type="TextString" value="x-Barcode" />
0055         <AttributeValue type="TextString" value="XXA012A1" />
0056     </Attribute>
0057     <Attribute>
0058         <AttributeName type="TextString" value="x-
VendorAttribute1" />
0059         <AttributeValue type="TextString" value="XXA012A1" />
0060     </Attribute>
0061     <Attribute>
0062         <AttributeName type="TextString" value="x-
VendorAttribute2" />

```



0115	</BatchItem>
0116	</ResponseMessage>

309

310 **3.1.93.3.3 TL-M-3-12 - Read an encrypted tape**

311 The Tape Library constructs an identifier string based on the method in 2.3, then requests the server to  
 312 Locate that string via ASI. A Get is then requested based on the key's unique identifier. The Tape Library  
 313 MAY update attributes associated with the Symmetric Key Managed Object. The following test case  
 314 shows extensive use of custom attributes. Custom attributes are not required if the Application Name is  
 315 unique within the Application Namespace. An implementation may also use custom attributes for vendor-  
 316 unique purposes, or to improve usability.

317 The test case destroys the key created in the previous test case to clean up after the test. Tape Library  
 318 implementations **MAY** may elect to not perform this step.

319

```

# TIME 0
0001 <RequestMessage>
0002   <RequestHeader>
0003     <ProtocolVersion>
0004       <ProtocolVersionMajor type="Integer" value="1"/>
0005       <ProtocolVersionMinor type="Integer" value="2"/>
0006     </ProtocolVersion>
0007     <BatchOrderOption type="Boolean" value="true"/>
0008     <BatchCount type="Integer" value="2"/>
0009   </RequestHeader>
0010   <BatchItem>
0011     <Operation type="Enumeration" value="Locate"/>
0012     <UniqueBatchItemID type="ByteString" value="01"/>
0013     <RequestPayload>
0014       <Attribute>
0015         <AttributeName type="TextString" value="Object Type"/>
0016         <AttributeValue type="Enumeration" value="SymmetricKey"/>
0017       </Attribute>
0018       <Attribute>
0019         <AttributeName type="TextString" value="Application Specific
Information"/>
0020         <AttributeValue>
0021           <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0022           <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0023         </AttributeValue>
0024       </Attribute>
0025     </RequestPayload>
0026   </BatchItem>
0027   <BatchItem>
0028     <Operation type="Enumeration" value="Get"/>
0029     <UniqueBatchItemID type="ByteString" value="02"/>
0030     <RequestPayload>
0031     </RequestPayload>
0032   </BatchItem>
0033 </RequestMessage>
0034 <ResponseMessage>
0035   <ResponseHeader>
0036     <ProtocolVersion>

```





```

Information"/>
0090     <AttributeValue>
0091     <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0092     <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0093     </AttributeValue>
0094     </Attribute>
0095     </RequestPayload>
0096 </BatchItem>
0097 <BatchItem>
0098   <Operation type="Enumeration" value="GetAttributeList"/>
0099   <UniqueBatchItemID type="ByteString" value="02"/>
0100   <RequestPayload>
0101   </RequestPayload>
0102 </BatchItem>
0103 </RequestMessage>
0104 <ResponseMessage>
0105   <ResponseHeader>
0106     <ProtocolVersion>
0107       <ProtocolVersionMajor type="Integer" value="1"/>
0108       <ProtocolVersionMinor type="Integer" value="2"/>
0109     </ProtocolVersion>
0110     <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0111     <BatchCount type="Integer" value="2"/>
0112   </ResponseHeader>
0113   <BatchItem>
0114     <Operation type="Enumeration" value="Locate"/>
0115     <UniqueBatchItemID type="ByteString" value="01"/>
0116     <ResultStatus type="Enumeration" value="Success"/>
0117     <ResponsePayload>
0118       <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER 0"/>
0119     </ResponsePayload>
0120   </BatchItem>
0121   <BatchItem>
0122     <Operation type="Enumeration" value="GetAttributeList"/>
0123     <UniqueBatchItemID type="ByteString" value="02"/>
0124     <ResultStatus type="Enumeration" value="Success"/>
0125     <ResponsePayload>
0126       <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0127       <AttributeName type="TextString" value="x-ID"/>
0128       <AttributeName type="TextString" value="x-Barcode"/>
0129       <AttributeName type="TextString" value="x-VendorAttribute1"/>
0130       <AttributeName type="TextString" value="x-VendorAttribute2"/>
0131       <AttributeName type="TextString" value="x-VendorAttribute3"/>
0132       <AttributeName type="TextString" value="Unique Identifier"/>
0133       <AttributeName type="TextString" value="Object Type"/>
0134       <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0135       <AttributeName type="TextString" value="Cryptographic
Length"/>
0136       <AttributeName type="TextString" value="Application Specific
Information"/>
0137       <AttributeName type="TextString" value="Cryptographic Usage
Mask"/>

```

0138	<AttributeName type="TextString" value="Digest"/>
0139	<AttributeName type="TextString" value="Initial Date"/>
0140	<AttributeName type="TextString" value="Last Change Date"/>
0141	<AttributeName type="TextString" value="Lease Time"/>
0142	<AttributeName type="TextString" value="Name"/>
0143	<AttributeName type="TextString" value="Original Creation Date"/>
0144	<AttributeName type="TextString" value="State"/>
0145	</ResponsePayload>
0146	</BatchItem>
0147	</ResponseMessage>
0148	# TIME 2 <RequestMessage>
0149	<RequestHeader>
0150	<ProtocolVersion>
0151	<ProtocolVersionMajor type="Integer" value="1"/>
0152	<ProtocolVersionMinor type="Integer" value="2"/>
0153	</ProtocolVersion>
0154	<BatchOrderOption type="Boolean" value="true"/>
0155	<BatchCount type="Integer" value="2"/>
0156	</RequestHeader>
0157	<BatchItem>
0158	<Operation type="Enumeration" value="Locate"/>
0159	<UniqueBatchItemID type="ByteString" value="01"/>
0160	<RequestPayload>
0161	<Attribute>
0162	<AttributeName type="TextString" value="Object Type"/>
0163	<AttributeValue type="Enumeration" value="SymmetricKey"/>
0164	</Attribute>
0165	<Attribute>
0166	<AttributeName type="TextString" value="Application Specific Information"/>
0167	<AttributeValue>
0168	<ApplicationNamespace type="TextString" value="LIBRARY-LTO"/>
0169	<ApplicationData type="TextString" value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000"/>
0170	</AttributeValue>
0171	</Attribute>
0172	</RequestPayload>
0173	</BatchItem>
0174	<BatchItem>
0175	<Operation type="Enumeration" value="GetAttributes"/>
0176	<UniqueBatchItemID type="ByteString" value="02"/>
0177	<RequestPayload>
0178	</RequestPayload>
0179	</BatchItem>
0180	</RequestMessage>
0181	<ResponseMessage>
0182	<ResponseHeader>
0183	<ProtocolVersion>
0184	<ProtocolVersionMajor type="Integer" value="1"/>
0185	<ProtocolVersionMinor type="Integer" value="2"/>
0186	</ProtocolVersion>
0187	<TimeStamp type="DateTime" value="2012-10-05T22:00:32+00:00"/>
0188	<BatchCount type="Integer" value="2"/>
0189	</ResponseHeader>

```

0190     <BatchItem>
0191         <Operation type="Enumeration" value="Locate"/>
0192         <UniqueBatchItemID type="ByteString" value="01"/>
0193         <ResultStatus type="Enumeration" value="Success"/>
0194         <ResponsePayload>
0195             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0196         </ResponsePayload>
0197     </BatchItem>
0198     <BatchItem>
0199         <Operation type="Enumeration" value="GetAttributes"/>
0200         <UniqueBatchItemID type="ByteString" value="02"/>
0201         <ResultStatus type="Enumeration" value="Success"/>
0202         <ResponsePayload>
0203             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0204             <Attribute>
0205                 <AttributeName type="TextString" value="x-ID"/>
0206                 <AttributeValue type="TextString" value="TL-M-2-12"/>
0207             </Attribute>
0208             <Attribute>
0209                 <AttributeName type="TextString" value="x-Barcode"/>
0210                 <AttributeValue type="TextString" value="XXA012A1"/>
0211             </Attribute>
0212             <Attribute>
0213                 <AttributeName type="TextString" value="x-
VendorAttribute1"/>
0214                 <AttributeValue type="TextString" value="XXA012A1"/>
0215             </Attribute>
0216             <Attribute>
0217                 <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0218                 <AttributeValue type="Integer" value="0"/>
0219             </Attribute>
0220             <Attribute>
0221                 <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0222                 <AttributeValue type="DateTime" value="2012-10-
05T22:08:19+00:00"/>
0223             </Attribute>
0224             <Attribute>
0225                 <AttributeName type="TextString" value="Unique Identifier"/>
0226                 <AttributeValue type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0227             </Attribute>
0228             <Attribute>
0229                 <AttributeName type="TextString" value="Object Type"/>
0230                 <AttributeValue type="Enumeration" value="SymmetricKey"/>
0231             </Attribute>
0232             <Attribute>
0233                 <AttributeName type="TextString" value="Cryptographic
Algorithm"/>
0234                 <AttributeValue type="Enumeration" value="AES"/>
0235             </Attribute>
0236             <Attribute>
0237                 <AttributeName type="TextString" value="Cryptographic
Length"/>
0238                 <AttributeValue type="Integer" value="256"/>

```

```

0239     </Attribute>
0240     <Attribute>
0241         <AttributeName type="TextString" value="Alternative Name"/>
0242         <AttributeValue>
0243             <AlternativeNameValue type="TextString" value="XXA012A1"/>
0244             <AlternativeNameType type="Enumeration"
value="UninterpretedTextString"/>
0245         </AttributeValue>
0246     </Attribute>
0247     <Attribute>
0248         <AttributeName type="TextString" value="Application Specific
Information"/>
0249         <AttributeValue>
0250             <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0251             <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0252         </AttributeValue>
0253     </Attribute>
0254     <Attribute>
0255         <AttributeName type="TextString" value="Cryptographic Usage
Mask"/>
0256         <AttributeValue type="Integer" value="Decrypt Encrypt"/>
0257     </Attribute>
0258     <Attribute>
0259         <AttributeName type="TextString" value="Digest"/>
0260         <AttributeValue>
0261             <HashingAlgorithm type="Enumeration" value="SHA_256"/>
0262             <DigestValue type="ByteString"
value="0bd3d7ada745e4eb34cc9d26cc84d4852a15906b2f77f80d01fc31c28108c
49b"/>
0263         <KeyFormatType type="Enumeration" value="Raw"/>
0264     </AttributeValue>
0265 </Attribute>
0266 <Attribute>
0267     <AttributeName type="TextString" value="Fresh"/>
0268     <AttributeValue type="Boolean" value="false"/>
0269 </Attribute>
0270 <Attribute>
0271     <AttributeName type="TextString" value="Initial Date"/>
0272     <AttributeValue type="DateTime" value="2012-10-
05T21:41:46+00:00"/>
0273 </Attribute>
0274 <Attribute>
0275     <AttributeName type="TextString" value="Last Change Date"/>
0276     <AttributeValue type="DateTime" value="2012-10-
05T21:41:46+00:00"/>
0277 </Attribute>
0278 <Attribute>
0279     <AttributeName type="TextString" value="Lease Time"/>
0280     <AttributeValue type="Interval" value="3600"/>
0281 </Attribute>
0282 <Attribute>
0283     <AttributeName type="TextString" value="Name"/>
0284     <AttributeValue>
0285         <NameValue type="TextString"
value="AAAAAAA1A1AA12345678901234567"/>

```

```

0286         <NameType type="Enumeration"
value="UninterpretedTextString"/>
0287         </AttributeValue>
0288         </Attribute>
0289         <Attribute>
0290         <AttributeName type="TextString" value="Original Creation
Date"/>
0291         <AttributeValue type="DateTime" value="2012-10-
05T21:41:46+00:00"/>
0292         </Attribute>
0293         <Attribute>
0294         <AttributeName type="TextString" value="State"/>
0295         <AttributeValue type="Enumeration" value="PreActive"/>
0296         </Attribute>
0297         </ResponsePayload>
0298         </BatchItem>
0299     </ResponseMessage>

# TIME 3
0300 <RequestMessage>
0301 <RequestHeader>
0302     <ProtocolVersion>
0303         <ProtocolVersionMajor type="Integer" value="1"/>
0304         <ProtocolVersionMinor type="Integer" value="2"/>
0305     </ProtocolVersion>
0306     <BatchOrderOption type="Boolean" value="true"/>
0307     <BatchCount type="Integer" value="3"/>
0308 </RequestHeader>
0309 <BatchItem>
0310     <Operation type="Enumeration" value="Locate"/>
0311     <UniqueBatchItemID type="ByteString" value="01"/>
0312     <RequestPayload>
0313         <Attribute>
0314             <AttributeName type="TextString" value="Object Type"/>
0315             <AttributeValue type="Enumeration" value="SymmetricKey"/>
0316         </Attribute>
0317         <Attribute>
0318             <AttributeName type="TextString" value="Application Specific
Information"/>
0319             <AttributeValue>
0320                 <ApplicationNamespace type="TextString" value="LIBRARY-
LTO"/>
0321                 <ApplicationData type="TextString"
value="123456789ABCDEF123456789123456789ABCDEF123456789ABCDEF1234000
000"/>
0322             </AttributeValue>
0323         </Attribute>
0324     </RequestPayload>
0325 </BatchItem>
0326 <BatchItem>
0327     <Operation type="Enumeration" value="ModifyAttribute"/>
0328     <UniqueBatchItemID type="ByteString" value="02"/>
0329     <RequestPayload>
0330         <Attribute>
0331             <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0332             <AttributeValue type="DateTime" value="2012-10-
05T22:27:06+00:00"/>
0333         </Attribute>

```

```

0334     </RequestPayload>
0335 </BatchItem>
0336 <BatchItem>
0337     <Operation type="Enumeration" value="ModifyAttribute"/>
0338     <UniqueBatchItemID type="ByteString" value="03"/>
0339     <RequestPayload>
0340         <Attribute>
0341             <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0342             <AttributeValue type="Integer" value="1"/>
0343         </Attribute>
0344     </RequestPayload>
0345 </BatchItem>
0346 </RequestMessage>
0347 <ResponseMessage>
0348     <ResponseHeader>
0349         <ProtocolVersion>
0350             <ProtocolVersionMajor type="Integer" value="1"/>
0351             <ProtocolVersionMinor type="Integer" value="2"/>
0352         </ProtocolVersion>
0353         <TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0354         <BatchCount type="Integer" value="3"/>
0355     </ResponseHeader>
0356     <BatchItem>
0357         <Operation type="Enumeration" value="Locate"/>
0358         <UniqueBatchItemID type="ByteString" value="01"/>
0359         <ResultStatus type="Enumeration" value="Success"/>
0360         <ResponsePayload>
0361             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0362         </ResponsePayload>
0363     </BatchItem>
0364     <BatchItem>
0365         <Operation type="Enumeration" value="ModifyAttribute"/>
0366         <UniqueBatchItemID type="ByteString" value="02"/>
0367         <ResultStatus type="Enumeration" value="Success"/>
0368         <ResponsePayload>
0369             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0370             <Attribute>
0371                 <AttributeName type="TextString" value="x-
VendorAttribute3"/>
0372                 <AttributeValue type="DateTime" value="2012-10-
05T22:27:06+00:00"/>
0373             </Attribute>
0374         </ResponsePayload>
0375     </BatchItem>
0376     <BatchItem>
0377         <Operation type="Enumeration" value="ModifyAttribute"/>
0378         <UniqueBatchItemID type="ByteString" value="03"/>
0379         <ResultStatus type="Enumeration" value="Success"/>
0380         <ResponsePayload>
0381             <UniqueIdentifier type="TextString"
value="$UNIQUE_IDENTIFIER_0"/>
0382             <Attribute>
0383                 <AttributeName type="TextString" value="x-
VendorAttribute2"/>
0384                 <AttributeValue type="Integer" value="1"/>

```

0385	</Attribute>
0386	</ResponsePayload>
0387	</BatchItem>
0388	</ResponseMessage>
0389	# TIME 4 <RequestMessage>
0390	<RequestHeader>
0391	<ProtocolVersion>
0392	<ProtocolVersionMajor type="Integer" value="1"/>
0393	<ProtocolVersionMinor type="Integer" value="2"/>
0394	</ProtocolVersion>
0395	<BatchCount type="Integer" value="1"/>
0396	</RequestHeader>
0397	<BatchItem>
0398	<Operation type="Enumeration" value="Destroy"/>
0399	<RequestPayload>
0400	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0401	</RequestPayload>
0402	</BatchItem>
0403	</RequestMessage>
0404	<ResponseMessage>
0405	<ResponseHeader>
0406	<ProtocolVersion>
0407	<ProtocolVersionMajor type="Integer" value="1"/>
0408	<ProtocolVersionMinor type="Integer" value="2"/>
0409	</ProtocolVersion>
0410	<TimeStamp type="DateTime" value="2012-10-05T22:00:33+00:00"/>
0411	<BatchCount type="Integer" value="1"/>
0412	</ResponseHeader>
0413	<BatchItem>
0414	<Operation type="Enumeration" value="Destroy"/>
0415	<ResultStatus type="Enumeration" value="Success"/>
0416	<ResponsePayload>
0417	<UniqueIdentifier type="TextString" value="\$UNIQUE_IDENTIFIER_0"/>
0418	</ResponsePayload>
0419	</BatchItem>
0420	</ResponseMessage>

320

321

322



## 323 4 Conformance

### 324 4.1 Tape Library Client KMIP v1.0 Conformance Statement

325 KMIP client ~~and server~~ implementations conformant to this profile:

- 326 1. SHALL support the Authentication Suite conditions (2.1);
- 327 2. SHALL support the ~~Baseline Tape Library~~Baseline Tape Library - Client conditions (~~2.22.2~~);
- 328 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (~~2.42.3~~);
- 329 4. SHALL support the Using Alternative Name for tape media barcode conditions (0); and
- 330 5. SHALL support all the Mandatory Test Cases KMIP v1.0 (3.1).

### 331 4.2 Tape Library Client KMIP v1.1 Conformance

332 KMIP client implementations conformant to this profile:

- 333 1. SHALL support the Authentication Suite conditions (2.1);
- 334 2. SHALL support the Baseline Tape Library - Client conditions (2.2);
- 335 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (2.4);
- 336 4.4. SHALL support the Using Alternative Name for tape media barcode conditions (~~02.4~~); and
- 337 5. SHALL support all the ~~Mandatory Test Cases~~Mandatory Test Cases KMIP v1.1 (3.1) for each
- 338 supported protocol version (major3.2).

### 339 4.3 Tape Library Client KMIP v1.2 Conformance

340 KMIP client implementations conformant to this profile:

- 341 1. SHALL support the Authentication Suite conditions (2.1);
- 342 2. SHALL support the Baseline Tape Library - Client conditions (2.2);
- 343 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (2.4);
- 344 4. SHALL support the Using Alternative Name for tape media barcode conditions (0); and ~~minor~~.
- 345
- 346 5. SHALL support all the Mandatory Test Cases KMIP v1.2 (3.3).

### 347 4.4 Tape Library Server KMIP v1.0 Conformance

348 KMIP server implementations conformant to this profile:

- 349 1. SHALL support the Authentication Suite conditions (2.1);
- 350 2. SHALL support the Baseline Tape Library - Server conditions (0);
- 351 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (2.4);
- 352 4. SHALL support the Using Alternative Name for tape media barcode conditions (0); and
- 353 5. SHALL support all the Mandatory Test Cases KMIP v1.0 (3.1).

### 354 4.5 Tape Library Server KMIP v1.1 Conformance

355 KMIP server implementations conformant to this profile:

- 356 1. SHALL support the Authentication Suite conditions (2.1);
- 357 2. SHALL support the Baseline Tape Library - Server conditions (0);
- 358 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (2.4);

- 359 4. SHALL support the Using Alternative Name for tape media barcode conditions (0); and  
360 5. SHALL support all the Mandatory Test Cases KMIP v1.1 (3.2).

## 361 **4.6 Tape Library Server KMIP v1.2 Conformance**

362 KMIP server implementations conformant to this profile:

- 363 1. SHALL support the Authentication Suite conditions (2.1);  
364 2. SHALL support the Baseline Tape Library - Server conditions (0);  
365 3. SHALL support the Using Application Specific Information for Key Identifiers conditions (2.4);  
366 4. SHALL support the Using Alternative Name for tape media barcode conditions (0); and  
367 5. SHALL support all the Mandatory Test Cases KMIP v1.2 (3.3).

### 368 **4.24.7 Permitted Test Case Variations**

369 Whilst the test cases provided in this Profile define the allowed request and response content, some  
370 inherent variations MAY occur and are permitted within a successfully completed test case.

371 Each test case MAY include allowed variations in the description of the test case in addition to the  
372 variations noted in this section.

373 Other variations not explicitly noted in this Profile SHALL be deemed non-conformant.

#### 374 **4.2.14.7.1 Variable Items**

375 An implementation conformant to this Profile MAY vary the following values:

- 376 1. UniqueIdentifier  
377 2. PrivateKeyUniqueIdentifier  
378 3. PublicKeyUniqueIdentifier  
379 4. UniqueBatchItemIdentifier  
380 5. AsynchronousCorrelationValue  
381 6. TimeStamp  
382 7. KeyValue / KeyMaterial including:  
383 a. key material content returned for managed cryptographic objects which are generated by  
384 the server  
385 b. wrapped versions of keys where the wrapping key is dynamic or the wrapping contains  
386 variable output for each wrap operation  
387 8. For response containing the output of cryptographic operation in Data / SignatureData/ MACData  
388 / IVCounterNonce where:  
389 a. the managed object is generated by the server; or  
390 b. the operation inherently contains variable output  
391 9. For the following DateTime attributes where the value is not specified in the request as a fixed  
392 DateTime value:  
393 a. ActivationDate  
394 b. ArchiveDate  
395 c. CompromiseDate  
396 d. CompromiseOccurrenceDate  
397 e. DeactivationDate  
398 f. DestroyDate  
399 g. InitialDate

- 400 h. LastChangeDate
- 401 i. ProtectStartDate
- 402 j. ProcessStopDate
- 403 k. ValidityDate
- 404 l. OriginalCreationDate
- 405 10. LinkedObjectIdentifier
- 406 11. DigestValue
  - 407 a. For those managed cryptographic objects which are dynamically generated
- 408 12. KeyFormatType
  - 409 a. The key format type selected by the server when it creates managed objects
- 410 13. Digest
  - 411 a. The HashingAlgorithm selected by the server when it calculates the digest for a managed
  - 412 object for which it has access to the key material
  - 413 b. The Digest Value
- 414 14. Extensions reported in Query for ExtensionList and ExtensionMap
- 415 15. Application Namespaces reported in Query
- 416 16. Object Types reported in Query other than those noted as required in this profile
- 417 17. Operation Types reported in Query other than those noted as required in this profile (or any
- 418 referenced profile documents)
- 419 18. For TextString attribute values containing test identifiers:
  - 420 a. Additional vendor or application prefixes
- 421 19. Additional attributes beyond those noted in the response

422  
423 An implementation conformant to this Profile MAY allow the following response variations:

- 424 | **4-20.** Object Group values – May or may not return one or more Object Group values not
- 425 | included in the requests
- 426 | **2-21.** y-CustomAttributes – May or may not include additional server-specific associated
- 427 | attributes not included in requests
- 428 | **3-22.** Message Extensions – May or may not include additional (non-critical) vendor extensions
- 429 | **4-23.** TemplateAttribute – May or may not be included in responses where the Template
- 430 | Attribute response is noted as optional in [KMIP-SPEC]
- 431 | **5-24.** AttributeIndex – May or may not include Attribute Index value where the Attribute Index
- 432 | value is 0 for Protocol Versions 1.1 and above.
- 433 | **6-25.** ResultMessage – May or may not be included in responses and the value (if included)
- 434 | may vary from the text contained within the test case.
- 435 | **7-26.** The list of Protocol Versions returned in a DiscoverVersion response may include
- 436 | additional protocol versions if the request has not specified a list of client supported Protocol
- 437 | Versions.
- 438 | **8-27.** VendorIdentification - The value (if included) may vary from the text contained within the
- 439 | test case.

#### 440 **4.2.24.7.2 Variable behavior**

441 An implementation conformant to this Profile SHALL allow variation of the following behavior:

- 442 1. A test may omit the clean-up requests and responses (containing Revoke and/or Destroy) at the
- 443 end of the test provided there is a separate mechanism to remove the created objects during
- 444 testing.

- 445 2. A test may omit the test identifiers if the client is unable to include them in requests. This includes  
446 the following attributes:
- 447 a. Name; and
  - 448 b. x-ID
- 449
- 450 3. A test MAY perform requests with multiple batch items or as multiple requests with a single batch  
451 item provided the sequence of operations are equivalent
- 452 4. A request MAY contain an optional *Authentication* [KMIP SPEC] structure within each request  
453

---

## Appendix A. Acknowledgments

454 The following individuals have participated in the creation of this specification and are gratefully  
455 acknowledged:

456 **Participants:**

457 Hal Aldridge, Sypris Electronics  
458 Mike Allen, Symantec  
459 Gordon Arnold, IBM  
460 Todd Arnold, IBM  
461 Richard Austin, Hewlett-Packard  
462 Lars Bagnert, PrimeKey  
463 Elaine Barker, NIST  
464 Peter Bartok, Venafi, Inc.  
465 Tom Benjamin, IBM  
466 Anthony Berglas, Cryptsoft  
467 Mathias Björkqvist, IBM  
468 Kevin Bocket, Venafi  
469 Anne Bolgert, IBM  
470 Alan Brown, Thales e-Security  
471 Tim Bruce, CA Technologies  
472 Chris Burchett, Credant Technologies, Inc.  
473 Kelley Burgin, National Security Agency  
474 Robert Burns, Thales e-Security  
475 Chuck Castleton, Venafi  
476 Kenli Chong, QuintessenceLabs  
477 John Clark, Hewlett-Packard  
478 Tom Clifford, Symantec Corp.  
479 Doron Cohen, SafeNet, Inc  
480 Tony Cox, Cryptsoft  
481 Russell Dietz, SafeNet, Inc  
482 Graydon Dodson, Lexmark International Inc.  
483 Vinod Duggirala, EMC Corporation  
484 Chris Dunn, SafeNet, Inc.  
485 Michael Duren, Sypris Electronics  
486 James Dzierzanowski, American Express CCoE  
487 Faisal Faruqui, Thales e-Security  
488 Stan Feather, Hewlett-Packard  
489 David Finkelstein, Symantec Corp.  
490 James Fitzgerald, SafeNet, Inc.  
491 Indra Fitzgerald, Hewlett-Packard  
492 Judith Furlong, EMC Corporation  
493 Susan Gleeson, Oracle  
494 Robert Griffin, EMC Corporation  
495 Paul Grojean, Individual  
496 Robert Haas, IBM  
497 Thomas Hardjono, M.I.T.  
498 ChengDong He, Huawei Technologies Co., Ltd.  
499 Steve He, Vormetric  
500 Kurt Heberlein, Hewlett-Packard  
501 Larry Hofer, Emulex Corporation  
502 Maryann Hondo, IBM  
503 Walt Hubis, NetApp  
504 Tim Hudson, Cryptsoft  
505 Jonas Iggbom, Venafi, Inc.

506 Sitaram Inguva, American Express CcoE  
507 Jay Jacobs, Target Corporation  
508 Glen Jaquette, IBM  
509 Mahadev Karadiguddi, NetApp  
510 Greg Kazmierczak, Wave Systems Corp.  
511 Marc Kenig, SafeNet, Inc.  
512 Mark Knight, Thales e-Security  
513 Kathy Kriese, Symantec Corporation  
514 Mark Lambiase, SecureAuth  
515 John Leiseboer, Quintessence Labs  
516 Hal Lockhart, Oracle Corporation  
517 Robert Lockhart, Thales e-Security  
518 Anne Luk, Cryptsoft  
519 Sairam Manidi, Freescale  
520 Luther Martin, Voltage Security  
521 Neil McEvoy, iFOSSF  
522 Marina Milshtein, Individual  
523 Dale Moberg, Axway Software  
524 Jishnu Mukeri, Hewlett-Packard  
525 Bryan Olson, Hewlett-Packard  
526 John Peck, IBM  
527 Rob Philpott, EMC Corporation  
528 Denis Pochuev, SafeNet, Inc.  
529 Reid Poole, Venafi, Inc.  
530 Ajai Puri, SafeNet, Inc.  
531 Saravanan Ramalingam, Thales e-Security  
532 Peter Reed, SafeNet, Inc.  
533 Bruce Rich, IBM  
534 Christina Richards, American Express CcoE  
535 Warren Robbins, Dell  
536 Peter Robinson, EMC Corporation  
537 Scott Rotondo, Oracle  
538 Saikat Saha, SafeNet, Inc.  
539 Anil Saldhana, Red Hat  
540 Subhash Sankuratripati, NetApp  
541 Boris Schumperli, Cryptomathic  
542 Greg Singh, QuintessenceLabs  
543 David Smith, Venafi, Inc  
544 Brian Spector, Certivox  
545 Terence Spies, Voltage Security  
546 Deborah Steckroth, RouteOne LLC  
547 Michael Stevens, QuintessenceLabs  
548 Marcus Streets, Thales e-Security  
549 Satish Sundar, IBM  
550 Kiran Thota, Vmware  
551 Somanchi Trinath, Freescale Semiconductor, Inc.  
552 Nathan Turajski, Thales e-Security  
553 Sean Turner, IECA, Inc.  
554 Paul Turner, Venafi, Inc.  
555 Rod Wideman, Quantum Corporation  
556 Steven Wierenga, Hewlett-Packard  
557 Jin Wong, QuintessenceLabs  
558 Sameer Yami, Thales e-Security  
559 Peter Yee, EMC Corporation  
560 Krishna Yellepeddy, IBM  
561 Catherine Ying, SafeNet, Inc.  
562 Tatu Ylonen, SSH Communications Security (Tectia Corp)

563 Michael Yoder, Vormetric. Inc.  
564 Magda Zdunkiewicz, Cryptsoft  
565 Peter Zelechowski, Election Systems & Software

## Appendix B. KMIP Specification Cross Reference

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
<b>1 Introduction</b>			
<i>Non-Normative References</i>	1.3.	1.3.	1.3.
<i>Normative References</i>	1.2.	1.2.	1.2.
<i>Terminology</i>	1.1.	1.1.	1.1.
<b>2 Objects</b>			
<i>Attribute</i>	2.1.1.	2.1.1.	2.1.1.
<i>Base Objects</i>	2.1.	2.1.	2.1.
<i>Certificate</i>	2.2.1.	2.2.1.	2.2.1.
<i>Credential</i>	2.1.2.	2.1.2.	2.1.2.
<i>Data</i>	-	-	2.1.10.
<i>Data Length</i>	-	-	2.1.11.
<i>Extension Information</i>	-	2.1.9.	2.1.9.
<i>Key Block</i>	2.1.3.	2.1.3.	2.1.3.
<i>Key Value</i>	2.1.4.	2.1.4.	2.1.4.
<i>Key Wrapping Data</i>	2.1.5.	2.1.5.	2.1.5.
<i>Key Wrapping Specification</i>	2.1.6.	2.1.6.	2.1.6.
<i>MAC Data</i>	-	-	2.1.13.
<i>Managed Objects</i>	2.2.	2.2.	2.2.
<i>Nonce</i>	-	-	2.1.14.
<i>Opaque Object</i>	2.2.8.	2.2.8.	2.2.8.
<i>PGP Key</i>	-	-	2.2.9.
<i>Private Key</i>	2.2.4.	2.2.4.	2.2.4.
<i>Public Key</i>	2.2.3.	2.2.3.	2.2.3.
<i>Secret Data</i>	2.2.7.	2.2.7.	2.2.7.
<i>Signature Data</i>	-	-	2.1.12.
<i>Split Key</i>	2.2.5.	2.2.5.	2.2.5.
<i>Symmetric Key</i>	2.2.2.	2.2.2.	2.2.2.
<i>Template</i>	2.2.6.	2.2.6.	2.2.6.
<i>Template-Attribute Structures</i>	2.1.8.	2.1.8.	2.1.8.
<i>Transparent DH Private Key</i>	2.1.7.6.	2.1.7.6.	2.1.7.6.
<i>Transparent DH Public Key</i>	2.1.7.7.	2.1.7.7.	2.1.7.7.
<i>Transparent DSA Private Key</i>	2.1.7.2.	2.1.7.2.	2.1.7.2.
<i>Transparent DSA Public Key</i>	2.1.7.3.	2.1.7.3.	2.1.7.3.
<i>Transparent ECDH Private Key</i>	2.1.7.10.	2.1.7.10.	2.1.7.10.
<i>Transparent ECDH Public Key</i>	2.1.7.11.	2.1.7.11.	2.1.7.11.
<i>Transparent ECDSA Private Key</i>	2.1.7.8.	2.1.7.8.	2.1.7.8.
<i>Transparent ECDSA Public Key</i>	2.1.7.9.	2.1.7.9.	2.1.7.9.
<i>Transparent ECMQV Private Key</i>	2.1.7.12.	2.1.7.12.	2.1.7.12.
<i>Transparent ECMQV Public Key</i>	2.1.7.13.	2.1.7.13.	2.1.7.13.
<i>Transparent Key Structures</i>	2.1.7.	2.1.7.	2.1.7.
<i>Transparent RSA Private Key</i>	2.1.7.4.	2.1.7.4.	2.1.7.4.
<i>Transparent RSA Public Key</i>	2.1.7.5.	2.1.7.5.	2.1.7.5.
<i>Transparent Symmetric Key</i>	2.1.7.1.	2.1.7.1.	2.1.7.1.
<b>3 Attributes</b>			
<i>Activation Date</i>	3.19.	3.24.	3.24.
<i>Alternative Name</i>	-	-	3.40.
<i>Application Specific Information</i>	3.30.	3.36.	3.36.
<i>Archive Date</i>	3.27.	3.32.	3.32.



<b>Reference Term</b>	<b>KMIP 1.0</b>	<b>KMIP 1.1</b>	<b>KMIP 1.2</b>
<i>Attributes</i>	3	3	3
<i>Certificate Identifier</i>	3.9.	3.13.	3.13.
<i>Certificate Issuer</i>	3.11.	3.15.	3.15.
<i>Certificate Length</i>	-	3.9.	3.9.
<i>Certificate Subject</i>	3.10.	3.14.	3.14.
<i>Certificate Type</i>	3.8.	3.8.	3.8.
<i>Compromise Date</i>	3.25.	3.30.	3.30.
<i>Compromise Occurrence Date</i>	3.24.	3.29.	3.29.
<i>Contact Information</i>	3.31.	3.37.	3.37.
<i>Cryptographic Algorithm</i>	3.4.	3.4.	3.4.
<i>Cryptographic Domain Parameters</i>	3.7.	3.7.	3.7.
<i>Cryptographic Length</i>	3.5.	3.5.	3.5.
<i>Cryptographic Parameters</i>	3.6.	3.6.	3.6.
<i>Custom Attribute</i>	3.33.	3.39.	3.39.
<i>Deactivation Date</i>	3.22.	3.27.	3.27.
<i>Default Operation Policy</i>	3.13.2.	3.18.2.	3.18.2.
<i>Default Operation Policy for Certificates and Public Key Objects</i>	3.13.2.2.	3.18.2.2.	3.18.2.2.
<i>Default Operation Policy for Secret Objects</i>	3.13.2.1.	3.18.2.1.	3.18.2.1.
<i>Default Operation Policy for Template Objects</i>	3.13.2.3.	3.18.2.3.	3.18.2.3.
<i>Destroy Date</i>	3.23.	3.28.	3.28.
<i>Digest</i>	3.12.	3.17.	3.17.
<i>Digital Signature Algorithm</i>	-	3.16.	3.16.
<i>Fresh</i>	-	3.34.	3.34.
<i>Initial Date</i>	3.18.	3.23.	3.23.
<i>Key Value Location</i>	-	-	3.42.
<i>Key Value Present</i>	-	-	3.41.
<i>Last Change Date</i>	3.32.	3.38.	3.38.
<i>Lease Time</i>	3.15.	3.20.	3.20.
<i>Link</i>	3.29.	3.35.	3.35.
<i>Name</i>	3.2.	3.2.	3.2.
<i>Object Group</i>	3.28.	3.33.	3.33.
<i>Object Type</i>	3.3.	3.3.	3.3.
<i>Operation Policy Name</i>	3.13.	3.18.	3.18.
<i>Operations outside of operation policy control</i>	3.13.1.	3.18.1.	3.18.1.
<i>Original Creation Date</i>	-	-	3.43.
<i>Process Start Date</i>	3.20.	3.25.	3.25.
<i>Protect Stop Date</i>	3.21.	3.26.	3.26.
<i>Revocation Reason</i>	3.26.	3.31.	3.31.
<i>State</i>	3.17.	3.22.	3.22.
<i>Unique Identifier</i>	3.1.	3.1.	3.1.
<i>Usage Limits</i>	3.16.	3.21.	3.21.
<i>X.509 Certificate Identifier</i>	-	3.10.	3.10.
<i>X.509 Certificate Issuer</i>	-	3.12.	3.12.
<i>X.509 Certificate Subject</i>	-	3.11.	3.11.
<b>4 Client-to-Server Operations</b>			
<i>Activate</i>	4.18.	4.19.	4.19.
<i>Add Attribute</i>	4.13.	4.14.	4.14.
<i>Archive</i>	4.21.	4.22.	4.22.
<i>Cancel</i>	4.25.	4.27.	4.27.
<i>Certify</i>	4.6.	4.7.	4.7.
<i>Check</i>	4.9.	4.10.	4.10.
<i>Create</i>	4.1.	4.1.	4.1.
<i>Create Key Pair</i>	4.2.	4.2.	4.2.

<b>Reference Term</b>	<b>KMIP 1.0</b>	<b>KMIP 1.1</b>	<b>KMIP 1.2</b>
<i>Create Split Key</i>	-	-	4.38.
<i>Decrypt</i>	-	-	4.30.
<i>Delete Attribute</i>	4.15.	4.16.	4.16.
<i>Derive Key</i>	4.5.	4.6.	4.6.
<i>Destroy</i>	4.20.	4.21.	4.21.
<i>Discover Versions</i>	-	4.26.	4.26.
<i>Encrypt</i>	-	-	4.29.
<i>Get</i>	4.10.	4.11.	4.11.
<i>Get Attribute List</i>	4.12.	4.13.	4.13.
<i>Get Attributes</i>	4.11.	4.12.	4.12.
<i>Get Usage Allocation</i>	4.17.	4.18.	4.18.
<i>Hash</i>	-	-	4.37.
<i>Join Split Key</i>	-	-	4.39.
<i>Locate</i>	4.8.	4.9.	4.9.
<i>MAC</i>	-	-	4.33.
<i>MAC Verify</i>	-	-	4.34.
<i>Modify Attribute</i>	4.14.	4.15.	4.15.
<i>Obtain Lease</i>	4.16.	4.17.	4.17.
<i>Poll</i>	4.26.	4.28.	4.28.
<i>Query</i>	4.24.	4.25.	4.25.
<i>Re-certify</i>	4.7.	4.8.	4.8.
<i>Recover</i>	4.22.	4.23.	4.23.
<i>Register</i>	4.3.	4.3.	4.3.
<i>Re-key</i>	4.4.	4.4.	4.4.
<i>Re-key Key Pair</i>	-	4.5.	4.5.
<i>Revoke</i>	4.19.	4.20.	4.20.
<i>RNG Retrieve</i>	-	-	4.35.
<i>RNG Seed</i>	-	-	4.36.
<i>Sign</i>	-	-	4.31.
<i>Signature Verify</i>	-	-	4.32.
<i>Validate</i>	4.23.	4.24.	4.24.
<b>5 Server-to-Client Operations</b>			
<i>Notify</i>	5.1.	5.1.	5.1.
<i>Put</i>	5.2.	5.2.	5.2.
<b>6 Message Contents</b>			
<i>Asynchronous Correlation Value</i>	6.8.	6.8.	6.8.
<i>Asynchronous Indicator</i>	6.7.	6.7.	6.7.
<i>Attestation Capable Indicator</i>	-	-	6.17.
<i>Batch Count</i>	6.14.	6.14.	6.14.
<i>Batch Error Continuation Option</i>	6.13.	6.13.	6.13.
<i>Batch Item</i>	6.15.	6.15.	6.15.
<i>Batch Order Option</i>	6.12.	6.12.	6.12.
<i>Maximum Response Size</i>	6.3.	6.3.	6.3.
<i>Message Extension</i>	6.16.	6.16.	6.16.
<i>Operation</i>	6.2.	6.2.	6.2.
<i>Protocol Version</i>	6.1.	6.1.	6.1.
<i>Result Message</i>	6.11.	6.11.	6.11.
<i>Result Reason</i>	6.10.	6.10.	6.10.
<i>Result Status</i>	6.9.	6.9.	6.9.
<i>Time Stamp</i>	6.5.	6.5.	6.5.
<i>Unique Batch Item ID</i>	6.4.	6.4.	6.4.
<b>7 Message Format</b>			

<b>Reference Term</b>	<b>KMIP 1.0</b>	<b>KMIP 1.1</b>	<b>KMIP 1.2</b>
<i>Message Structure</i>	7.1.	7.1.	7.1.
<i>Operations</i>	7.2.	7.2.	7.2.
<b>8 Authentication</b>			
<i>Authentication</i>	8	8	8
<b>9 Message Encoding</b>			
<i>Alternative Name Type Enumeration</i>	-	-	9.1.3.2.34.
<i>Attestation Type Enumeration</i>	-	-	9.1.3.2.36.
<i>Batch Error Continuation Option Enumeration</i>	9.1.3.2.29.	9.1.3.2.30.	9.1.3.2.30.
<i>Bit Masks</i>	9.1.3.3.	9.1.3.3.	9.1.3.3.
<i>Block Cipher Mode Enumeration</i>	9.1.3.2.13.	9.1.3.2.14.	9.1.3.2.14.
<i>Cancellation Result Enumeration</i>	9.1.3.2.24.	9.1.3.2.25.	9.1.3.2.25.
<i>Certificate Request Type Enumeration</i>	9.1.3.2.21.	9.1.3.2.22.	9.1.3.2.22.
<i>Certificate Type Enumeration</i>	9.1.3.2.6.	9.1.3.2.6.	9.1.3.2.6.
<i>Credential Type Enumeration</i>	9.1.3.2.1.	9.1.3.2.1.	9.1.3.2.1.
<i>Cryptographic Algorithm Enumeration</i>	9.1.3.2.12.	9.1.3.2.13.	9.1.3.2.13.
<i>Cryptographic Usage Mask</i>	9.1.3.3.1.	9.1.3.3.1.	9.1.3.3.1.
<i>Defined Values</i>	9.1.3.	9.1.3.	9.1.3.
<i>Derivation Method Enumeration</i>	9.1.3.2.20.	9.1.3.2.21.	9.1.3.2.21.
<i>Digital Signature Algorithm Enumeration</i>	-	9.1.3.2.7.	9.1.3.2.7.
<i>Encoding Option Enumeration</i>	-	9.1.3.2.32.	9.1.3.2.32.
<i>Enumerations</i>	9.1.3.2.	9.1.3.2.	9.1.3.2.
<i>Examples</i>	9.1.2.	9.1.2.	9.1.2.
<i>Hashing Algorithm Enumeration</i>	9.1.3.2.15.	9.1.3.2.16.	9.1.3.2.16.
<i>Item Length</i>	9.1.1.3.	9.1.1.3.	9.1.1.3.
<i>Item Tag</i>	9.1.1.1.	9.1.1.1.	9.1.1.1.
<i>Item Type</i>	9.1.1.2.	9.1.1.2.	9.1.1.2.
<i>Item Value</i>	9.1.1.4.	9.1.1.4.	9.1.1.4.
<i>Key Compression Type Enumeration</i>	9.1.3.2.2.	9.1.3.2.2.	9.1.3.2.2.
<i>Key Format Type Enumeration</i>	9.1.3.2.3.	9.1.3.2.3.	9.1.3.2.3.
<i>Key Role Type Enumeration</i>	9.1.3.2.16.	9.1.3.2.17.	9.1.3.2.17.
<i>Key Value Location Type Enumeration</i>	-	-	9.1.3.2.35.
<i>Link Type Enumeration</i>	9.1.3.2.19.	9.1.3.2.20.	9.1.3.2.20.
<i>Name Type Enumeration</i>	9.1.3.2.10.	9.1.3.2.11.	9.1.3.2.11.
<i>Object Group Member Enumeration</i>	-	9.1.3.2.33.	9.1.3.2.33.
<i>Object Type Enumeration</i>	9.1.3.2.11.	9.1.3.2.12.	9.1.3.2.12.
<i>Opaque Data Type Enumeration</i>	9.1.3.2.9.	9.1.3.2.10.	9.1.3.2.10.
<i>Operation Enumeration</i>	9.1.3.2.26.	9.1.3.2.27.	9.1.3.2.27.
<i>Padding Method Enumeration</i>	9.1.3.2.14.	9.1.3.2.15.	9.1.3.2.15.
<i>Put Function Enumeration</i>	9.1.3.2.25.	9.1.3.2.26.	9.1.3.2.26.
<i>Query Function Enumeration</i>	9.1.3.2.23.	9.1.3.2.24.	9.1.3.2.24.
<i>Recommended Curve Enumeration for ECDSA, ECDH, and ECMQV</i>	9.1.3.2.5.	9.1.3.2.5.	9.1.3.2.5.
<i>Result Reason Enumeration</i>	9.1.3.2.28.	9.1.3.2.29.	9.1.3.2.29.
<i>Result Status Enumeration</i>	9.1.3.2.27.	9.1.3.2.28.	9.1.3.2.28.
<i>Revocation Reason Code Enumeration</i>	9.1.3.2.18.	9.1.3.2.19.	9.1.3.2.19.
<i>Secret Data Type Enumeration</i>	9.1.3.2.8.	9.1.3.2.9.	9.1.3.2.9.
<i>Split Key Method Enumeration</i>	9.1.3.2.7.	9.1.3.2.8.	9.1.3.2.8.
<i>State Enumeration</i>	9.1.3.2.17.	9.1.3.2.18.	9.1.3.2.18.
<i>Storage Status Mask</i>	9.1.3.3.2.	9.1.3.3.2.	9.1.3.3.2.
<i>Tags</i>	9.1.3.1.	9.1.3.1.	9.1.3.1.
<i>TTLV Encoding</i>	9.1.	9.1.	9.1.
<i>TTLV Encoding Fields</i>	9.1.1.	9.1.1.	9.1.1.
<i>Usage Limits Unit Enumeration</i>	9.1.3.2.30.	9.1.3.2.31.	9.1.3.2.31.

<b>Reference Term</b>	<b>KMIP 1.0</b>	<b>KMIP 1.1</b>	<b>KMIP 1.2</b>
<i>Validity Indicator Enumeration</i>	9.1.3.2.22.	9.1.3.2.23.	9.1.3.2.23.
<i>Wrapping Method Enumeration</i>	9.1.3.2.4.	9.1.3.2.4.	9.1.3.2.4.
<i>XML Encoding</i>	9.2.	-	-
<b>10 Transport</b>			
<i>Transport</i>	10	10	10
<b>12 KMIP Server and Client Implementation Conformance</b>			
<i>Conformance clauses for a KMIP Server</i>	12.1.	-	-
<i>KMIP Client Implementation Conformance</i>	-	12.2.	12.2.
<i>KMIP Server Implementation Conformance</i>	-	12.1.	12.1.

566

---

## Appendix C. Revision History

Revision	Date	Editor	Changes Made
wd01	27-Jun-2013	Tim Hudson / Rod Wideman / Stan Feather	Converted from draft proposal to OASIS template incorporating updates from Stan Feather and inclusion of references to KMIP 1.2 documents. Editorial and formatting cleanup.
wd02	6-August-2013	Tim Hudson	Updated to include Permitted Test Case Variations and updated Test Cases based on July 2013 Interop
wd03	8-August-2013	Stan Feather / Rod Wideman	Editorial changes to section 3
wd04	10-August-2013	Tim Hudson	Updated Permitted Test Case Variations
wd04a	24-October-2013	Tim Hudson	Editorial update to include VendorIdentification in the list of allowed variations as per TC motion.
<a href="#">pr01update</a>	<a href="#">11-June-2014</a>	<a href="#">Tim Hudson</a>	<a href="#">Updated following Public Review</a>