

KMIP Suite B Profile Version 1.0

Committee Specification Draft 02 / Public Review Draft 02

19 June 2014

Specification URIs

This version:

<http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/csprd02/kmip-suite-b-profile-v1.0-csprd02.doc> (Authoritative)
<http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/csprd02/kmip-suite-b-profile-v1.0-csprd02.html>
<http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/csprd02/kmip-suite-b-profile-v1.0-csprd02.pdf>

Previous version:

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

Latest version:

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

Technical Committee:

OASIS Key Management Interoperability Protocol (KMIP) TC

Chairs:

Subhash Sankuratripati (Subhash.Sankuratripati@netapp.com), NetApp
Saikat Saha (saikat.saha@oracle.com), Oracle

Editors:

Kelley Burgin (kwburgi@tycho.ncsc.mil), National Security Agency
Tim Hudson (tjh@cryptsoft.com), Cryptsoft

Related work:

This specification is related to:

- *Key Management Interoperability Protocol Profiles Version 1.0*. Edited by Robert Griffin and Subhash Sankuratripati. 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*. Edited by Robert Haas and Indra Fitzgerald. 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*. Edited by Kiran Thota and Kelley Burgin. Latest version: <http://docs.oasis-open.org/kmip/spec/v1.2/kmip-spec-v1.2.html>.

Abstract:

Describes a profile for KMIP clients and KMIP servers using Suite B cryptography that has been approved by NIST for use by the U.S. Government and specified in NIST standards or recommendations.

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 <https://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 (<https://www.oasis-open.org/committees/kmip/ipr.php>).

Citation format:

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

[kmip-suite-b-v1.0]

KMIP Suite B Profile Version 1.0. Edited by Kelley Burgin and Tim Hudson. 19 June 2014. OASIS Committee Specification Draft 02 / Public Review Draft 02. <http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/csprd02/kmip-suite-b-profile-v1.0-csprd02.html>. Latest version: <http://docs.oasis-open.org/kmip/kmip-suite-b-profile/v1.0/kmip-suite-b-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 <https://www.oasis-open.org/policies-guidelines/trademark> for above guidance.

Table of Contents

1	Introduction.....	6
1.1	Terminology.....	7
1.2	Normative References.....	7
2	Suite B minLOS_128 Profile.....	8
2.1	Authentication Suite.....	8
2.1.1	Protocols.....	8
2.1.2	Cipher Suites.....	8
2.1.3	Client Authenticity.....	8
2.1.4	Object Owner.....	8
2.1.5	KMIP Port Number.....	8
2.2	Suite B minLOS_128 - Client.....	8
2.3	Suite B minLOS_128 - Server.....	9
3	Suite B minLOS_128 Test Cases.....	11
3.1	Mandatory Suite B minLOS_128 Test Cases KMIP 1.0.....	11
3.1.1	SUITEB_128-M-1-10 - Query.....	11
3.2	Mandatory Suite B minLOS_128 Test Cases KMIP 1.1.....	12
3.2.1	SUITEB_128-M-1-11 - Query.....	12
3.3	Mandatory Suite B minLOS_128 Test Cases KMIP 1.2.....	14
3.3.1	SUITEB_128-M-1-12 - Query.....	14
4	Suite B minLOS_192 Profile.....	16
4.1	Authentication Suite.....	16
4.1.1	Protocols.....	16
4.1.2	Cipher Suites.....	16
4.1.3	Client Authenticity.....	16
4.1.4	Object Owner.....	16
4.1.5	KMIP Port Number.....	16
4.2	Suite B minLOS_192 - Client.....	16
4.3	Suite B minLOS_192 - Server.....	17
5	Suite B minLOS_192 Test Cases.....	19
5.1	Mandatory Suite B minLOS_192 Test Cases - KMIP v1.0.....	19
5.1.1	SUITEB_192-M-1-10 - Query.....	19
5.2	Mandatory Suite B minLOS_192 Test Cases KMIP 1.1.....	20
5.2.1	SUITEB_192-M-1-11 - Query.....	20
5.3	Mandatory Suite B minLOS_192 Test Cases KMIP 1.2.....	22
5.3.1	SUITEB_192-M-1-12 - Query.....	22
6	Conformance.....	24
6.1	Suite B minLOS_128 Client KMIP V1.0 Profile Conformance.....	24
6.2	Suite B minLOS_128 Client KMIP V1.1 Profile Conformance.....	24
6.3	Suite B minLOS_128 Client KMIP V1.2 Profile Conformance.....	24
6.4	Suite B minLOS_128 Server KMIP V1.0 Profile Conformance.....	24
6.5	Suite B minLOS_128 Server KMIP V1.1 Profile Conformance.....	24
6.6	Suite B minLOS_128 Server KMIP V1.2 Profile Conformance.....	24
6.7	Suite B minLOS_192 Client KMIP V1.0 Profile Conformance.....	24

6.8 Suite B minLOS_192 Client KMIP V1.1 Profile Conformance.....	25
6.9 Suite B minLOS_192 Client KMIP V1.2 Profile Conformance.....	25
6.10 Suite B minLOS_192 Server KMIP V1.0 Profile Conformance	25
6.11 Suite B minLOS_192 Server KMIP V1.1 Profile Conformance	25
6.12 Suite B minLOS_192 Server KMIP V1.2 Profile Conformance	25
6.13 Permitted Test Case Variations	25
6.13.1 Variable Items	25
6.13.2 Variable behavior	27
Appendix A. Acknowledgments	28
Appendix B. KMIP Specification Cross Reference.....	31
Appendix C. Revision History	36

1 Introduction

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

Suite B [SuiteB] requires that key establishment and signature algorithms be based upon Elliptic Curve Cryptography and that the encryption algorithm be AES [FIPS197]. Suite B includes:

Encryption	Advanced Encryption Standard (AES) (key sizes of 128 and 256 bits)
Digital Signature	Elliptic Curve Digital Signature Algorithm (ECDSA) (using the curves with 256-bit and 384-bit prime moduli)
Key Exchange	Elliptic Curve Diffie-Hellman (ECDH), (using the curves with 256-bit and 384-bit prime moduli)
Hashes	SHA-256 and SHA-384

Suite B provides for two levels of cryptographic security, namely a 128-bit minimum level of security (minLOS_128) and a 192-bit minimum level of security (minLOS_192). Each level defines a minimum strength that all cryptographic algorithms must provide. A KMIP product configured at a minimum level of security of 128 bits provides adequate protection for classified information up to the SECRET level. A KMIP product configured at a minimum level of security of 192 bits is required to protect classified information at the TOP SECRET level.

The Suite B non-signature primitives are divided into two columns as shown below.

	Column 1	Column 2
Encryption	AES-128	AES-256
Key Agreement	ECDH on P-256	ECDH on P-384
Hash for PRF/MAC	SHA-256	SHA-384

At the 128-bit minimum level of security, the non-signature primitives MUST either come exclusively from Column 1 or exclusively from Column 2.

At the 192-bit minimum level of security, the non-signature primitives MUST come exclusively from Column 2.

Digital signatures using ECDSA MUST be used for authentication. Following the direction of RFC 4754, ECDSA-256 represents an instantiation of the ECDSA algorithm using the P-256 curve and the SHA-256 hash function. ECDSA-384 represents an instantiation of the ECDSA algorithm using the P-384 curve and the SHA-384 hash function.

If configured at a minimum level of security of 128 bits, a KMIP product MUST use either ECDSA-256 or ECDSA-384 for authentication. It is allowable for one party to authenticate with ECDSA-256 and the other party to authenticate with ECDSA-384. This flexibility will allow interoperability between a KMIP client and server that have different sizes of ECDSA authentication keys. KMIP products configured at a minimum level of security of 128 bits MUST be able to verify ECDSA-256 signatures and SHOULD be able to verify ECDSA-384 signatures. If configured at a minimum level of security of 192 bits, ECDSA-384 MUST be used by both the KMIP client and server for authentication. KMIP products configured at a minimum level of security of 192 bits MUST be able to verify ECDSA-384 signatures.

32 KMIP products, at both minimum levels of security, MUST each use an X.509 certificate that complies
33 with the "Suite B Certificate and Certificate Revocation List (CRL) Profile" [RFC5759] and that contains an
34 elliptic curve public key with the key usage bit set for digital signature.

35 1.1 Terminology

36 The key words "MUST", "SHALL", "SHOULD", and "MAY" in this document are to be interpreted as
37 described in [RFC2119].

38 1.2 Normative References

- 39 **[CNSSP-15]** N.S.A., "National Information Assurance Policy on the Use of Public Standards
40 for the Secure Sharing of Information Among National Security Systems", 1
41 October 2013,
42 [https://www.cnss.gov/Assets/pdf/CNSSP_No%2015_minorUpdate1_Oct12012.p](https://www.cnss.gov/Assets/pdf/CNSSP_No%2015_minorUpdate1_Oct12012.pdf)
43 [df](https://www.cnss.gov/Assets/pdf/CNSSP_No%2015_minorUpdate1_Oct12012.pdf).
- 44 **[RFC2119]** Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP
45 14, RFC 2119, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>.
- 46 **[KMIP-ENCODE]** *KMIP Additional Message Encodings Version 1.0*.
47 [URL](#)
48 Candidate OASIS Standard 01. **DD MMM YYYY**.
- 49 **[RFC5246]** Dierks, T. and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version*
50 *1.2*, IETF RFC 5246, August 2008, <http://www.ietf.org/rfc/rfc5246.txt>.
- 51 **[KMIP-SPEC]** One or more of [KMIP-SPEC-1_0], [KMIP-SPEC-1_1], [KMIP-SPEC-1_2]
- 52 **[KMIP-SPEC-1_0]** *Key Management Interoperability Protocol Specification Version 1.0*,
53 <http://docs.oasis-open.org/kmip/spec/v1.0/os/kmip-spec-1.0-os.doc>,
54 OASIS Standard, 1 October 2010.
- 55 **[KMIP-SPEC-1_1]** *Key Management Interoperability Protocol Specification Version 1.1*,
56 <http://docs.oasis-open.org/kmip/spec/v1.1/os/kmip-spec-v1.1-os.doc>,
57 OASIS Standard, 24 January 2013.
- 58 **[KMIP-SPEC-1_2]** *Key Management Interoperability Protocol Specification Version 1.2*,
59 [URL](#), Candidate OASIS Standard 01, **DD MMM YYYY**.
- 60 **[KMIP-PROF]** One or more of [KMIP-PROF-1_0], [KMIP-PROF-1_1], [KMIP-PROF-1_2]
- 61 **[KMIP-PROF-1_0]** *Key Management Interoperability Protocol Profiles Version 1.0*, [http://docs.oasis-](http://docs.oasis-open.org/kmip/profiles/v1.0/os/kmip-profiles-1.0-os.doc)
62 [open.org/kmip/profiles/v1.0/os/kmip-profiles-1.0-os.doc](http://docs.oasis-open.org/kmip/profiles/v1.0/os/kmip-profiles-1.0-os.doc),
63 OASIS Standard, 1 October 2010.
- 64 **[KMIP-PROF-1_1]** *Key Management Interoperability Protocol Profiles Version 1.1*,
65 <http://docs.oasis-open.org/kmip/profiles/v1.1/os/kmip-profiles-v1.1-os.doc>,
66 OASIS Standard 01, 24 January 2013.
- 67 **[KMIP-PROF-1_2]** *Key Management Interoperability Protocol Profiles Version 1.2*,
68 [URL](#), Candidate OASIS Standard 01, **DD MMM YYYY**.
- 69 **[SuiteB]** *Suite B Cryptography / Cryptographic Interoperability*,
70 http://www.nsa.gov/ia/programs/suiteb_cryptography/
71

72 2 Suite B minLOS_128 Profile

73 The Suite B minLOS_128 Profile describes a KMIP client interacting with a KMIP server as an information
74 assurance product to provide a minimum level of security of 128 bits.
75 (http://www.nsa.gov/ia/programs/suiteb_cryptography/)

76 2.1 Authentication Suite

77 Implementations conformant to this profile SHALL use TLS to negotiate a mutually-authenticated
78 connection.

79 2.1.1 Protocols

80 Conformant KMIP clients and servers SHALL support:

- 81 • TLS v1.2 [RFC5246]

82 2.1.2 Cipher Suites

83 Conformant KMIP servers SHALL support the following cipher suites:

- 84 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

85 2.1.3 Client Authenticity

86 Conformant KMIP servers and clients SHALL handle client authenticity in accordance with section 3.2.3
87 of the TLS 1.2 Authentication Suite [KMIP-PROF].

88 2.1.4 Object Owner

89 Conformant KMIP servers and clients SHALL handle object owner in accordance with section 3.2.4 of the
90 TLS 1.2 Authentication Suite [KMIP-PROF].

91 2.1.5 KMIP Port Number

92 Conformant KMIP servers and clients SHALL handle the KMIP port number in accordance with section
93 3.2.5 of the TLS 1.2 Authentication Suite [KMIP-PROF].

94 2.2 Suite B minLOS_128 - Client

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

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

97 KMIP clients conformant to this profile under [KMIP-SPEC-1_1]:

- 98 2. SHALL conform to the *Baseline Client Clause* (section 5.12) of [KMIP-PROF-1_1]

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

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

101 KMIP clients conformant to this profile:

- 102 4. SHALL restrict use of the enumerated types listed in item 8 of the server list in section 2.3 to the
103 values noted against each item
- 104 5. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause
105 within this section 2.2.
- 106 6. MAY support extensions outside the scope of this standard (e.g., vendor extensions,
107 conformance clauses) that do not conflict with any KMIP or [CNSSP-15] requirements.

108 2.3 Suite B minLOS_128 - Server

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

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

111 KMIP servers conformant to this profile under [KMIP-SPEC-1_1]:

112 2. SHALL conform to the *Baseline Server* of [KMIP-PROF-1_1]

113 KMIP servers conformant to this profile under [KMIP-SPEC-1_2]:

114 3. SHALL conform to the *Baseline Server* of [KMIP-PROF-1_2]

115 KMIP servers conformant to this profile:

116 4. SHALL support the following *Objects* [KMIP-SPEC]

117 a. *Certificate* [KMIP-SPEC]

118 b. *Symmetric Key* [KMIP-SPEC]

119 c. *Public Key* [KMIP-SPEC]

120 d. *Private Key* [KMIP-SPEC]

121 5. SHALL support the following *Attributes* [KMIP-SPEC]

122 a. *Cryptographic Algorithm* [KMIP-SPEC]

123 b. *Cryptographic Length* [KMIP-SPEC] value :

124 i. 128-bit (combined with AES)

125 ii. 256-bit (combined with SHA, ECDH or ECDSA)

126 6. MAY support the following *Attributes* [KMIP-SPEC]

127 a. *Cryptographic Length* [KMIP-SPEC] value :

128 i. 256-bit (combined with AES)

129 ii. 384-bit bit (combined with SHA, ECDH or ECDSA)

130 7. SHALL support the following *Client-to-Server Operations* [KMIP-SPEC]:

131 a. *Create* [KMIP-SPEC]

132 b. *Create Key Pair* [KMIP-SPEC]

133 c. *Register* [KMIP-SPEC]

134 d. *Re-key* [KMIP-SPEC]

135 e. *Re-key Key Pair* [KMIP-SPEC]

136 8. SHALL support the following *Message Encoding* [KMIP-SPEC]:

137 a. *Recommended Curve Enumeration* [KMIP-SPEC] value:

138 i. P-256 (SECP256R1)

139 b. *Certificate Type Enumeration* [KMIP-SPEC] value:

140 i. X.509

141 c. *Cryptographic Algorithm Enumeration* [KMIP-SPEC] value:

142 i. AES

143 ii. ECDSA

144 iii. ECDH

145 iv. HMAC-SHA256

146 d. *Hashing Algorithm Enumeration* [KMIP-SPEC]

147 i. SHA-256

148 e. *Object Type Enumeration* [KMIP-SPEC] value:

149 i. Certificate

- 150 ii. Symmetric Key
151 iii. Public Key
152 iv. Private Key
153 f. *Key Format Type Enumeration* [KMIP-SPEC] value:
154 i. Raw
155 ii. ECPrivateKey
156 iii. X.509
157 iv. Transparent ECDSA Private Key
158 v. Transparent ECDSA Public Key
159 vi. Transparent ECDH Private Key
160 vii. Transparent ECDH Public Key
161 g. *Digital Signature Algorithm Enumeration* [KMIP-SPEC] value:
162 i. ECDSA with SHA256 (on P-256)
163 9. MAY support the following *Message Encoding* [KMIP-SPEC]:
164 a. *Recommended Curve* [KMIP-SPEC] value:
165 i. P-384 (SECP384R1)
166 b. *Cryptographic Algorithm Enumeration* [KMIP-SPEC] value:
167 i. HMAC-SHA384
168 c. *Hashing Algorithm Enumeration* [KMIP-SPEC]
169 i. SHA-384
170 d. *Digital Signature Algorithm Enumeration*
171 i. ECDSA with SHA384 (on P-384)
172 10. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause
173 within this section 2.3.
174 11. MAY support extensions outside the scope of this standard (e.g., vendor extensions,
175 conformance clauses) that do not conflict with any KMIP or [CNSSP-15] requirements.

176 3 Suite B minLOS_128 Test Cases

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

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

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

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

188 Note: the values for the returned items and the custom attributes are illustrative. Actual values from a real
189 client or server system may vary as specified in section 6.10

190 3.1 Mandatory Suite B minLOS_128 Test Cases KMIP 1.0

191 3.1.1 SUITEB_128-M-1-10 - Query

192 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
193 successful response.

194 The specific list of operations and object types returned in the response MAY vary.

195 The TLS protocol version and cipher suite SHALL be as specified in section 2.1

```
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     <RequestPayload>
0013       <QueryFunction type="Enumeration" value="QueryOperations"/>
0014       <QueryFunction type="Enumeration" value="QueryObjects"/>
0015     </RequestPayload>
0016   </BatchItem>
0017 </RequestMessage>
0018 <ResponseMessage>
0019   <ResponseHeader>
0020     <ProtocolVersion>
0021       <ProtocolVersionMajor type="Integer" value="1"/>
0022       <ProtocolVersionMinor type="Integer" value="0"/>
0023     </ProtocolVersion>
0024     <TimeStamp type="DateTime" value="2013-06-26T09:09:17+00:00"/>
0025     <BatchCount type="Integer" value="1"/>
0026   </ResponseHeader>
0027   <BatchItem>
0028     <Operation type="Enumeration" value="Query"/>
```

```

0028 <ResultStatus type="Enumeration" value="Success"/>
0029 <ResponsePayload>
0030 <Operation type="Enumeration" value="Query"/>
0031 <Operation type="Enumeration" value="Locate"/>
0032 <Operation type="Enumeration" value="Destroy"/>
0033 <Operation type="Enumeration" value="Get"/>
0034 <Operation type="Enumeration" value="Create"/>
0035 <Operation type="Enumeration" value="Register"/>
0036 <Operation type="Enumeration" value="GetAttributes"/>
0037 <Operation type="Enumeration" value="GetAttributeList"/>
0038 <Operation type="Enumeration" value="AddAttribute"/>
0039 <Operation type="Enumeration" value="ModifyAttribute"/>
0040 <Operation type="Enumeration" value="DeleteAttribute"/>
0041 <Operation type="Enumeration" value="Activate"/>
0042 <Operation type="Enumeration" value="Revoke"/>
0043 <Operation type="Enumeration" value="Poll"/>
0044 <Operation type="Enumeration" value="Cancel"/>
0045 <Operation type="Enumeration" value="Check"/>
0046 <Operation type="Enumeration" value="GetUsageAllocation"/>
0047 <Operation type="Enumeration" value="CreateKeyPair"/>
0048 <Operation type="Enumeration" value="ReKey"/>
0049 <Operation type="Enumeration" value="Archive"/>
0050 <Operation type="Enumeration" value="Recover"/>
0051 <Operation type="Enumeration" value="ObtainLease"/>
0052 <Operation type="Enumeration" value="Certify"/>
0053 <Operation type="Enumeration" value="ReCertify"/>
0054 <Operation type="Enumeration" value="Notify"/>
0055 <Operation type="Enumeration" value="Put"/>
0056 <ObjectType type="Enumeration" value="Certificate"/>
0057 <ObjectType type="Enumeration" value="SymmetricKey"/>
0058 <ObjectType type="Enumeration" value="SecretData"/>
0059 <ObjectType type="Enumeration" value="PublicKey"/>
0060 <ObjectType type="Enumeration" value="PrivateKey"/>
0061 <ObjectType type="Enumeration" value="Template"/>
0062 <ObjectType type="Enumeration" value="OpaqueObject"/>
0063 <ObjectType type="Enumeration" value="SplitKey"/>
0064 </ResponsePayload>
0065 </BatchItem>
0066 </ResponseMessage>

```

196

197 3.2 Mandatory Suite B minLOS_128 Test Cases KMIP 1.1

198 3.2.1 SUITEB_128-M-1-11 - Query

199 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
200 successful response.

201 The specific list of operations and object types returned in the response MAY vary.

202 The TLS protocol version and cipher suite SHALL be as specified in section 2.1

```

0001 # TIME 0
0002 <RequestMessage>
0003 <RequestHeader>
0004 <ProtocolVersion>
0005 <ProtocolVersionMajor type="Integer" value="1"/>
0006 <ProtocolVersionMinor type="Integer" value="1"/>

```

0007	<BatchCount type="Integer" value="1"/>
0008	</RequestHeader>
0009	<BatchItem>
0010	<Operation type="Enumeration" value="Query"/>
0011	<RequestPayload>
0012	<QueryFunction type="Enumeration" value="QueryOperations"/>
0013	<QueryFunction type="Enumeration" value="QueryObjects"/>
0014	</RequestPayload>
0015	</BatchItem>
0016	</RequestMessage>
0017	<ResponseMessage>
0018	<ResponseHeader>
0019	<ProtocolVersion>
0020	<ProtocolVersionMajor type="Integer" value="1"/>
0021	<ProtocolVersionMinor type="Integer" value="1"/>
0022	</ProtocolVersion>
0023	<TimeStamp type="DateTime" value="2014-06-11T09:22:39+00:00"/>
0024	<BatchCount type="Integer" value="1"/>
0025	</ResponseHeader>
0026	<BatchItem>
0027	<Operation type="Enumeration" value="Query"/>
0028	<ResultStatus type="Enumeration" value="Success"/>
0029	<ResponsePayload>
0030	<Operation type="Enumeration" value="Query"/>
0031	<Operation type="Enumeration" value="Locate"/>
0032	<Operation type="Enumeration" value="Destroy"/>
0033	<Operation type="Enumeration" value="Get"/>
0034	<Operation type="Enumeration" value="Create"/>
0035	<Operation type="Enumeration" value="Register"/>
0036	<Operation type="Enumeration" value="GetAttributes"/>
0037	<Operation type="Enumeration" value="GetAttributeList"/>
0038	<Operation type="Enumeration" value="AddAttribute"/>
0039	<Operation type="Enumeration" value="ModifyAttribute"/>
0040	<Operation type="Enumeration" value="DeleteAttribute"/>
0041	<Operation type="Enumeration" value="Activate"/>
0042	<Operation type="Enumeration" value="Revoke"/>
0043	<Operation type="Enumeration" value="Poll"/>
0044	<Operation type="Enumeration" value="Cancel"/>
0045	<Operation type="Enumeration" value="Check"/>
0046	<Operation type="Enumeration" value="GetUsageAllocation"/>
0047	<Operation type="Enumeration" value="CreateKeyPair"/>
0048	<Operation type="Enumeration" value="ReKey"/>
0049	<Operation type="Enumeration" value="Archive"/>
0050	<Operation type="Enumeration" value="Recover"/>
0051	<Operation type="Enumeration" value="ObtainLease"/>
0052	<Operation type="Enumeration" value="ReKeyKeyPair"/>
0053	<Operation type="Enumeration" value="Certify"/>
0054	<Operation type="Enumeration" value="ReCertify"/>
0055	<Operation type="Enumeration" value="DiscoverVersions"/>
0056	<Operation type="Enumeration" value="Notify"/>
0057	<Operation type="Enumeration" value="Put"/>
0058	<ObjectType type="Enumeration" value="Certificate"/>
0059	<ObjectType type="Enumeration" value="SymmetricKey"/>
0060	<ObjectType type="Enumeration" value="SecretData"/>
0061	<ObjectType type="Enumeration" value="PublicKey"/>
0062	<ObjectType type="Enumeration" value="PrivateKey"/>
0063	<ObjectType type="Enumeration" value="Template"/>
0064	<ObjectType type="Enumeration" value="OpaqueObject"/>

0065	<ObjectType type="Enumeration" value="SplitKey"/>
0066	</ResponsePayload>
0067	</BatchItem>
0068	</ResponseMessage>

203

204 3.3 Mandatory Suite B minLOS_128 Test Cases KMIP 1.2

205 3.3.1 SUITEB_128-M-1-12 - Query

206 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
207 successful response.

208 The specific list of operations and object types returned in the response MAY vary.

209 The TLS protocol version and cipher suite SHALL be as specified in section 2.1

	# 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	<RequestPayload>
0012	<QueryFunction type="Enumeration" value="QueryOperations"/>
0013	<QueryFunction type="Enumeration" value="QueryObjects"/>
0014	</RequestPayload>
0015	</BatchItem>
0016	</RequestMessage>
0017	<ResponseMessage>
0018	<ResponseHeader>
0019	<ProtocolVersion>
0020	<ProtocolVersionMajor type="Integer" value="1"/>
0021	<ProtocolVersionMinor type="Integer" value="2"/>
0022	</ProtocolVersion>
0023	<TimeStamp type="DateTime" value="2014-06-11T09:23:21+00:00"/>
0024	<BatchCount type="Integer" value="1"/>
0025	</ResponseHeader>
0026	<BatchItem>
0027	<Operation type="Enumeration" value="Query"/>
0028	<ResultStatus type="Enumeration" value="Success"/>
0029	<ResponsePayload>
0030	<Operation type="Enumeration" value="Query"/>
0031	<Operation type="Enumeration" value="Locate"/>
0032	<Operation type="Enumeration" value="Destroy"/>
0033	<Operation type="Enumeration" value="Get"/>
0034	<Operation type="Enumeration" value="Create"/>
0035	<Operation type="Enumeration" value="Register"/>
0036	<Operation type="Enumeration" value="GetAttributes"/>
0037	<Operation type="Enumeration" value="GetAttributeList"/>
0038	<Operation type="Enumeration" value="AddAttribute"/>
0039	<Operation type="Enumeration" value="ModifyAttribute"/>
0040	<Operation type="Enumeration" value="DeleteAttribute"/>
0041	<Operation type="Enumeration" value="Activate"/>

```
0042 <Operation type="Enumeration" value="Revoke"/>
0043 <Operation type="Enumeration" value="Poll"/>
0044 <Operation type="Enumeration" value="Cancel"/>
0045 <Operation type="Enumeration" value="Check"/>
0046 <Operation type="Enumeration" value="GetUsageAllocation"/>
0047 <Operation type="Enumeration" value="CreateKeyPair"/>
0048 <Operation type="Enumeration" value="ReKey"/>
0049 <Operation type="Enumeration" value="Archive"/>
0050 <Operation type="Enumeration" value="Recover"/>
0051 <Operation type="Enumeration" value="ObtainLease"/>
0052 <Operation type="Enumeration" value="ReKeyKeyPair"/>
0053 <Operation type="Enumeration" value="Certify"/>
0054 <Operation type="Enumeration" value="ReCertify"/>
0055 <Operation type="Enumeration" value="DiscoverVersions"/>
0056 <Operation type="Enumeration" value="Notify"/>
0057 <Operation type="Enumeration" value="Put"/>
0058 <Operation type="Enumeration" value="RNGRetrieve"/>
0059 <Operation type="Enumeration" value="RNGSeed"/>
0060 <Operation type="Enumeration" value="Encrypt"/>
0061 <Operation type="Enumeration" value="Decrypt"/>
0062 <Operation type="Enumeration" value="Sign"/>
0063 <Operation type="Enumeration" value="SignatureVerify"/>
0064 <Operation type="Enumeration" value="MAC"/>
0065 <Operation type="Enumeration" value="MACVerify"/>
0066 <Operation type="Enumeration" value="Hash"/>
0067 <Operation type="Enumeration" value="CreateSplitKey"/>
0068 <Operation type="Enumeration" value="JoinSplitKey"/>
0069 <ObjectType type="Enumeration" value="Certificate"/>
0070 <ObjectType type="Enumeration" value="SymmetricKey"/>
0071 <ObjectType type="Enumeration" value="SecretData"/>
0072 <ObjectType type="Enumeration" value="PublicKey"/>
0073 <ObjectType type="Enumeration" value="PrivateKey"/>
0074 <ObjectType type="Enumeration" value="Template"/>
0075 <ObjectType type="Enumeration" value="OpaqueObject"/>
0076 <ObjectType type="Enumeration" value="SplitKey"/>
0077 <ObjectType type="Enumeration" value="PGPKey"/>
0078 </ResponsePayload>
0079 </BatchItem>
0080 </ResponseMessage>
```

210

211 4 Suite B minLOS_192 Profile

212 The Suite B minLOS_192 Profile describes a KMIP client interacting with a KMIP server as an information
213 assurance product to provide a minimum level of security of 192 bits.
214 (http://www.nsa.gov/ia/programs/suiteb_cryptography/)

215 4.1 Authentication Suite

216 Implementations conformant to this profile SHALL use TLS to negotiate a mutually-authenticated
217 connection.

218 4.1.1 Protocols

219 Conformant KMIP clients and servers SHALL support:

- 220 • TLS v1.2 [RFC5246]

221 4.1.2 Cipher Suites

222 Conformant KMIP servers SHALL support the following cipher suites:

- 223 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

224 4.1.3 Client Authenticity

225 Conformant KMIP servers and clients SHALL handle client authenticity in accordance with section 3.2.3
226 of the TLS 1.2 Authentication Suite [KMIP-PROF].

227 4.1.4 Object Owner

228 Conformant KMIP servers and clients SHALL handle object owner in accordance with section 3.2.4 of the
229 TLS 1.2 Authentication Suite [KMIP-PROF].

230 4.1.5 KMIP Port Number

231 Conformant KMIP servers and clients SHALL handle the KMIP port number in accordance with section
232 3.2.5 of the TLS 1.2 Authentication Suite [KMIP-PROF].

233 4.2 Suite B minLOS_192 - Client

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

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

236 KMIP clients conformant to this profile under [KMIP-SPEC-1_1]:

- 237 2. SHALL conform to the *Baseline Client Clause* (section 5.12) of [KMIP-PROF-1_1]

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

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

240 KMIP clients conformant to this profile under [KMIP-SPEC]:

- 241 4. SHALL restrict use of the enumerated types listed in item 7 of the server list in section 4.3 to the
242 values noted against each item
- 243 5. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause
244 within this section 4.2.
- 245 6. MAY support extensions outside the scope of this standard (e.g., vendor extensions,
246 conformance clauses) that do not conflict with any KMIP or [CNSSP-15] requirements.

247 4.3 Suite B minLOS_192 - Server

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

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

250 KMIP servers conformant to this profile under [KMIP-SPEC-1_1]:

251 2. SHALL conform to the *Baseline Server* of [KMIP-PROF-1_1]

252 KMIP servers conformant to this profile under [KMIP-SPEC-1_2]:

253 3. SHALL conform to the *Baseline Server* of [KMIP-PROF-1_2]

254 KMIP servers conformant to this profile under [KMIP-SPEC]:

255 4. SHALL support the following *Objects* [KMIP-SPEC]

256 a. *Certificate* [KMIP-SPEC]

257 b. *Symmetric Key* [KMIP-SPEC]

258 c. *Public Key* [KMIP-SPEC]

259 d. *Private Key* [KMIP-SPEC]

260 5. SHALL support the following *Attributes* [KMIP-SPEC]

261 e. *Cryptographic Algorithm* [KMIP-SPEC]

262 f. *Cryptographic Length* [KMIP-SPEC] value:

263 i. 384-bit bit (combined with SHA, ECDH or ECDSA)

264 6. SHALL support the following *Client-to-Server Operations* [KMIP-SPEC]:

265 g. *Create* [KMIP-SPEC]

266 h. *Create Key Pair* [KMIP-SPEC]

267 i. *Register* [KMIP-SPEC]

268 j. *Re-key* [KMIP-SPEC]

269 k. *Re-key Key Pair* [KMIP-SPEC]

270 7. SHALL support the following *Message Encoding* [KMIP-SPEC]:

271 l. *Recommended Curve Enumeration* [KMIP-SPEC] value:

272 i. P-384 (SECP384R1)

273 m. *Certificate Type Enumeration* [KMIP-SPEC] value:

274 i. X.509

275 n. *Cryptographic Algorithm Enumeration* [KMIP-SPEC] value:

276 i. AES

277 ii. ECDSA

278 iii. ECDH

279 iv. HMAC-SHA384

280 o. *Hashing Algorithm Enumeration* [KMIP-SPEC]

281 i. SHA-384

282 p. *Object Type Enumeration* [KMIP-SPEC] value:

283 i. Certificate

284 ii. Symmetric Key

285 iii. Public Key

286 iv. Private Key

287 q. *Key Format Type Enumeration* [KMIP-SPEC] value:

288 i. Raw

- 289 ii. ECPrivateKey
- 290 iii. X.509
- 291 iv. Transparent ECDSA Private Key
- 292 v. Transparent ECDSA Public Key
- 293 vi. Transparent ECDH Private Key
- 294 vii. Transparent ECDH Public Key
- 295 r. *Digital Signature Algorithm Enumeration* [KMIP-SPEC] value:
- 296 i. ECDSA with SHA384 (on P-384)
- 297 8. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause
- 298 within this section 4.3.
- 299 9. MAY support extensions outside the scope of this standard (e.g., vendor extensions,
- 300 conformance clauses) that do not conflict with any KMIP or [CNSSP-15] requirements.

301 5 Suite B minLOS_192 Test Cases

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

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

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

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

313 Note: the values for the returned items and the custom attributes are illustrative. Actual values from a real
314 client or server system may vary as specified in section 6.10

315 5.1 Mandatory Suite B minLOS_192 Test Cases - KMIP v1.0

316 This section documents the test cases that a client or server conformant to this profile SHALL support.

317 5.1.1 SUITEB_192-M-1-10 - Query

318 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
319 successful response.

320 The specific list of operations and object types returned in the response MAY vary.

321 The TLS protocol version and cipher suite SHALL be as specified in section 4.1

0001	<code># TIME 0</code>
0002	<code><RequestMessage></code>
0003	<code> <RequestHeader></code>
0004	<code> <ProtocolVersion></code>
0005	<code> <ProtocolVersionMajor type="Integer" value="1"/></code>
0006	<code> <ProtocolVersionMinor type="Integer" value="0"/></code>
0007	<code> </ProtocolVersion></code>
0008	<code> <BatchCount type="Integer" value="1"/></code>
0009	<code> </RequestHeader></code>
0010	<code> <BatchItem></code>
0011	<code> <Operation type="Enumeration" value="Query"/></code>
0012	<code> <RequestPayload></code>
0013	<code> <QueryFunction type="Enumeration" value="QueryOperations"/></code>
0014	<code> <QueryFunction type="Enumeration" value="QueryObjects"/></code>
0015	<code> </RequestPayload></code>
0016	<code> </BatchItem></code>
0017	<code></RequestMessage></code>
0018	<code><ResponseMessage></code>
0019	<code> <ResponseHeader></code>
0020	<code> <ProtocolVersion></code>
0021	<code> <ProtocolVersionMajor type="Integer" value="1"/></code>
0022	<code> <ProtocolVersionMinor type="Integer" value="0"/></code>
0023	<code> </ProtocolVersion></code>
0024	<code> <TimeStamp type="DateTime" value="2013-06-26T09:09:17+00:00"/></code>
0025	<code> <BatchCount type="Integer" value="1"/></code>
	<code> </ResponseHeader></code>

```

0026 <BatchItem>
0027   <Operation type="Enumeration" value="Query"/>
0028   <ResultStatus type="Enumeration" value="Success"/>
0029   <ResponsePayload>
0030     <Operation type="Enumeration" value="Query"/>
0031     <Operation type="Enumeration" value="Locate"/>
0032     <Operation type="Enumeration" value="Destroy"/>
0033     <Operation type="Enumeration" value="Get"/>
0034     <Operation type="Enumeration" value="Create"/>
0035     <Operation type="Enumeration" value="Register"/>
0036     <Operation type="Enumeration" value="GetAttributes"/>
0037     <Operation type="Enumeration" value="GetAttributeList"/>
0038     <Operation type="Enumeration" value="AddAttribute"/>
0039     <Operation type="Enumeration" value="ModifyAttribute"/>
0040     <Operation type="Enumeration" value="DeleteAttribute"/>
0041     <Operation type="Enumeration" value="Activate"/>
0042     <Operation type="Enumeration" value="Revoke"/>
0043     <Operation type="Enumeration" value="Poll"/>
0044     <Operation type="Enumeration" value="Cancel"/>
0045     <Operation type="Enumeration" value="Check"/>
0046     <Operation type="Enumeration" value="GetUsageAllocation"/>
0047     <Operation type="Enumeration" value="CreateKeyPair"/>
0048     <Operation type="Enumeration" value="ReKey"/>
0049     <Operation type="Enumeration" value="Archive"/>
0050     <Operation type="Enumeration" value="Recover"/>
0051     <Operation type="Enumeration" value="ObtainLease"/>
0052     <Operation type="Enumeration" value="Certify"/>
0053     <Operation type="Enumeration" value="ReCertify"/>
0054     <Operation type="Enumeration" value="Notify"/>
0055     <Operation type="Enumeration" value="Put"/>
0056     <ObjectType type="Enumeration" value="Certificate"/>
0057     <ObjectType type="Enumeration" value="SymmetricKey"/>
0058     <ObjectType type="Enumeration" value="SecretData"/>
0059     <ObjectType type="Enumeration" value="PublicKey"/>
0060     <ObjectType type="Enumeration" value="PrivateKey"/>
0061     <ObjectType type="Enumeration" value="Template"/>
0062     <ObjectType type="Enumeration" value="OpaqueObject"/>
0063     <ObjectType type="Enumeration" value="SplitKey"/>
0064   </ResponsePayload>
0065 </BatchItem>
0066 </ResponseMessage>

```

322

323 5.2 Mandatory Suite B minLOS_192 Test Cases KMIP 1.1

324 5.2.1 SUITEB_192-M-1-11 - Query

325 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
326 successful response.

327 The specific list of operations and object types returned in the response MAY vary.

328 The TLS protocol version and cipher suite SHALL be as specified in section 4.1

```

# 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	<RequestPayload>
0012	<QueryFunction type="Enumeration" value="QueryOperations"/>
0013	<QueryFunction type="Enumeration" value="QueryObjects"/>
0014	</RequestPayload>
0015	</BatchItem>
0016	</RequestMessage>
0017	<ResponseMessage>
0018	<ResponseHeader>
0019	<ProtocolVersion>
0020	<ProtocolVersionMajor type="Integer" value="1"/>
0021	<ProtocolVersionMinor type="Integer" value="1"/>
0022	</ProtocolVersion>
0023	<TimeStamp type="DateTime" value="2014-06-11T09:22:39+00:00"/>
0024	<BatchCount type="Integer" value="1"/>
0025	</ResponseHeader>
0026	<BatchItem>
0027	<Operation type="Enumeration" value="Query"/>
0028	<ResultStatus type="Enumeration" value="Success"/>
0029	<ResponsePayload>
0030	<Operation type="Enumeration" value="Query"/>
0031	<Operation type="Enumeration" value="Locate"/>
0032	<Operation type="Enumeration" value="Destroy"/>
0033	<Operation type="Enumeration" value="Get"/>
0034	<Operation type="Enumeration" value="Create"/>
0035	<Operation type="Enumeration" value="Register"/>
0036	<Operation type="Enumeration" value="GetAttributes"/>
0037	<Operation type="Enumeration" value="GetAttributeList"/>
0038	<Operation type="Enumeration" value="AddAttribute"/>
0039	<Operation type="Enumeration" value="ModifyAttribute"/>
0040	<Operation type="Enumeration" value="DeleteAttribute"/>
0041	<Operation type="Enumeration" value="Activate"/>
0042	<Operation type="Enumeration" value="Revoke"/>
0043	<Operation type="Enumeration" value="Poll"/>
0044	<Operation type="Enumeration" value="Cancel"/>
0045	<Operation type="Enumeration" value="Check"/>
0046	<Operation type="Enumeration" value="GetUsageAllocation"/>
0047	<Operation type="Enumeration" value="CreateKeyPair"/>
0048	<Operation type="Enumeration" value="ReKey"/>
0049	<Operation type="Enumeration" value="Archive"/>
0050	<Operation type="Enumeration" value="Recover"/>
0051	<Operation type="Enumeration" value="ObtainLease"/>
0052	<Operation type="Enumeration" value="ReKeyKeyPair"/>
0053	<Operation type="Enumeration" value="Certify"/>
0054	<Operation type="Enumeration" value="ReCertify"/>
0055	<Operation type="Enumeration" value="DiscoverVersions"/>
0056	<Operation type="Enumeration" value="Notify"/>
0057	<Operation type="Enumeration" value="Put"/>
0058	<ObjectType type="Enumeration" value="Certificate"/>
0059	<ObjectType type="Enumeration" value="SymmetricKey"/>
0060	<ObjectType type="Enumeration" value="SecretData"/>
0061	<ObjectType type="Enumeration" value="PublicKey"/>
0062	<ObjectType type="Enumeration" value="PrivateKey"/>

0063	<ObjectType type="Enumeration" value="Template"/>
0064	<ObjectType type="Enumeration" value="OpaqueObject"/>
0065	<ObjectType type="Enumeration" value="SplitKey"/>
0066	</ResponsePayload>
0067	</BatchItem>
0068	</ResponseMessage>

329

330 5.3 Mandatory Suite B minLOS_192 Test Cases KMIP 1.2

331 5.3.1 SUITEB_192-M-1-12 - Query

332 Perform a Query operation, querying the Operations and Objects supported by the server, and get a
 333 successful response.

334 The specific list of operations and object types returned in the response MAY vary.

335 The TLS protocol version and cipher suite SHALL be as specified in section 4.1

	# 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	<RequestPayload>
0012	<QueryFunction type="Enumeration" value="QueryOperations"/>
0013	<QueryFunction type="Enumeration" value="QueryObjects"/>
0014	</RequestPayload>
0015	</BatchItem>
0016	</RequestMessage>
0017	<ResponseMessage>
0018	<ResponseHeader>
0019	<ProtocolVersion>
0020	<ProtocolVersionMajor type="Integer" value="1"/>
0021	<ProtocolVersionMinor type="Integer" value="2"/>
0022	</ProtocolVersion>
0023	<TimeStamp type="DateTime" value="2014-06-11T09:23:21+00:00"/>
0024	<BatchCount type="Integer" value="1"/>
0025	</ResponseHeader>
0026	<BatchItem>
0027	<Operation type="Enumeration" value="Query"/>
0028	<ResultStatus type="Enumeration" value="Success"/>
0029	<ResponsePayload>
0030	<Operation type="Enumeration" value="Query"/>
0031	<Operation type="Enumeration" value="Locate"/>
0032	<Operation type="Enumeration" value="Destroy"/>
0033	<Operation type="Enumeration" value="Get"/>
0034	<Operation type="Enumeration" value="Create"/>
0035	<Operation type="Enumeration" value="Register"/>
0036	<Operation type="Enumeration" value="GetAttributes"/>
0037	<Operation type="Enumeration" value="GetAttributeList"/>
0038	<Operation type="Enumeration" value="AddAttribute"/>
0039	<Operation type="Enumeration" value="ModifyAttribute"/>

```

0040     <Operation type="Enumeration" value="DeleteAttribute"/>
0041     <Operation type="Enumeration" value="Activate"/>
0042     <Operation type="Enumeration" value="Revoke"/>
0043     <Operation type="Enumeration" value="Poll"/>
0044     <Operation type="Enumeration" value="Cancel"/>
0045     <Operation type="Enumeration" value="Check"/>
0046     <Operation type="Enumeration" value="GetUsageAllocation"/>
0047     <Operation type="Enumeration" value="CreateKeyPair"/>
0048     <Operation type="Enumeration" value="ReKey"/>
0049     <Operation type="Enumeration" value="Archive"/>
0050     <Operation type="Enumeration" value="Recover"/>
0051     <Operation type="Enumeration" value="ObtainLease"/>
0052     <Operation type="Enumeration" value="ReKeyKeyPair"/>
0053     <Operation type="Enumeration" value="Certify"/>
0054     <Operation type="Enumeration" value="ReCertify"/>
0055     <Operation type="Enumeration" value="DiscoverVersions"/>
0056     <Operation type="Enumeration" value="Notify"/>
0057     <Operation type="Enumeration" value="Put"/>
0058     <Operation type="Enumeration" value="RNGRetrieve"/>
0059     <Operation type="Enumeration" value="RNGSeed"/>
0060     <Operation type="Enumeration" value="Encrypt"/>
0061     <Operation type="Enumeration" value="Decrypt"/>
0062     <Operation type="Enumeration" value="Sign"/>
0063     <Operation type="Enumeration" value="SignatureVerify"/>
0064     <Operation type="Enumeration" value="MAC"/>
0065     <Operation type="Enumeration" value="MACVerify"/>
0066     <Operation type="Enumeration" value="Hash"/>
0067     <Operation type="Enumeration" value="CreateSplitKey"/>
0068     <Operation type="Enumeration" value="JoinSplitKey"/>
0069     <ObjectType type="Enumeration" value="Certificate"/>
0070     <ObjectType type="Enumeration" value="SymmetricKey"/>
0071     <ObjectType type="Enumeration" value="SecretData"/>
0072     <ObjectType type="Enumeration" value="PublicKey"/>
0073     <ObjectType type="Enumeration" value="PrivateKey"/>
0074     <ObjectType type="Enumeration" value="Template"/>
0075     <ObjectType type="Enumeration" value="OpaqueObject"/>
0076     <ObjectType type="Enumeration" value="SplitKey"/>
0077     <ObjectType type="Enumeration" value="PGPKey"/>
0078     </ResponsePayload>
0079     </BatchItem>
0080 </ResponseMessage>

```

337 6 Conformance

338 6.1 Suite B minLOS_128 Client KMIP V1.0 Profile Conformance

339 KMIP client implementations conformant to this profile:

- 340 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 341 2. SHALL support the conditions as specified in Section 2.2 of this profile.
- 342 3. SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.0 (3.1)

343 6.2 Suite B minLOS_128 Client KMIP V1.1 Profile Conformance

344 KMIP client implementations conformant to this profile:

- 345 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 346 2. SHALL support the conditions as specified in Section 2.2 of this profile.
- 347 3. SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.1 (3.2)

348 6.3 Suite B minLOS_128 Client KMIP V1.2 Profile Conformance

349 KMIP client implementations conformant to this profile:

- 350 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 351 2. SHALL support the conditions as specified in Section 2.2 of this profile.
- 352 3. SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.2 (3.3)

353 6.4 Suite B minLOS_128 Server KMIP V1.0 Profile Conformance

354 KMIP server implementations conformant to this profile:

- 355 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 356 2. SHALL support the conditions as specified in Section 2.3 of this profile.
- 357 3. SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.0 (3.1)

358 6.5 Suite B minLOS_128 Server KMIP V1.1 Profile Conformance

359 KMIP server implementations conformant to this profile:

- 360 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 361 2. SHALL support the conditions as specified in Section 2.3 of this profile.
- 362 3. SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.1 (3.2)

363 6.6 Suite B minLOS_128 Server KMIP V1.2 Profile Conformance

364 KMIP server implementations conformant to this profile:

- 365 1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
- 366 2. SHALL support the conditions as specified in Section 2.3 of this profile.
- 367 SHALL support all the Mandatory Suite B minLOS_128 Test Cases KMIP 1.2 (3.3)

368 6.7 Suite B minLOS_192 Client KMIP V1.0 Profile Conformance

369 KMIP client implementations conformant to this profile:

- 370 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.

- 371 2. SHALL support the conditions as specified in Section 4.2 of this profile.
372 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases - KMIP v1.0 (5.1)

373 **6.8 Suite B minLOS_192 Client KMIP V1.1 Profile Conformance**

374 KMIP client implementations conformant to this profile:

- 375 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.
376 2. SHALL support the conditions as specified in Section 4.2 of this profile.
377 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases KMIP 1.1(5.2)

378 **6.9 Suite B minLOS_192 Client KMIP V1.2 Profile Conformance**

379 KMIP client implementations conformant to this profile:

- 380 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.
381 2. SHALL support the conditions as specified in Section 4.2 of this profile.
382 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases KMIP 1.2 (5.3)

383 **6.10 Suite B minLOS_192 Server KMIP V1.0 Profile Conformance**

384 KMIP server implementations conformant to this profile:

- 385 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.
386 2. SHALL support the conditions as specified in Section 4.3 of this profile.
387 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases - KMIP v1.0 (5.1)

388 **6.11 Suite B minLOS_192 Server KMIP V1.1 Profile Conformance**

389 KMIP server implementations conformant to this profile:

- 390 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.
391 2. SHALL support the conditions as specified in Section 4.3 of this profile.
392 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases KMIP 1.1(5.2)

393 **6.12 Suite B minLOS_192 Server KMIP V1.2 Profile Conformance**

394 KMIP server implementations conformant to this profile:

- 395 1. SHALL support the Authentication Suite conditions as specified in Section 4.1 of this profile.
396 2. SHALL support the conditions as specified in Section 4.3 of this profile.
397 3. SHALL support all the Mandatory Suite B minLOS_192 Test Cases KMIP 1.2 (5.3)

398 **6.13 Permitted Test Case Variations**

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

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

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

404 **6.13.1 Variable Items**

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

- 406 1. UniqueIdentifier

- 407 2. PrivateKeyUniqueIdentifier
- 408 3. PublicKeyUniqueIdentifier
- 409 4. UniqueBatchItemIdentifier
- 410 5. AsynchronousCorrelationValue
- 411 6. TimeStamp
- 412 7. KeyValue / KeyMaterial including:
 - 413 a. key material content returned for managed cryptographic objects which are generated by
 - 414 the server
 - 415 b. wrapped versions of keys where the wrapping key is dynamic or the wrapping contains
 - 416 variable output for each wrap operation
- 417 8. For response containing the output of cryptographic operation in Data / SignatureData/ MACData
- 418 / IVCounterNonce where:
 - 419 a. the managed object is generated by the server; or
 - 420 b. the operation inherently contains variable output
- 421 9. For the following DateTime attributes where the value is not specified in the request as a fixed
- 422 DateTime value:
 - 423 a. ActivationDate
 - 424 b. ArchiveDate
 - 425 c. CompromiseDate
 - 426 d. CompromiseOccurrenceDate
 - 427 e. DeactivationDate
 - 428 f. DestroyDate
 - 429 g. InitialDate
 - 430 h. LastChangeDate
 - 431 i. ProtectStartDate
 - 432 j. ProcessStopDate
 - 433 k. ValidityDate
 - 434 l. OriginalCreationDate
- 435 10. LinkedObjectIdentifier
- 436 11. DigestValue
 - 437 a. For those managed cryptographic objects which are dynamically generated
- 438 12. KeyFormatType
 - 439 a. The key format type selected by the server when it creates managed objects
- 440 13. Digest
 - 441 a. The HashingAlgorithm selected by the server when it calculates the digest for a managed
 - 442 object for which it has access to the key material
 - 443 b. The Digest Value
- 444 14. Extensions reported in Query for ExtensionList and ExtensionMap
- 445 15. Application Namespaces reported in Query
- 446 16. Object Types reported in Query other than those noted as required in this profile
- 447 17. Operation Types reported in Query other than those noted as required in this profile (or any
- 448 referenced profile documents)
- 449 18. For TextString attribute values containing test identifiers:
 - 450 a. Additional vendor or application prefixes

- 451 19. Additional attributes beyond those noted in the response
452
- 453 An implementation conformant to this Profile MAY allow the following response variations:
- 454 20. Object Group values – May or may not return one or more Object Group values not included in
455 the requests
- 456 21. y-CustomAttributes – May or may not include additional server-specific associated attributes not
457 included in requests
- 458 22. Message Extensions – May or may not include additional (non-critical) vendor extensions
- 459 23. TemplateAttribute – May or may not be included in responses where the Template Attribute
460 response is noted as optional in [KMIP-SPEC]
- 461 24. AttributeIndex – May or may not include Attribute Index value where the Attribute Index value is 0
462 for Protocol Versions 1.1 and above.
- 463 25. ResultMessage – May or may not be included in responses and the value (if included) may vary
464 from the text contained within the test case.
- 465 26. The list of Protocol Versions returned in a DiscoverVersion response may include additional
466 protocol versions if the request has not specified a list of client supported Protocol Versions.
- 467 27. VendorIdentification - The value (if included) may vary from the text contained within the test
468 case.

469 **6.13.2 Variable behavior**

- 470 An implementation conformant to this Profile SHALL allow variation of the following behavior:
- 471 1. A test may omit the clean-up requests and responses (containing Revoke and/or Destroy) at the
472 end of the test provided there is a separate mechanism to remove the created objects during
473 testing.
- 474 2. A test may omit the test identifiers if the client is unable to include them in requests. This includes
475 the following attributes:
- 476 a. Name; and
477 b. x-ID
- 478 3. A test MAY perform requests with multiple batch items or as multiple requests with a single batch
479 item provided the sequence of operations are equivalent
- 480 4. A request MAY contain an optional *Authentication* [KMIP_SPEC] structure within each request

Appendix A. Acknowledgments

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

Participants:

481 Hal Aldridge, Sypris Electronics
482 Mike Allen, Symantec
483 Gordon Arnold, IBM
484 Todd Arnold, IBM
485 Richard Austin, Hewlett-Packard
486 Lars Bagnert, PrimeKey
487 Elaine Barker, NIST
488 Peter Bartok, Venafi, Inc.
489 Tom Benjamin, IBM
490 Anthony Berglas, Cryptsoft
491 Mathias Björkqvist, IBM
492 Kevin Bocket, Venafi
493 Anne Bolgert, IBM
494 Alan Brown, Thales e-Security
495 Tim Bruce, CA Technologies
496 Chris Burchett, Credant Technologies, Inc.
497 Kelley Burgin, National Security Agency
498 Robert Burns, Thales e-Security
499 Chuck Castleton, Venafi
500 Kenli Chong, QuintessenceLabs
501 John Clark, Hewlett-Packard
502 Tom Clifford, Symantec Corp.
503 Doron Cohen, SafeNet, Inc
504 Tony Cox, Cryptsoft
505 Russell Dietz, SafeNet, Inc
506 Graydon Dodson, Lexmark International Inc.
507 Vinod Duggirala, EMC Corporation
508 Chris Dunn, SafeNet, Inc.
509 Michael Duren, Sypris Electronics
510 James Dzierzanowski, American Express CCoE
511 Faisal Faruqui, Thales e-Security
512 Stan Feather, Hewlett-Packard
513 David Finkelstein, Symantec Corp.
514 James Fitzgerald, SafeNet, Inc.
515 Indra Fitzgerald, Hewlett-Packard
516 Judith Furlong, EMC Corporation
517 Susan Gleeson, Oracle
518 Robert Griffin, EMC Corporation
519 Paul Grojean, Individual
520 Robert Haas, IBM
521 Thomas Hardjono, M.I.T.
522 ChengDong He, Huawei Technologies Co., Ltd.
523 Steve He, Vormetric
524 Kurt Heberlein, Hewlett-Packard
525 Larry Hofer, Emulex Corporation
526 Maryann Hondo, IBM
527 Walt Hubis, NetApp
528 Tim Hudson, Cryptsoft
529 Jonas Iggbom, Venafi, Inc.

530 Sitaram Inguva, American Express CCoE
531 Jay Jacobs, Target Corporation
532 Glen Jaquette, IBM
533 Mahadev Karadiguddi, NetApp
534 Greg Kazmierczak, Wave Systems Corp.
535 Marc Kenig, SafeNet, Inc.
536 Mark Knight, Thales e-Security
537 Kathy Kriese, Symantec Corporation
538 Mark Lambiase, SecureAuth
539 John Leiseboer, Quintessence Labs
540 Hal Lockhart, Oracle Corporation
541 Robert Lockhart, Thales e-Security
542 Anne Luk, Cryptsoft
543 Sairam Manidi, Freescale
544 Luther Martin, Voltage Security
545 Neil McEvoy, iFOSSF
546 Marina Milshtein, Individual
547 Dale Moberg, Axway Software
548 Jishnu Mukeri, Hewlett-Packard
549 Bryan Olson, Hewlett-Packard
550 John Peck, IBM
551 Rob Philpott, EMC Corporation
552 Denis Pochuev, SafeNet, Inc.
553 Reid Poole, Venafi, Inc.
554 Ajai Puri, SafeNet, Inc.
555 Saravanan Ramalingam, Thales e-Security
556 Peter Reed, SafeNet, Inc.
557 Bruce Rich, IBM
558 Christina Richards, American Express CCoE
559 Warren Robbins, Dell
560 Peter Robinson, EMC Corporation
561 Scott Rotondo, Oracle
562 Saikat Saha, SafeNet, Inc.
563 Anil Saldhana, Red Hat
564 Subhash Sankuratripati, NetApp
565 Boris Schumperli, Cryptomathic
566 Greg Singh, QuintessenceLabs
567 David Smith, Venafi, Inc
568 Brian Spector, Certivox
569 Terence Spies, Voltage Security
570 Deborah Steckroth, RouteOne LLC
571 Michael Stevens, QuintessenceLabs
572 Marcus Streets, Thales e-Security
573 Satish Sundar, IBM
574 Kiran Thota, VMware
575 Somanchi Trinath, Freescale Semiconductor, Inc.
576 Nathan Turajski, Thales e-Security
577 Sean Turner, IECA, Inc.
578 Paul Turner, Venafi, Inc.
579 Rod Wideman, Quantum Corporation
580 Steven Wierenga, Hewlett-Packard
581 Jin Wong, QuintessenceLabs
582 Sameer Yami, Thales e-Security
583 Peter Yee, EMC Corporation
584 Krishna Yellepeddy, IBM
585 Catherine Ying, SafeNet, Inc.
586 Tatu Ylonen, SSH Communications Security (Tectia Corp)

587 Michael Yoder, Vormetric. Inc.
588 Magda Zdunkiewicz, Cryptsoft
589 Peter Zelechowski, Election Systems & Software

Appendix B. KMIP Specification Cross Reference

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
1 Introduction			
<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.
2 Objects			
<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.
3 Attributes			
<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.

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
<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.
4 Client-to-Server Operations			
<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.

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
<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.
5 Server-to-Client Operations			
<i>Notify</i>	5.1.	5.1.	5.1.
<i>Put</i>	5.2.	5.2.	5.2.
6 Message Contents			
<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.
7 Message Format			

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
<i>Message Structure</i>	7.1.	7.1.	7.1.
<i>Operations</i>	7.2.	7.2.	7.2.
8 Authentication			
<i>Authentication</i>	8	8	8
9 Message Encoding			
<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.

Reference Term	KMIP 1.0	KMIP 1.1	KMIP 1.2
<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.	-	-
10 Transport			
<i>Transport</i>	10	10	10
12 KMIP Server and Client Implementation Conformance			
<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.

590

Appendix C. Revision History

Revision	Date	Editor	Changes Made
wd01	10 July 2013	Kelley Burgin / Tim Hudson	Initial Draft
wd02	8 August 2013	Kelley Burgin	Editorial updates and inclusion of a corresponding restriction on client enumeration usage
wd03	10 August 2013	Tim Hudson	Updated Permitted Test Case Variations
wd03a	24-October-2013	Tim Hudson	Editorial update to include VendorIdentification in the list of allowed variations as per TC motion.
pr01update	11-June-2014	Tim Hudson	Updated following Public Review

591