KMIP Post-Quantum Cryptography Profile Working Draft 02

OASIS Working Draft

9 May 2017

Specification URIs

This version:

<<INSERT>>

Latest version:

<<INSERT>>

Technical Committee:

OASIS Key Management Interoperability Protocol (KMIP) TC

Chairs:

Saikat Saha (saikat.saha@oracle.com), Oracle
Tony Cox (tony.cox@cryptsoft.com), Cryptsoft

Editors:

Tim Hudson (tjh@cryptsoft.com), Cryptsoft
Tony Cox (tony.cox@cryptsoft.com), Cryptsoft

Related work:

This specification is related to:

- *Key Management Interoperability Protocol Profiles Version 1.3*. Edited by Tim Hudson and Robert Lockhart. Latest version: http://docs.oasis-open.org/kmip/profiles/v1.3/kmip-profiles-v1.3.html.
- *Key Management Interoperability Protocol Specification Version 1.3*. Edited by Tony Cox and Kiran Thota. Latest version: http://docs.oasis-open.org/kmip/spec/v1.3/kmip-spec-v1.3.html.

Abstract:

Describes a profile for KMIP clients and KMIP servers using post-quantum cryptography long term security as recommended.

Status:

This document was last revised or approved by the membership of OASIS 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. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=kmip#technical.

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.

Notices

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

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

Implementations conforming to this profile will allow users to switch to post-quantum cryptography: cryptographic systems that are not merely secure for today but that should also remain secure long-term against attacks by quantum computers.

Post-quantum cryptographic primitives include:

Encryption	SHOULD ChaCha20 (with 256-bit key) MAY AES-256
Digital Signature	SHOULD SPHINCS-256 (stateless) SHOULD XMSS (statefull)
Key Exchange	SHALL McEliece (with binary Goppa codes using length n = 6960, dimension k = 5413 and adding t = 119 errors).
Encryption with Authentication	SHOULD ChaCha20Poly1305 (with 256-bit key) MAY AES-256 (with 96 bit nonce in GCM)
Hashes	SHOULD SHA3-384 or SHA3-512 MAY SHA-384 or SHA-512

1.1 Terminology

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

1.2 Normative References

[CHACHA] *D. J. Bernstein. ChaCha, a variant of Salsa20*. https://cr.yp.to/chacha/chacha-20080128.pdf

[KMIP-SPEC] *Key Management Interoperability Protocol Specification Version 1.3*. Edited by Tony Cox and Kiran Thota. Latest version: http://docs.oasis-open.org/kmip/spec/v1.3/kmip-spec-v1.3.html.

[KMIP-PROF] *Key Management Interoperability Protocol Profiles Version 1.3*. Edited by Tim Hudson and Robert Lockhart. Latest version: http://docs.oasis-open.org/kmip/profiles/v1.3/kmip-profiles-v1.3.html.

[POLY1305] Daniel J. Bernstein. The Poly1305-AES Message-Authentication Code. In Henri Gilbert and Helena Handschuh, editors, Fast Software Encryption: 12th International Workshop, FSE 2005, Paris, France, February 21-23, 2005, Revised Selected Papers, volume 3557 of Lecture Notes in Computer Science, pages 32–49. Springer, 2005.

[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.

[RFC5246] Dierks, T. and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.2*, IETF RFC 5246, August 2008, http://www.ietf.org/rfc/rfc5246.txt.

[RFC8032] Josefsson, S. and Liusvaara, I, *Edwards-Curve Digital Signature Algorithm (EdDSA)*, IETF RFC 8032, January 2017, http://www.ietf.org/rfc/rfc8032.txt.

[SPHINCS] Daniel J. Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe, and Zooko Wilcox-O'Hearn. SPHINCS: Practical Stateless Hash-Based Signatures. In Elisabeth Oswald and Marc Fischlin, editors, Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I, volume 9056 of Lecture Notes in Computer Science, pages 368–397. Springer, 2015.

[XMSS] Johannes A. Buchmann, Erik Dahmen, and Andreas Hülsing. XMSS - A Practical Forward Secure Signature Scheme Based on Minimal Security Assumptions. In BoYin Yang, editor, Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29 - December 2, 2011. Proceedings, volume 7071 of Lecture Notes in Computer Science, pages 117–129. Springer, 2011.

2 Post-Quantum Cryptography Profile

The Post-Quantum Cryptography Profile describes a KMIP client interacting with a KMIP server in a manner that should also remain secure long-term against attacks by quantum computers, whilst providing a more flexible set of options for handling known or suspected PQC vulnerabilities.

2.1 Authentication Suite

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

2.1.1 Protocols

Conformant KMIP clients and servers SHOULD support:

- TLS v1.3 [RFC-PENDING]

Conformant KMIP clients and servers MAY support:

- TLS v1.2 [RFC5246]

Conformant KMIP clients and servers SHALL NOT support:

- Any other TLS or SSL protocol version

2.1.2 Cipher Suites

Conformant KMIP servers SHALL support the following cipher suites for TLSv1.3 if TLSv1.3 is supported:

- TLS13-CHACHA20-POLY1305-SHA256
- TLS13-AES-256-GCM-SHA384

Conformant KMIP servers SHALL support the following cipher suites for TLSv1.2 if TLSv1.2 is supported:

- TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

2.1.3 Client Authenticity

Conformant KMIP servers and clients SHALL handle client authenticity in accordance with section 3.1.3 of the Basic Authentication Suite [KMIP-PROF].

2.1.4 KMIP Port Number

Conformant KMIP servers and clients SHALL handle the KMIP port number in in accordance with section 3.1.4 of the Basic Authentication Suite [KMIP-PROF].

2.2 Post-Quantum Cryptography - Client

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

1. SHALL conform to the *Baseline Client* (section 5.1) of [KMIP-PROF]
2. SHALL restrict use of the enumerated types listed in item 5 of the server list in section 2.3 to the values noted against each item
3. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause within this section 2.2.
4. MAY support extensions outside the scope of this standard (e.g., vendor extensions, conformance clauses) that do not conflict with any KMIP requirements.

2.3 Post-Quantum Cryptography - Server

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

1. SHALL conform to the *Baseline Server* of [KMIP-PROF]
2. SHALL support the following *Objects* [KMIP-SPEC]
    a. *Certificate* [KMIP-SPEC]
    b. *Symmetric Key* [KMIP-SPEC]
    c. *Public Key* [KMIP-SPEC]
    d. *Private Key* [KMIP-SPEC]
3. SHALL support the following *Attributes* [KMIP-SPEC]
    a. *Cryptographic Algorithm* [KMIP-SPEC]
    b. *Cryptographic Length* [KMIP-SPEC] value:
    c. *Protection Period* [KMIP-SPEC] [INTERVAL]
    d. *Protection Type* [KMIP-SPEC] (LEVEL - LOW, MEDIUM, HIGH)
    e. *Post-Quantum Crypto* [KMIP-SPEC] (Boolean)
4. SHALL support the following C*lient-to-Server Operations* [KMIP-SPEC]:
    a. *Create* [KMIP-SPEC]
    b. *Create Key Pair* [KMIP-SPEC]
    c. *Register* [KMIP-SPEC]
    d. *Re-key* [KMIP-SPEC]
    e. *Re-key Key Pair* [KMIP-SPEC]
    f. *Certify* [KMIP-SPEC]
    g. *Re-Certify* [KMIP-SPEC]
    h. *Encrypt* [KMIP-SPEC]
    i. *Decrypt* [KMIP-SPEC]
    j. *Sign* [KMIP-SPEC]
    k. *SignatureVerify* [KMIP-SPEC]
5. SHALL support the following Server-*to-Client Operations* [KMIP-SPEC]:
    a. *Notify* [KMIP-SPEC]
6. SHALL support the following *Message Encoding* [KMIP-SPEC]:
    a. *Recommended Curve* [KMIP-SPEC] value:
        i. P-384 (SECP384R1)
        ii. P-521
    b. *Certificate Type Enumeration* [KMIP-SPEC] value:
        i. X.509
    c. *Cryptographic Algorithm Enumeration* [KMIP-SPEC] value:
        i. AES
        ii. ChaCha20
        iii. ChaChar20Poly1305
    d. *Hashing Algorithm Enumeration* [KMIP-SPEC]
        i. SHA-384
        ii. SHA-512
        iii. SHA3-256

iv. SHA3-384
v. SHA3-512

e. *Object Type Enumeration* [KMIP-SPEC] value:
   i. Certificate
   ii. Symmetric Key
   iii. Public Key
   iv. Private Key

f. *Key Format Type Enumeration* [KMIP-SPEC] value:
   i. Raw
   ii. X.509

g. *Digital Signature Algorithm Enumeration* [KMIP-SPEC] value:
   i. ECDSA with SHA384 (on P-384)
   ii. Ed25519 with Ed25519

7. MAY support any clause within [KMIP-SPEC] provided it does not conflict with any other clause within this section 2.3.
8. MAY support extensions outside the scope of this standard (e.g., vendor extensions, conformance clauses) that do not conflict with any KMIP requirements.

3 Post-Quantum Cryptography Test Cases

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-PROF] 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.

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.

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.

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.

Note: the values for the returned items and the custom attributes are illustrative. Actual values from a real client or server system may vary as specified in section 4.2

3.1 Mandatory Post-Quantum Cryptography Test Cases - KMIP v2.0

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

3.1.1 PQC-M-1-12 - Query

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

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

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="2"/>
0005        <ProtocolVersionMinor type="Integer" value="0"/>
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="2"/>
0021        <ProtocolVersionMinor type="Integer" value="0"/>
0022      </ProtocolVersion>
0023      <TimeStamp type="DateTime" value="2017-04-26T09:09:17+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="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>
```

165

166 3.1.2 PQC-M-2-20 - Create

167 Perform a Create operation, stating the period the key must be able to offer protection (Protection Period)
168 and the relative sensitivity of the information (Protection Type).

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

170 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="2"/>
0005          <ProtocolVersionMinor type="Integer" value="0"/>
0006        </ProtocolVersion>
```



```
0007      <BatchCount type="Integer" value="1"/>
0008    </RequestHeader>
0009    <BatchItem>
0010      <Operation type="Enumeration" value="Create"/>
0011      <RequestPayload>
0012  …
0013      </RequestPayload>
0014    </BatchItem>
0015  </RequestMessage>
0016
```

```
0017  <ResponseMessage>
0018    <ResponseHeader>
0019      <ProtocolVersion>
0020        <ProtocolVersionMajor type="Integer" value="2"/>
0021        <ProtocolVersionMinor type="Integer" value="0"/>
0022      </ProtocolVersion>
0023      <TimeStamp type="DateTime" value="2016-06-26T09:09:17+00:00"/>
0024      <BatchCount type="Integer" value="1"/>
0025    </ResponseHeader>
0026    <BatchItem>
0027      <Operation type="Enumeration" value="Create"/>
0028      <ResultStatus type="Enumeration" value="Success"/>
0029      <ResponsePayload>
0030  …
0031      </ResponsePayload>
0032    </BatchItem>
0033  </ResponseMessage>
0034
```

171

4 Conformance

4.1 Post-Quantum Cryptography Client KMIP V2.0 Profile Conformance

KMIP client implementations conformant to this profile:

1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
2. SHALL support the conditions as specified in Section 2.2 of this profile.
3. SHALL support one or more of the Mandatory Post-Quantum Cryptography Test Cases - KMIP v (3.1)

4.2 Post-Quantum Cryptography Server KMIP V2.0 Profile Conformance

KMIP server implementations conformant to this profile:

1. SHALL support the Authentication Suite conditions as specified in Section 2.1 of this profile.
2. SHALL support the conditions as specified in Section 2.3 of this profile.
3. SHALL support all the Mandatory Post-Quantum Cryptography Test Cases - KMIP v (3.1)

4.3 Permitted Test Case Variations

Conformant KMIP servers and clients SHALL handle permitted test case variations in accordance with section 4.1 Permitted Test Case Variations of [KMIP-PROF].

Appendix A. Acknowledgments

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

Participants:

Prof. Dr. Tanja Lange

<<INSERT>>

Appendix B. KMIP Specification Cross Reference

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

Reference Term	**KMIP 1.0**	**KMIP 1.1**	**KMIP 1.2 / KMIP 1.3**
Archive Date	3.27.	3.32.	3.32.
Attributes	3	3	3
Certificate Identifier	3.9.	3.13.	3.13.
Certificate Issuer	3.11.	3.15.	3.15.
Certificate Length	-	3.9.	3.9.
Certificate Subject	3.10.	3.14.	3.14.
Certificate Type	3.8.	3.8.	3.8.
Compromise Date	3.25.	3.30.	3.30.
Compromise Occurrence Date	3.24.	3.29.	3.29.
Contact Information	3.31.	3.37.	3.37.
Cryptographic Algorithm	3.4.	3.4.	3.4.
Cryptographic Domain Parameters	3.7.	3.7.	3.7.
Cryptographic Length	3.5.	3.5.	3.5.
Cryptographic Parameters	3.6.	3.6.	3.6.
Custom Attribute	3.33.	3.39.	3.39.
Deactivation Date	3.22.	3.27.	3.27.
Default Operation Policy	3.13.2.	3.18.2.	3.18.2.
Default Operation Policy for Certificates and Public Key Objects	3.13.2.2.	3.18.2.2.	3.18.2.2.
Default Operation Policy for Secret Objects	3.13.2.1.	3.18.2.1.	3.18.2.1.
Default Operation Policy for Template Objects	3.13.2.3.	3.18.2.3.	3.18.2.3.
Destroy Date	3.23.	3.28.	3.28.
Digest	3.12.	3.17.	3.17.
Digital Signature Algorithm	-	3.16.	3.16.
Fresh	-	3.34.	3.34.
Initial Date	3.18.	3.23.	3.23.
Key Value Location	-	-	3.42.
Key Value Present	-	-	3.41.
Last Change Date	3.32.	3.38.	3.38.
Lease Time	3.15.	3.20.	3.20.
Link	3.29.	3.35.	3.35.
Name	3.2.	3.2.	3.2.
Object Group	3.28.	3.33.	3.33.
Object Type	3.3.	3.3.	3.3.
Operation Policy Name	3.13.	3.18.	3.18.
Operations outside of operation policy control	3.13.1.	3.18.1.	3.18.1.
Original Creation Date	-	-	3.43.
Process Start Date	3.20.	3.25.	3.25.
Protect Stop Date	3.21.	3.26.	3.26.
Revocation Reason	3.26.	3.31.	3.31.
State	3.17.	3.22.	3.22.
Unique Identifier	3.1.	3.1.	3.1.
Usage Limits	3.16.	3.21.	3.21.
X.509 Certificate Identifier	-	3.10.	3.10.
X.509 Certificate Issuer	-	3.12.	3.12.
X.509 Certificate Subject	-	3.11.	3.11.
4 Client-to-Server Operations			
Activate	4.18.	4.19.	4.19.
Add Attribute	4.13.	4.14.	4.14.
Archive	4.21.	4.22.	4.22.
Cancel	4.25.	4.27.	4.27.
Certify	4.6.	4.7.	4.7.
Check	4.9.	4.10.	4.10.

Reference Term	**KMIP 1.0**	**KMIP 1.1**	**KMIP 1.2 / KMIP 1.3**
Create	4.1.	4.1.	4.1.
Create Key Pair	4.2.	4.2.	4.2.
Create Split Key	-	-	4.38.
Decrypt	-	-	4.30.
Delete Attribute	4.15.	4.16.	4.16.
Derive Key	4.5.	4.6.	4.6.
Destroy	4.20.	4.21.	4.21.
Discover Versions	-	4.26.	4.26.
Encrypt	-	-	4.29.
Get	4.10.	4.11.	4.11.
Get Attribute List	4.12.	4.13.	4.13.
Get Attributes	4.11.	4.12.	4.12.
Get Usage Allocation	4.17.	4.18.	4.18.
Hash	-	-	4.37.
Join Split Key	-	-	4.39.
Locate	4.8.	4.9.	4.9.
MAC	-	-	4.33.
MAC Verify	-	-	4.34.
Modify Attribute	4.14.	4.15.	4.15.
Obtain Lease	4.16.	4.17.	4.17.
Poll	4.26.	4.28.	4.28.
Query	4.24.	4.25.	4.25.
Re-certify	4.7.	4.8.	4.8.
Recover	4.22.	4.23.	4.23.
Register	4.3.	4.3.	4.3.
Re-key	4.4.	4.4.	4.4.
Re-key Key Pair	-	4.5.	4.5.
Revoke	4.19.	4.20.	4.20.
RNG Retrieve	-	-	4.35.
RNG Seed	-	-	4.36.
Sign	-	-	4.31.
Signature Verify	-	-	4.32.
Validate	4.23.	4.24.	4.24.
5 Server-to-Client Operations			
Notify	5.1.	5.1.	5.1.
Put	5.2.	5.2.	5.2.
6 Message Contents			
Asynchronous Correlation Value	6.8.	6.8.	6.8.
Asynchronous Indicator	6.7.	6.7.	6.7.
Attestation Capable Indicator	-	-	6.17.
Batch Count	6.14.	6.14.	6.14.
Batch Error Continuation Option	6.13.	6.13.	6.13.
Batch Item	6.15.	6.15.	6.15.
Batch Order Option	6.12.	6.12.	6.12.
Maximum Response Size	6.3.	6.3.	6.3.
Message Extension	6.16.	6.16.	6.16.
Operation	6.2.	6.2.	6.2.
Protocol Version	6.1.	6.1.	6.1.
Result Message	6.11.	6.11.	6.11.
Result Reason	6.10.	6.10.	6.10.
Result Status	6.9.	6.9.	6.9.
Time Stamp	6.5.	6.5.	6.5.

Reference Term	**KMIP 1.0**	**KMIP 1.1**	**KMIP 1.2 / KMIP 1.3**
Unique Batch Item ID	6.4.	6.4.	6.4.
7 Message Format			
Message Structure	7.1.	7.1.	7.1.
Operations	7.2.	7.2.	7.2.
8 Authentication			
Authentication	8	8	8
9 Message Encoding			
Alternative Name Type Enumeration	-	-	9.1.3.2.34.
Attestation Type Enumeration	-	-	9.1.3.2.36.
Batch Error Continuation Option Enumeration	9.1.3.2.29.	9.1.3.2.30.	9.1.3.2.30.
Bit Masks	9.1.3.3.	9.1.3.3.	9.1.3.3.
Block Cipher Mode Enumeration	9.1.3.2.13.	9.1.3.2.14.	9.1.3.2.14.
Cancellation Result Enumeration	9.1.3.2.24.	9.1.3.2.25.	9.1.3.2.25.
Certificate Request Type Enumeration	9.1.3.2.21.	9.1.3.2.22.	9.1.3.2.22.
Certificate Type Enumeration	9.1.3.2.6.	9.1.3.2.6.	9.1.3.2.6.
Credential Type Enumeration	9.1.3.2.1.	9.1.3.2.1.	9.1.3.2.1.
Cryptographic Algorithm Enumeration	9.1.3.2.12.	9.1.3.2.13.	9.1.3.2.13.
Cryptographic Usage Mask	9.1.3.3.1.	9.1.3.3.1.	9.1.3.3.1.
Defined Values	9.1.3.	9.1.3.	9.1.3.
Derivation Method Enumeration	9.1.3.2.20.	9.1.3.2.21.	9.1.3.2.21.
Digital Signature Algorithm Enumeration	-	9.1.3.2.7.	9.1.3.2.7.
Encoding Option Enumeration	-	9.1.3.2.32.	9.1.3.2.32.
Enumerations	9.1.3.2.	9.1.3.2.	9.1.3.2.
Examples	9.1.2.	9.1.2.	9.1.2.
Hashing Algorithm Enumeration	9.1.3.2.15.	9.1.3.2.16.	9.1.3.2.16.
Item Length	9.1.1.3.	9.1.1.3.	9.1.1.3.
Item Tag	9.1.1.1.	9.1.1.1.	9.1.1.1.
Item Type	9.1.1.2.	9.1.1.2.	9.1.1.2.
Item Value	9.1.1.4.	9.1.1.4.	9.1.1.4.
Key Compression Type Enumeration	9.1.3.2.2.	9.1.3.2.2.	9.1.3.2.2.
Key Format Type Enumeration	9.1.3.2.3.	9.1.3.2.3.	9.1.3.2.3.
Key Role Type Enumeration	9.1.3.2.16.	9.1.3.2.17.	9.1.3.2.17.
Key Value Location Type Enumeration	-	-	9.1.3.2.35.
Link Type Enumeration	9.1.3.2.19.	9.1.3.2.20.	9.1.3.2.20.
Name Type Enumeration	9.1.3.2.10.	9.1.3.2.11.	9.1.3.2.11.
Object Group Member Enumeration	-	9.1.3.2.33.	9.1.3.2.33.
Object Type Enumeration	9.1.3.2.11.	9.1.3.2.12.	9.1.3.2.12.
Opaque Data Type Enumeration	9.1.3.2.9.	9.1.3.2.10.	9.1.3.2.10.
Operation Enumeration	9.1.3.2.26.	9.1.3.2.27.	9.1.3.2.27.
Padding Method Enumeration	9.1.3.2.14.	9.1.3.2.15.	9.1.3.2.15.
Put Function Enumeration	9.1.3.2.25.	9.1.3.2.26.	9.1.3.2.26.
Query Function Enumeration	9.1.3.2.23.	9.1.3.2.24.	9.1.3.2.24.
Recommended Curve Enumeration for ECDSA, ECDH, and ECMQV	9.1.3.2.5.	9.1.3.2.5.	9.1.3.2.5.
Result Reason Enumeration	9.1.3.2.28.	9.1.3.2.29.	9.1.3.2.29.
Result Status Enumeration	9.1.3.2.27.	9.1.3.2.28.	9.1.3.2.28.
Revocation Reason Code Enumeration	9.1.3.2.18.	9.1.3.2.19.	9.1.3.2.19.
Secret Data Type Enumeration	9.1.3.2.8.	9.1.3.2.9.	9.1.3.2.9.
Split Key Method Enumeration	9.1.3.2.7.	9.1.3.2.8.	9.1.3.2.8.
State Enumeration	9.1.3.2.17.	9.1.3.2.18.	9.1.3.2.18.
Storage Status Mask	9.1.3.3.2.	9.1.3.3.2.	9.1.3.3.2.

Reference Term	**KMIP 1.0**	**KMIP 1.1**	**KMIP 1.2 / KMIP 1.3**
Tags	9.1.3.1.	9.1.3.1.	9.1.3.1.
TTLV Encoding	9.1.	9.1.	9.1.
TTLV Encoding Fields	9.1.1.	9.1.1.	9.1.1.
Usage Limits Unit Enumeration	9.1.3.2.30.	9.1.3.2.31.	9.1.3.2.31.
Validity Indicator Enumeration	9.1.3.2.22.	9.1.3.2.23.	9.1.3.2.23.
Wrapping Method Enumeration	9.1.3.2.4.	9.1.3.2.4.	9.1.3.2.4.
XML Encoding	9.2.	-	-
10 Transport			
Transport	10	10	10
12 KMIP Server and Client Implementation Conformance			
Conformance clauses for a KMIP Server	12.1.	-	-
KMIP Client Implementation Conformance	-	12.2.	12.2.
KMIP Server Implementation Conformance	-	12.1.	12.1.

192

Appendix C. Revision History

Revision	Date	Editor	Changes Made
wd02	9 May 2017	Tim Hudson / Tony Cox	Updates based on discussions
wd01	10 August 2016	Tim Hudson	Initial Working Draft

193

