Key Management Interoperability Protocol Profiles Version 1.2

OASIS Standard

19 May 2015

Specification URIs

This version:

http://docs.oasis-open.org/kmip/profiles/v1.2/os/kmip-profiles-v1.2-os.doc (Authoritative)
http://docs.oasis-open.org/kmip/profiles/v1.2/os/kmip-profiles-v1.2-os.html
http://docs.oasis-open.org/kmip/profiles/v1.2/os/kmip-profiles-v1.2-os.pdf

Previous version:

http://docs.oasis-open.org/kmip/profiles/v1.2/csprd01/kmip-profiles-v1.2-csprd01.doc (Authoritative)
http://docs.oasis-open.org/kmip/profiles/v1.2/csprd01/kmip-profiles-v1.2-csprd01.html
http://docs.oasis-open.org/kmip/profiles/v1.2/csprd01/kmip-profiles-v1.2-csprd01.pdf

Latest version:

http://docs.oasis-open.org/kmip/profiles/v1.2/kmip-profiles-v1.2.doc (Authoritative)
http://docs.oasis-open.org/kmip/profiles/v1.2/kmip-profiles-v1.2.html
http://docs.oasis-open.org/kmip/profiles/v1.2/kmip-profiles-v1.2.pdf

Technical Committee:

OASIS Key Management Interoperability Protocol (KMIP) TC

Chairs:

Saikat Saha (saikat.saha@oracle.com), Oracle
Tony Cox (tjc@cryptsoft.com), Cryptsoft Pty Ltd.

Editors:

Tim Hudson (tjh@cryptsoft.com), Cryptsoft Pty Ltd.
Robert Lockhart (Robert.Lockhart@thalesesec.com), Thales e-Security

Related work:

This specification replaces or supersedes:

- *Key Management Interoperability Protocol Profiles Version 1.1*. Edited by Robert Griffin and Subhash Sankuratripati. Latest version http://docs.oasis-open.org/kmip/profiles/v1.1/kmip-profiles-v1.1.html.

This specification is related to:

- *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.
- *Key Management Interoperability Protocol Test Cases Version 1.2*. Edited by Tim Hudson and Faisal Faruqui. Latest version. http://docs.oasis-open.org/kmip/testcases/v1.2/kmip-testcases-v1.2.html.
- *Key Management Interoperability Protocol Usage Guide Version 1.2*. Edited by Indra Fitzgerald and Judith Furlong. Latest version. http://docs.oasis-open.org/kmip/ug/v1.2/kmip-ug-v1.2.html.

Abstract:

This document is intended for developers and architects who wish to design systems and applications that conform to the Key Management Interoperability Protocol specification.

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.

Citation format:

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

[KMIP-Profiles]

Key Management Interoperability Protocol Profiles Version 1.2. Edited by Tim Hudson and Robert Lockhart. 19 May 2015. OASIS Standard. http://docs.oasis-open.org/kmip/profiles/v1.2/os/kmip-profiles-v1.2-os.html. Latest version: http://docs.oasis-open.org/kmip/profiles/v1.2/kmip-profiles-v1.2.html.

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

OASIS requires a conformance section in an approved committee specification ([KMIP-SPEC] [TC-PROC], section 2.18 Work Product Quality, paragraph 8a):

A specification that is approved by the TC at the Public Review Draft, Committee Specification or OASIS Standard level must include a separate section, listing a set of numbered conformance clauses, to which any implementation of the specification must adhere in order to claim conformance to the specification (or any optional portion thereof).

This document intends to meet this OASIS requirement on conformance clauses for a KMIP server or KMIP client ([KMIP-SPEC] 12.1, 12.2) through profiles that define the use of KMIP objects, attributes, operations, message elements and authentication methods within specific contexts of KMIP server and client interaction.

These profiles define a set of normative constraints for employing KMIP within a particular environment or context of use. They may, optionally, require the use of specific KMIP functionality or in other respects define the processing rules to be followed by profile actors.

For normative definition of the elements of KMIP specified in these profiles, see the KMIP Specification ([KMIP-SPEC]).

1.1 Terminology

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

1.2 Normative References

[KMIP-SPEC] *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.doc.

[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.

[RFC2246] T. Dierks & C.Allen, *The TLS Protocol, Version 1.0***,** http://www.ietf.org/rfc/rfc2246.txt**,** IETF RFC 2246, January 1999

[RFC3268] P. Chown, *Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security* (TLS), http://www.ietf.org/rfc/rfc3268.txt, IETF RFC 3268, June 2002

[RFC4346] T. Dierks & E. Rescorla, *The Transport Layer Security (TLS) Protocol, Version 1.1*, http://www.ietf.org/rfc/rfc4346.txt, IETF RFC 4346, April 2006

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

1.3 Non-Normative References

[TC-PROC] *OASIS TC Process.* 14 February 2013. OASIS Process. https://www.oasis-open.org/policies-guidelines/tc-process.

2 Profiles

This document defines a selected set of conformance clauses and authentication suites which when combined form KMIP Profiles.

2.1 Guidelines for Specifying Conformance Clauses

This section provides a checklist of issues that SHALL be addressed by each clause.

1. Implement functionality as mandated by **[KMIP-SPEC]** Section 12 (Conformance clauses for a KMIP server or a KMIP client)
2. Specify the list of additional objects that SHALL be supported
3. Specify the list of additional attributes that SHALL be supported
4. Specify the list of additional operations that SHALL be supported
5. Specify any additional message content that SHALL be supported

2.2 Guidelines for Specifying Authentication Suites

1. Channel Security – For all operations, communication between client and server SHALL establish and maintain channel confidentiality and integrity,.
2. Channel Options – Options like protocol version and cipher suite
3. Server and Client Authenticity – For all operations, communication between client and server SHALL provide assurance of server authenticity and client authenticity

2.3 Guidelines for Specifying KMIP Profiles

Any vendor or organization, such as other standards bodies, MAY create a KMIP Profile and publish it.

1. The profile SHALL be publicly available.
2. The KMIP Technical Committee SHALL be formally advised of the availability of the profile and the location of the published profile.
3. The profile SHALL be defined as a tuple of {Conformance Clause, Authentication Suite}.
4. The KMIP Technical Committee SHOULD review the profile prior to publication.

2.4 Guidelines for Validating Conformance to KMIP Server Profiles

A KMIP server implementation SHALL claim conformance to a specific server profile only if it supports all required objects, operations, messaging and attributes of that profile

1. All objects specified as required in that profile
2. All operations specified as required in that profile
3. All attributes specified as required in that profile
4. The defined wire protocols (TLS, SSL, IPSec, etc…) for that profile
5. The defined methods of authentication for that profile

2.5 Guidelines for Validating Conformance to KMIP Client Profiles

A KMIP client implementation SHALL claim conformance to a specific client profile only if it supports all required objects, operations, messaging and attributes of that profile

1. All objects specified as required in that profile
2. All operations specified as required in that profile
3. All attributes specified as required in that profile
4. The defined wire protocols (TLS, SSL, IPSec, etc…) for that profile
5. The defined methods of authentication for that profile

3 Authentication Suites

This section contains the list of protocol versions and cipher suites that are to be used by profiles contained within this document.

3.1 Basic Authentication Suite

This authentication set stipulates that a conformant KMIP client or server SHALL use TLS to negotiate a secure connection.

3.1.1 Protocols

Conformant KMIP clients or servers SHALL support:

- TLS v1.0 [RFC2246] and [RFC3268]

Conformant KMIP clients or servers MAY support:

- TLS v1.1 [RFC4346]
- TLS v1.2 [RFC5246]

Conformant KMIP clients or servers SHALL NOT support:

- SSL v3.0
- SSL v2.0
- SSL v1.0

3.1.2 Cipher Suites

Conformant KMIP clients or servers SHALL support the following cipher suites:

- TLS_RSA_WITH_AES_128_CBC_SHA

Conformant KMIP clients and servers MAY support the following cipher suites:

- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA
- TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_DH_DSS_WITH_AES_128_CBC_SHA
- TLS_DH_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_DSS_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DH_DSS_WITH_AES_256_CBC_SHA
- TLS_DH_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_DSS_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DH_DSS_WITH_AES_128_CBC_SHA256
- TLS_DH_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_DSS_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DH_DSS_WITH_AES_256_CBC_SHA256
- TLS_DH_RSA_WITH_AES_256_CBC_SHA256
- TLS_DHE_DSS_WITH_AES_256_CBC_SHA256

- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
- TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_PSK_WITH_3DES_EDE_CBC_SHA
- TLS_PSK_WITH_AES_128_CBC_SHA
- TLS_PSK_WITH_AES_256_CBC_SHA
- TLS_DHE_PSK_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_PSK_WITH_AES_128_CBC_SHA
- TLS_DHE_PSK_WITH_AES_256_CBC_SHA
- TLS_RSA_PSK_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_PSK_WITH_AES_128_CBC_SHA
- TLS_RSA_PSK_WITH_AES_256_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384

Conformant KMIP clients or servers SHALL NOT support any cipher suite not listed above.

NOTE: TLS 1.0 has known security issues and implementations that need protections against known issues SHOULD considering using the TLS 1.2 Authentication Suite (3.2)

3.1.3 Client Authenticity

Conformant KMIP servers SHALL require the use of channel (TLS) mutual authentication to provide assurance of client authenticity for all operations other than:

- Query
- Discover Versions

Conformant KMIP servers SHALL use the identity derived from the channel mutual authentication to determine the client identity if the KMIP client requests do not contain an Authentication object.

Conformant KMIP servers SHALL use the identity derived from the channel mutual authentication along with the Credential information to determine the client identity if the KMIP client requests contain an Authentication object.

The exact mechanisms determining the client identity are outside the scope of this specification.

3.1.4 KMIP Port Number

Conformant KMIP servers SHOULD use TCP port number 5696, as assigned by IANA.

3.2 TLS 1.2 Authentication Suite

This authentication set stipulates that a conformant KMIP client and server SHALL use TLS to negotiate a mutually-authenticated connection.

3.2.1 Protocols

Conformant KMIP clients and servers SHALL support:

- TLS v1.2 [RFC2246]

3.2.2 Cipher Suites

Conformant KMIP servers SHALL support the following cipher suites:

- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256

Conformant KMIP servers and clients MAY support the cipher suites specified as MAY in section 3.2.2 of the Basic Authentication Suite.

3.2.3 Client Authenticity

Conformant KMIP servers and clients SHALL handle client authenticity in accordance with section 3.2.3 of the Basic Authentication Suite.

3.2.4 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.

4 KMIP Profiles

This section lists the KMIP profiles that are defined in this specification.

A KMIP server or KMIP client MAY support more than one profile at the same time provided there are no conflicting requirements between any of the supported profiles.

4.1 Baseline Server Basic KMIP Profile

The profile that consists of the tuple {Baseline Server, Basic Authentication Suite}.

4.2 Baseline Server TLS v1.2 KMIP Profile

A profile that consists of the tuple {Baseline Server, TLS 1.2 Authentication Suite}.

4.3 Baseline Client Basic KMIP Profile

The profile that consists of the tuple {Baseline Client, Basic Authentication Suite}.

4.4 Baseline Client TLS v1.2 KMIP Profile

A profile that consists of the tuple {Baseline Client, TLS 1.2 Authentication Suite}.

4.5 Complete Server Basic KMIP Profile

The profile that consists of the tuple {Complete Server, Basic Authentication Suite}.

4.6 Complete Server TLS v1.2 KMIP Profile

A profile that consists of the tuple {Complete Server, TLS 1.2 Authentication Suite}.

5 Conformance

The baseline server and client profiles provide the most basic functionality that is expected of a conformant KMIP client or server. The complete server profile defines a KMIP server that implements the entire specification. A KMIP implementation conformant to this specification (the Key Management Interoperability Protocol Profiles) SHALL meet all the conditions documented in one or more of the following sections.

Specific combinations of KMIP objects, operations, messaging and attributes beyond those defined in the following sections are specified in separate profile documents.

5.1 Baseline Server

The Baseline Server provides the most basic functionality that is expected of a conformant KMIP server – the ability to provide information about the server and the managed objects supported by the server.

An implementation is a conforming Baseline Server if it meets the following conditions:

1. Supports the conditions required by the KMIP Server conformance clauses ([KMIP-SPEC] 12.1)
2. Supports the following objects:
    a. Attribute ([KMIP-SPEC] 2.1.1)
    b. Credential ([KMIP-SPEC] 2.1.2)
    c. Key Block ([KMIP-SPEC] 2.1.3)
    d. Key Value ([KMIP-SPEC] 2.1.4)
    e. Template-Attribute Structure ([KMIP-SPEC] 2.1.8)
    f. Extension Information ([KMIP-SPEC] 2.1.9)
3. Supports the following subsets of attributes:
    a. Unique Identifier ([KMIP-SPEC] 3.1)
    b. Name ([KMIP-SPEC] 3.2)
    c. Object Type ([KMIP-SPEC] 3.3)
    d. Cryptographic Algorithm ([KMIP-SPEC] 3.4)
    e. Cryptographic Length ([KMIP-SPEC] 3.5)
    f. Cryptographic Parameters ([KMIP-SPEC] 3.6)
    g. Digest ([KMIP-SPEC] 3.17)
    h. Default Operation Policy ([KMIP-SPEC] 3.18.2)
    i. Cryptographic Usage Mask ([KMIP-SPEC] 3.19)
    j. State ([KMIP-SPEC] 3.22)
    k. Initial Date ([KMIP-SPEC] 3.23)
    l. Activation Date ([KMIP-SPEC] 3.24)
    m. Deactivation Date ([KMIP-SPEC] 3.27)
    n. Compromise Occurrence Date ([KMIP-SPEC] 3.29)
    o. Compromise Date ([KMIP-SPEC] 3.30)
    p. Revocation Reason ([KMIP-SPEC] 3.31)
    q. Last Change Date ([KMIP-SPEC] 3.38)
4. Supports the ID Placeholder ([KMIP-SPEC] 4)
5. Supports the following client-to-server operations:
    a. Locate ([KMIP-SPEC] 4.9)
    b. Check ([KMIP-SPEC] 4.10)
    c. Get ([KMIP-SPEC] 4.11)
    d. Get Attributes ([KMIP-SPEC] 4.12)
    e. Get Attribute List ([KMIP-SPEC] 4.13)
    f. Add Attribute ([KMIP-SPEC] 4.14)
    g. Modify Attribute ([KMIP-SPEC] 4.15)
    h. Delete Attribute ([KMIP-SPEC] 4.16)

i. Activate ([KMIP-SPEC] 4.19)
j. Revoke ([KMIP-SPEC] 4.20)
k. Destroy ([KMIP-SPEC] 4.21)
l. Query ([KMIP-SPEC] 4.25)
m. Discover Versions ([KMIP-SPEC] 4.26)

6. Supports the following message contents:
    a. Protocol Version ([KMIP-SPEC] 6.1)
    b. Operation ([KMIP-SPEC] 6.2)
    c. Maximum Response Size ([KMIP-SPEC] 6.3)
    d. Unique Batch Item ID ([KMIP-SPEC] 6.4)
    e. Time Stamp ([KMIP-SPEC] 6.5)
    f. Asynchronous Indicator ([KMIP-SPEC] 6.7)
    g. Result Status ([KMIP-SPEC] 6.9)
    h. Result Reason ([KMIP-SPEC] 6.10)
    i. Batch Order Option ([KMIP-SPEC] 6.12)
    j. Batch Error Continuation Option ([KMIP-SPEC] 6.13)
    k. Batch Count ([KMIP-SPEC] 6.14)
    l. Batch Item ([KMIP-SPEC] 6.15)
    m. Attestation Capable Indicator ([KMIP-SPEC] 6.17)
7. Supports Message Format ([KMIP-SPEC] 7)
8. Supports Authentication ([KMIP-SPEC] 8)
9. Supports the TTLV encoding ([KMIP-SPEC] 9.1)
10. Supports the transport requirements ([KMIP-SPEC] 10)
11. Supports Error Handling ([KMIP-SPEC] 11) for any supported object, attribute, or operation
12. Optionally supports any clause within [KMIP-SPEC] that is not listed above
13. Optionally supports extensions outside the scope of this standard (e.g., vendor extensions, conformance clauses) that do not contradict any KMIP requirements

5.2 Baseline Client

The Baseline Client provides some of the most basic functionality that is expected of a conformant KMIP client – the ability to request information about the server.

An implementation is a conforming Baseline Client Clause if it meets the following conditions:

1. Supports the conditions required by the KMIP Client conformance clauses ([KMIP-SPEC] 12.2)
2. Supports the following objects:
    a. Attribute ([KMIP-SPEC] 2.1.1)
    b. Template-Attribute Structure ([KMIP-SPEC] 2.1.8)
3. Supports the following subsets of attributes:
    a. Unique Identifier ([KMIP-SPEC] 3.1)
    b. Object Type ([KMIP-SPEC] 3.3)
    c. Digest ([KMIP-SPEC] 3.17)
    d. Default Operation Policy ([KMIP-SPEC] 3.18.2)
    e. State ([KMIP-SPEC] 3.22)
    f. Initial Date ([KMIP-SPEC] 3.23)
    g. Activation Date ([KMIP-SPEC] 3.24)
    h. Deactivation Date ([KMIP-SPEC] 3.27)
    i. Last Change Date ([KMIP-SPEC] 3.38)
4. Supports the ID Placeholder ([KMIP-SPEC] 4)
5. Supports the following client-to-server operations:
    a. Locate ([KMIP-SPEC] 4.9)
    b. Get ([KMIP-SPEC] 4.11)

c. Get Attributes ([KMIP-SPEC] 4.12)
d. Query ([KMIP-SPEC] 4.25)

6. Supports the following message contents:
   a. Protocol Version ([KMIP-SPEC] 6.1)
   b. Operation ([KMIP-SPEC] 6.2)
   c. Maximum Response Size ([KMIP-SPEC] 6.3)
   d. Unique Batch Item ID ([KMIP-SPEC] 6.4)
   e. Time Stamp ([KMIP-SPEC] 6.5)
   f. Asynchronous Indicator ([KMIP-SPEC] 6.7)
   g. Result Status ([KMIP-SPEC] 6.9)
   h. Result Reason ([KMIP-SPEC] 6.10)
   i. Batch Order Option ([KMIP-SPEC] 6.12)
   j. Batch Error Continuation Option ([KMIP-SPEC] 6.13)
   k. Batch Count ([KMIP-SPEC] 6.14)
   l. Batch Item ([KMIP-SPEC] 6.15)
14. Supports Message Format ([KMIP-SPEC] 7)
15. Supports Authentication ([KMIP-SPEC] 8)
16. Supports the TTLV encoding ([KMIP-SPEC] 9.1)
17. Supports the transport requirements ([KMIP-SPEC] 10)
18. Supports Error Handling ([KMIP-SPEC] 11) for any supported object, attribute, or operation
19. Optionally supports any clause within [KMIP-SPEC] that is not listed above.
20. Optionally supports extensions outside the scope of this standard (e.g., vendor extensions, conformance clauses) that do not contradict any KMIP requirements

5.3 Complete Server

The Complete Server provides functionality that is expected of a conformant KMIP server that implements the entire specification.

An implementation is a conforming Complete Server if it meets the following conditions:

1. Supports KMIP Server conformance clauses ([KMIP-SPEC] 12.1)
2. Supports Objects ([KMIP-SPEC] 2)
3. Supports Attributes ([KMIP-SPEC] 3)
4. Supports Client-to-Server operations ([KMIP-SPEC] 4)
5. Supports Server-to-Client operations ([KMIP-SPEC] 5)
6. Supports Message Contents ([KMIP-SPEC] 6)
7. Supports Message Formats ([KMIP-SPEC] 7)
8. Supports Authentication ([KMIP-SPEC] 8)
9. Supports Message Encodings ([KMIP-SPEC] 9)
10. Supports Error Handling ([KMIP-SPEC] 11)
11. Optionally supports extensions outside the scope of this standard (e.g., vendor extensions, conformance clauses) that do not contradict any KMIP requirements

Appendix A. Acknowledgments

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

Participants:

Hal Aldridge, Sypris Electronics
Mike Allen, Symantec
Gordon Arnold, IBM
Todd Arnold, IBM
Richard Austin, Hewlett-Packard
Lars Bagnert, PrimeKey
Elaine Barker, NIST
Peter Bartok, Venafi, Inc.
Tom Benjamin, IBM
Anthony Berglas, Cryptsoft
Mathias Björkqvist, IBM
Kevin Bocket, Venafi
Anne Bolgert, IBM
Alan Brown, Thales e-Security
Tim Bruce, CA Technologies
Chris Burchett, Credant Technologies, Inc.
Kelley Burgin, National Security Agency
Robert Burns, Thales e-Security
Chuck Castleton, Venafi
Kenli Chong, QuintessenceLabs
John Clark, Hewlett-Packard
Tom Clifford, Symantec Corp.
Doron Cohen, SafeNet, Inc
Tony Cox, Cryptsoft
Russell Dietz, SafeNet, Inc
Graydon Dodson, Lexmark International Inc.
Vinod Duggirala, EMC Corporation
Chris Dunn, SafeNet, Inc.
Michael Duren, Sypris Electronics
James Dzierzanowski, American Express CCoE
Faisal Faruqui, Thales e-Security
Stan Feather, Hewlett-Packard
David Finkelstein, Symantec Corp.
James Fitzgerald, SafeNet, Inc.
Indra Fitzgerald, Hewlett-Packard
Judith Furlong, EMC Corporation
Susan Gleeson, Oracle
Robert Griffin, EMC Corporation
Paul Grojean, Individual
Robert Haas, IBM
Thomas Hardjono, M.I.T.
ChengDong He, Huawei Technologies Co., Ltd.
Steve He, Vormetric
Kurt Heberlein, Hewlett-Packard
Larry Hofer, Emulex Corporation
Maryann Hondo, IBM
Walt Hubis, NetApp
Tim Hudson, Cryptsoft
Jonas Iggbom, Venafi, Inc.

Sitaram Inguva, American Express CCoE
Jay Jacobs, Target Corporation
Glen Jaquette, IBM
Mahadev Karadiguddi, NetApp
Greg Kazmierczak, Wave Systems Corp.
Marc Kenig, SafeNet, Inc.
Mark Knight, Thales e-Security
Kathy Kriese, Symantec Corporation
Mark Lambiase, SecureAuth
John Leiseboer, Quintenssence Labs
Hal Lockhart, Oracle Corporation
Robert Lockhart, Thales e-Security
Anne Luk, Cryptsoft
Sairam Manidi, Freescale
Luther Martin, Voltage Security
Neil McEvoy, iFOSSF
Marina Milshtein, Individual
Dale Moberg, Axway Software
Jishnu Mukeri, Hewlett-Packard
Bryan Olson, Hewlett-Packard
John Peck, IBM
Rob Philpott, EMC Corporation
Denis Pochuev, SafeNet, Inc.
Reid Poole, Venafi, Inc.
Ajai Puri, SafeNet, Inc.
Saravanan Ramalingam, Thales e-Security
Peter Reed, SafeNet, Inc.
Bruce Rich, IBM
Christina Richards, American Express CCoE
Warren Robbins, Dell
Peter Robinson, EMC Corporation
Scott Rotondo, Oracle
Saikat Saha, SafeNet, Inc.
Anil Saldhana, Red Hat
Subhash Sankuratripati, NetApp
Boris Schumperli, Cryptomathic
Greg Singh, QuintessenceLabs
David Smith, Venafi, Inc
Brian Spector, Certivox
Terence Spies, Voltage Security
Deborah Steckroth, RouteOne LLC
Michael Stevens, QuintessenceLabs
Marcus Streets, Thales e-Security
Satish Sundar, IBM
Kiran Thota, VMware
Somanchi Trinath, Freescale Semiconductor, Inc.
Nathan Turajski, Thales e-Security
Sean Turner, IECA, Inc.
Paul Turner, Venafi, Inc.
Rod Wideman, Quantum Corporation
Steven Wierenga, Hewlett-Packard
Jin Wong, QuintessenceLabs
Sameer Yami, Thales e-Security
Peter Yee, EMC Corporation
Krishna Yellepeddy, IBM
Catherine Ying, SafeNet, Inc.
Tatu Ylonen, SSH Communications Security (Tectia Corp)

Michael Yoder, Vormetric. Inc.
Magda Zdunkiewicz, Cryptsoft
Peter Zelechoski, Election Systems & Software

Appendix B. Revision History

Revision	Date	Editor	Changes Made
wd01	23-May-2013	Tim Hudson	Initial revision based on the KMIP 1.1 equivalent document and TC discussions
wd02	25-June-2013	Tim Hudson	Removed comments, updated participant list, included line numbers.
pr01update	11-June-2014	Tim Hudson	Updated following Public Review