

Key Management Interoperability Protocol Test Cases Version 1.4

Committee Note Draft 01 /
Public Review Draft 01

30 March 2017

Specification URIs

This version:

<http://docs.oasis-open.org/kmip/testcases/v1.4/cnprd01/kmip-testcases-v1.4-cnprd01.docx> (Authoritative)

<http://docs.oasis-open.org/kmip/testcases/v1.4/cnprd01/kmip-testcases-v1.4-cnprd01.html>

<http://docs.oasis-open.org/kmip/testcases/v1.4/cnprd01/kmip-testcases-v1.4-cnprd01.pdf>

Previous version:

N/A

Latest version:

<http://docs.oasis-open.org/kmip/testcases/v1.4/kmip-testcases-v1.4.docx> (Authoritative)

<http://docs.oasis-open.org/kmip/testcases/v1.4/kmip-testcases-v1.4.html>

<http://docs.oasis-open.org/kmip/testcases/v1.4/kmip-testcases-v1.4.pdf>

Technical Committee:

[OASIS Key Management Interoperability Protocol (KMIP) TC](https://www.oasis-open.org/committees/kmip/)

Chairs:

Tony Cox (tony.cox@cryptsoft.com), [Cryptsoft Pty Ltd.](http://cryptsoft.com/)

Saikat Saha (saikat.saha@oracle.com), [Oracle](http://www.oracle.com/)

Editors:

Tim Hudson (tjh@cryptsoft.com), [Cryptsoft Pty Ltd.](http://cryptsoft.com/)

Mark Joseph (mark@p6r.com), [P6R, Inc](http://www.p6r.com/)

Additional artifacts:

This document is one component of a Work Product that also includes:

* Test cases: <http://docs.oasis-open.org/kmip/testcases/v1.4/cnprd01/test-cases/kmip-v1.4/>.

Related work:

This specification replaces or supersedes:

* *Key Management Interoperability Protocol Test Cases Version 1.3*. Edited by Tim Hudson and Mark Joseph. Latest version. <http://docs.oasis-open.org/kmip/testcases/v1.3/kmip-testcases-v1.3.html>.

This specification is related to:

* *Key Management Interoperability Protocol Specification Version 1.4*. Edited by Tony Cox. Latest version: <http://docs.oasis-open.org/kmip/spec/v1.4/kmip-spec-v1.4.html>.
* *Key Management Interoperability Protocol Profiles Version 1.4.* Edited by Tim Hudson and Robert Lockhart. Latest version: <http://docs.oasis-open.org/kmip/profiles/v1.4/kmip-profiles-v1.4.html>.
* *Key Management Interoperability Protocol Usage Guide Version 1.4*. Edited by Judith Furlong. Latest version: <http://docs.oasis-open.org/kmip/ug/v1.4/kmip-ug-v1.4.html>.

Abstract:

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

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 (TC) members should send comments on this document to the TC’s email list. Others should send comments to the TC’s public comment list, after subscribing to it by following the instructions at the “[Send A Comment](https://www.oasis-open.org/committees/comments/index.php?wg_abbrev=kmip)” button on the TC’s web page at <https://www.oasis-open.org/committees/kmip/>.

Citation format:

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

**[kmip-testcases-v1.4]**

*Key Management Interoperability Protocol Test Cases Version 1.4.* Edited by Tim Hudson and Mark Joseph. 30 March 2017. OASIS Committee Note Draft 01 / Public Review Draft 01. <http://docs.oasis-open.org/kmip/testcases/v1.4/cnprd01/kmip-testcases-v1.4-cnprd01.html>. Latest version: <http://docs.oasis-open.org/kmip/testcases/v1.4/kmip-testcases-v1.4.html>.

Copyright © OASIS Open 2017. 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](https://www.oasis-open.org/policies-guidelines/ipr) 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.

Table of Contents

[1 Introduction 7](#_Toc479076929)

[1.1 References (non-normative) 7](#_Toc479076930)

[2 KMIP Test Cases 8](#_Toc479076931)

[2.1 KMIP 1.4 Test Cases 8](#_Toc479076932)

[2.1.1 TC-CERTATTR-1-14 8](#_Toc479076933)

[2.1.2 TC-CREG-2-14 8](#_Toc479076934)

[2.1.3 TC-CREATE-SD-1-14 9](#_Toc479076935)

[2.1.4 TC-CS-CORVAL-1-14 9](#_Toc479076936)

[2.1.5 TC-DERIVEKEY-1-10 9](#_Toc479076937)

[2.1.6 TC-DERIVEKEY-1-11 9](#_Toc479076938)

[2.1.7 TC-DERIVEKEY-1-12 9](#_Toc479076939)

[2.1.8 TC-DERIVEKEY-1-13 9](#_Toc479076940)

[2.1.9 TC-DERIVEKEY-1-14 9](#_Toc479076941)

[2.1.10 TC-DERIVEKEY-2-10 9](#_Toc479076942)

[2.1.11 TC-DERIVEKEY-2-11 10](#_Toc479076943)

[2.1.12 TC-DERIVEKEY-2-12 10](#_Toc479076944)

[2.1.13 TC-DERIVEKEY-2-13 10](#_Toc479076945)

[2.1.14 TC-DERIVEKEY-2-14 10](#_Toc479076946)

[2.1.15 TC-DERIVEKEY-3-10 10](#_Toc479076947)

[2.1.16 TC-DERIVEKEY-3-11 10](#_Toc479076948)

[2.1.17 TC-DERIVEKEY-3-12 10](#_Toc479076949)

[2.1.18 TC-DERIVEKEY-3-13 10](#_Toc479076950)

[2.1.19 TC-DERIVEKEY-3-14 10](#_Toc479076951)

[2.1.20 TC-DERIVEKEY-4-10 11](#_Toc479076952)

[2.1.21 TC-DERIVEKEY-4-11 11](#_Toc479076953)

[2.1.22 TC-DERIVEKEY-4-12 11](#_Toc479076954)

[2.1.23 TC-DERIVEKEY-4-13 11](#_Toc479076955)

[2.1.24 TC-DERIVEKEY-4-14 11](#_Toc479076956)

[2.1.25 TC-DERIVEKEY-5-10 11](#_Toc479076957)

[2.1.26 TC-DERIVEKEY-5-11 11](#_Toc479076958)

[2.1.27 TC-DERIVEKEY-5-12 11](#_Toc479076959)

[2.1.28 TC-DERIVEKEY-5-13 11](#_Toc479076960)

[2.1.29 TC-DERIVEKEY-5-14 12](#_Toc479076961)

[2.1.30 TC-DERIVEKEY-6-14 12](#_Toc479076962)

[2.1.31 TC-ECC-1-14 12](#_Toc479076963)

[2.1.32 TC-ECC-2-14 12](#_Toc479076964)

[2.1.33 TC-ECC-3-14 12](#_Toc479076965)

[2.1.34 TC-ECDSA-SIGN-DIGESTEDDATA 1-14 12](#_Toc479076966)

[2.1.35 TC-ECDSA-SIGN-1-14 12](#_Toc479076967)

[2.1.36 TC-I18N-1-10 12](#_Toc479076968)

[2.1.37 TC-I18N-3-10 13](#_Toc479076969)

[2.1.38 TC-I18N-1-11 13](#_Toc479076970)

[2.1.39 TC-I18N-3-11 13](#_Toc479076971)

[2.1.40 TC-I18N-1-12 13](#_Toc479076972)

[2.1.41 TC-I18N-2-12 13](#_Toc479076973)

[2.1.42 TC-I18N-3-12 13](#_Toc479076974)

[2.1.43 TC-I18N-1-13 14](#_Toc479076975)

[2.1.44 TC-I18N-2-13 14](#_Toc479076976)

[2.1.45 TC-I18N-3-13 14](#_Toc479076977)

[2.1.46 TC-I18N-1-14 14](#_Toc479076978)

[2.1.47 TC-I18N-2-14 14](#_Toc479076979)

[2.1.48 TC-I18N-3-14 14](#_Toc479076980)

[2.1.49 TC-MDO-1-14 14](#_Toc479076981)

[2.1.50 TC-MDO-2-14 15](#_Toc479076982)

[2.1.51 TC-MDO-3-14 15](#_Toc479076983)

[2.1.52 TC-NP-1-14 15](#_Toc479076984)

[2.1.53 TC-NP-2-14 15](#_Toc479076985)

[2.1.54 TC-OFFSET-1-14 15](#_Toc479076986)

[2.1.55 TC-OFFSET-2-14 15](#_Toc479076987)

[2.1.56 TC-OTP-1-14 15](#_Toc479076988)

[2.1.57 TC-OTP-2-14 16](#_Toc479076989)

[2.1.58 TC-OTP-3-14 16](#_Toc479076990)

[2.1.59 TC-OTP-4-14 16](#_Toc479076991)

[2.1.60 TC-OTP-5-14 16](#_Toc479076992)

[2.1.61 TC-PGP-1-14 16](#_Toc479076993)

[2.1.62 TC-PKCS12-1-14 16](#_Toc479076994)

[2.1.63 TC-PKCS12-2-14 17](#_Toc479076995)

[2.1.64 TC-Q-CAP-1-14 17](#_Toc479076996)

[2.1.65 TC-Q-CAP-2-14 17](#_Toc479076997)

[2.1.66 TC-Q-CAP-3-14 17](#_Toc479076998)

[2.1.67 TC-Q-CREG-1-14 17](#_Toc479076999)

[2.1.68 TC-Q-PROF-1-14 17](#_Toc479077000)

[2.1.69 TC-Q-PROF-2-14 17](#_Toc479077001)

[2.1.70 TC-Q-PROF-3-14 17](#_Toc479077002)

[2.1.71 TC-Q-RNGS-1-14 18](#_Toc479077003)

[2.1.72 TC-Q-RNGS-2-14 18](#_Toc479077004)

[2.1.73 TC-Q-RNGS-3-14 18](#_Toc479077005)

[2.1.74 TC-Q-RNGS-4-14 18](#_Toc479077006)

[2.1.75 TC-Q-RNGS-5-14 18](#_Toc479077007)

[2.1.76 TC-Q-RNGS-6-14 18](#_Toc479077008)

[2.1.77 TC-Q-S2C-1-14 18](#_Toc479077009)

[2.1.78 TC-Q-S2C-2-14 19](#_Toc479077010)

[2.1.79 TC-Q-S2C-PROF-1-14 19](#_Toc479077011)

[2.1.80 TC-Q-S2C-PROF-2-14 19](#_Toc479077012)

[2.1.81 TC-Q-VAL-1-14 19](#_Toc479077013)

[2.1.82 TC-Q-VAL-2-14 19](#_Toc479077014)

[2.1.83 TC-REKEY-1-10 19](#_Toc479077015)

[2.1.84 TC-REKEY-1-11 19](#_Toc479077016)

[2.1.85 TC-REKEY-1-12 19](#_Toc479077017)

[2.1.86 TC-REKEY-1-13 19](#_Toc479077018)

[2.1.87 TC-REKEY-1-14 20](#_Toc479077019)

[2.1.88 TC-RNG-ATTR-1-14 20](#_Toc479077020)

[2.1.89 TC-RNG-ATTR-2-14 20](#_Toc479077021)

[2.1.90 TC-RSA-SIGN-DIGESTEDDATA 1-14 20](#_Toc479077022)

[2.1.91 TC-SJ-1-14 20](#_Toc479077023)

[2.1.92 TC-SJ-2-14 20](#_Toc479077024)

[2.1.93 TC-SJ-3-14 20](#_Toc479077025)

[2.1.94 TC-SJ-4-14 20](#_Toc479077026)

[2.1.95 TC-STREAM-ENC-1-14 21](#_Toc479077027)

[2.1.96 TC-STREAM-ENC-2-14 21](#_Toc479077028)

[2.1.97 TC-STREAM-ENCDEC-1-14 21](#_Toc479077029)

[2.1.98 TC-STREAM-HASH-1-14 21](#_Toc479077030)

[2.1.99 TC-STREAM-HASH-2-14 21](#_Toc479077031)

[2.1.100 TC-STREAM-HASH-3-14 21](#_Toc479077032)

[2.1.101 TC-STREAM-SIGN-1-14 21](#_Toc479077033)

[2.1.102 TC-STREAM-SIGNVFY-1-14 22](#_Toc479077034)

[2.1.103 TC-WRAP-1-14 22](#_Toc479077035)

[2.1.104 TC-WRAP-2-14 22](#_Toc479077036)

[2.1.105 TC-WRAP-3-14 22](#_Toc479077037)

[2.1.106 TC-SENSITIVE-1-14 22](#_Toc479077038)

[2.1.107 TC-EXTRACTABLE-1-14 22](#_Toc479077039)

[2.1.108 TC-STREAM-MAC-1-14 22](#_Toc479077040)

[3 KMIP Test Cases Setup 23](#_Toc479077041)

[3.1 KMIP 1.4 Test Cases Setup 23](#_Toc479077042)

[3.1.1 TC-CREG-1-14 23](#_Toc479077043)

[3.1.2 TC-CREG-3-14 23](#_Toc479077044)

[Appendix A. Acknowledgments 24](#_Toc479077045)

[Appendix B. Revision History 26](#_Toc479077046)

1. Introduction

The purpose of this document is to describe test cases to demonstrate the Key Management Interoperability Protocol (KMIP) [KMIP-SPEC]. The test cases illustrate that the concepts within the protocol are sound and how the protocol may be used when implementing KMIP in applications. These test cases are not intended to fully test an implementation of KMIP.

## References (non-normative)

[KMIP-SPEC]

*Key Management Interoperability Protocol Specification Version 1.4*. Edited by Tony Cox and Charles White. Latest version: <http://docs.oasis-open.org/kmip/spec/v1.4/kmip-spec-v1.4.html>.

[KMIP-PROFILES]

*Key Management Interoperability Protocol Profiles Version 1.4.* Edited by Tim Hudson and Robert Lockhart. Latest version: <http://docs.oasis-open.org/kmip/profiles/v1.4/kmip-profiles-v1.4.html>.

[XML]

*XML 1.0 Recommendation,* T. Bray, J. Paoli, M. Sperberg-McQueen, Editors, W3C Recommendation, February 10, 1998, <http://www.w3.org/TR/1998/REC-xml-19980210>. Latest version available at <http://www.w3.org/TR/REC-xml>.

1. KMIP 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-PROFILES] intended to be both human-readable and usable by automated tools.

Each test case has a unique label (the section name) which the protocol version 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.

The test cases show one possible way to construct the messages, and the messages shown are not necessarily the only conformant constructions as many items within KMIP are optional and server behavior depends on the server's policy. Support for a test case is predicated on a server matching the test case assumptions and the behavior shown in the request-response pairs.

Symbolic identifiers are of the form $UPPERCASE\_NAME followed by optional unique index value. Wherever a symbolic identifier occurs in a test cases the implementation must replace it with a reasonable appearing datum of the expected type. Time values can be specified in terms of an offset from the current time in seconds of the form $NOW or $NOW-n or $NOW+n.

## KMIP 1.4 Test Cases

### TC-CERTATTR-1-14

A client registers a certificate and the server creates the certificate attributes based on the subject and issuer distinguished name values.

See [test-cases/kmip-v1.4/TC-CERTATTR-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-CERTATTR-1-14.xml)

### TC-CREG-2-14

Assuming that a KMIP server has set up a keypair and corresponding certificate (or will generate these on-the-fly) for a given one time credential (username and password or OTP value) return in a single request the public key, private key, and corresponding certificate for use in subsequent connections.

How the server creates the keypair and certificate is outside of the scope of KMIP (it MAY be performed via KMIP operations or via an entirely separate non-KMIP approach).

It is assumed that the server implements an appropriate policy to only accept the client provided credential once with a time limit on how soon the credential remains valid and that the public key, private key, and certificate will only be returned once. The server may elect to keep, archive, or destroy the managed objects after the client has completed this request.

See <test-cases/kmip-v1.4/TC-CREG-2-14.xml>

### TC-CREATE-SD-1-14

A client requests a server to create a secret data managed object.

See [test-cases/kmip-v1.4/TC-CREATE-SD-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-CREATE-SD-1-14.xml)

### TC-CS-CORVAL-1-14

A client sets a client correlation value and the server also responds with a server correlation value.

See [test-cases/kmip-v1.4/TC-CS-CORVAL-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-CS-CORVAL-1-14.xml)

### TC-DERIVEKEY-1-10

A client uses Derive Key using SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-1-10.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-1-10.xml)

### TC-DERIVEKEY-1-11

A client uses Derive Key using SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-1-11.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-1-11.xml)

### TC-DERIVEKEY-1-12

A client uses Derive Key using SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-1-12.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-1-12.xml)

### TC-DERIVEKEY-1-13

A client uses Derive Key using SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-1-13.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-1-13.xml)

### TC-DERIVEKEY-1-14

A client uses Derive Key using SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-1-14.xml)

### TC-DERIVEKEY-2-10

A client uses Derive Key using HMAC-SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-2-10.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-2-10.xml)

### TC-DERIVEKEY-2-11

A client uses Derive Key using HMAC-SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-2-11.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-2-11.xml)

### TC-DERIVEKEY-2-12

A client uses Derive Key using HMAC-SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-2-12.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-2-12.xml)

### TC-DERIVEKEY-2-13

A client uses Derive Key using HMAC-SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-2-13.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-2-13.xml)

### TC-DERIVEKEY-2-14

A client uses Derive Key using HMAC-SHA\_256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-2-14.xml)

### TC-DERIVEKEY-3-10

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-3-10.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-3-10.xml)

### TC-DERIVEKEY-3-11

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-3-11.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-3-11.xml)

### TC-DERIVEKEY-3-12

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-3-12.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-3-12.xml)

### TC-DERIVEKEY-3-13

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-3-13.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-3-13.xml)

### TC-DERIVEKEY-3-14

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-3-14.xml)

### TC-DERIVEKEY-4-10

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-4-10.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-4-10.xml)

### TC-DERIVEKEY-4-11

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-4-11.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-4-11.xml)

### TC-DERIVEKEY-4-12

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-4-12.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-4-12.xml)

### TC-DERIVEKEY-4-13

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-4-13.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-4-13.xml)

### TC-DERIVEKEY-4-14

A client uses Derive Key using PBKDF2.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-4-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-4-14.xml)

### TC-DERIVEKEY-5-10

A client uses Derive Key using PBKDF2 and SHA-256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-5-10.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-5-10.xml)

### TC-DERIVEKEY-5-11

A client uses Derive Key using PBKDF2 and SHA-256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-5-11.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-5-14.xml)

### TC-DERIVEKEY-5-12

A client uses Derive Key using PBKDF2 and SHA-256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-5-12.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-5-12.xml)

### TC-DERIVEKEY-5-13

A client uses Derive Key using PBKDF2 and SHA-256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-5-13.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-5-13.xml)

### TC-DERIVEKEY-5-14

A client uses Derive Key using PBKDF2 and SHA-256.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-5-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-5-14.xml)

### TC-DERIVEKEY-6-14

A client uses Derive Key using ASYMMETRIC\_KEY and ECDH.

See [test-cases/kmip-v1.4/TC-DERIVEKEY-6-14.xml](test-cases%5Ckmip-v1.4%5CTC-DERIVEKEY-6-14.xml)

### TC-ECC-1-14

A client registers and EC private key in ECPrivateKey format and EC public key in X.509 format using the EC cryptographic algorithm.

See [test-cases/kmip-v1.4/TC-ECC-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-ECC-1-14.xml)

### TC-ECC-2-14

A client registers and EC private key in PKCS8 format and EC public key in X.509 format using the EC cryptographic algorithm.

See [test-cases/kmip-v1.4/TC-ECC-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-ECC-2-14.xml)

### TC-ECC-3-14

A client registers and EC private key in ECPrivateKey format and EC public key in X.509 format using the EC cryptographic algorithm.

See [test-cases/kmip-v1.4/TC-ECC-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-ECC-3-14.xml)

### TC-ECDSA-SIGN-DIGESTEDDATA 1-14

ECDSA Signing with the digested data provided by the client.

See [test-cases/kmip-v1.4/TC-ECDSA-SIGN-DIGESTEDDATA-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-ECDSA-SIGN-DIGESTEDDATA-1-14.xml)

### TC-ECDSA-SIGN-1-14

A client registers and EC private key in ECPrivateKey format and EC public key in X.509 format using the EC cryptographic algorithm and performs a Sign operation followed by a Signature Verify operation.

See [test-cases/kmip-v1.4/TC-ECDSA-SIGN-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-ECDSA-SIGN-1-14.xml)

### TC-I18N-1-10

Client provides a key name containing a Greek capital Alpha.

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-1-10.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-1-10.xml)

### TC-I18N-3-10

Client provides a customer attribute containing a Greek capital Alpha with the attribute value containing a Greek capital Omega

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.See [test-cases/kmip-v1.4/TC-I18N-3-10.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-3-10.xml)

### TC-I18N-1-11

Client provides a key name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-1-11.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-1-11.xml)

### TC-I18N-3-11

Client provides a customer attribute containing a Greek capital Alpha with the attribute value containing a Greek capital Omega

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-3-11.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-3-11.xml)

### TC-I18N-1-12

Client provides a key name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-1-12.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-1-12.xml)

### TC-I18N-2-12

Client provides a key alternative name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-2-12.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-2-12.xml)

### TC-I18N-3-12

Client provides a customer attribute containing a Greek capital Alpha with the attribute value containing a Greek capital Omega

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-3-12.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-3-12.xml)

### TC-I18N-1-13

Client provides a key name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-1-13.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-1-13.xml)

### TC-I18N-2-13

Client provides a key alternative name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-2-13.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-2-13.xml)

### TC-I18N-3-13

Client provides a customer attribute containing a Greek capital Alpha with the attribute value containing a Greek capital Omega

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-3-13.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-3-13.xml)

### TC-I18N-1-14

Client provides a key name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-1-14.xml)

### TC-I18N-2-14

Client provides a key alternative name containing a Greek capital Alpha

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-2-14.xml)

### TC-I18N-3-14

Client provides a customer attribute containing a Greek capital Alpha with the attribute value containing a Greek capital Omega

Note: the encoding in XML has to be correctly converted into the valid UTF-8 format.

See [test-cases/kmip-v1.4/TC-I18N-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-I18N-3-14.xml)

### TC-MDO-1-14

A client requests a meta-data-only object (no key material).

See [test-cases/kmip-v1.4/TC-MDO-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-MDO-1-14.xml)

### TC-MDO-2-14

A client requests a meta-data-only object (no key material) and an object with key material and performs Locate that only returns the meta-data-only object.

See [test-cases/kmip-v1.4/TC-MDO-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-MDO-2-14.xml)

### TC-MDO-3-14

A client requests a meta-data-only object (no key material) using the URL format of the Key Value Location and performs Locate.

See [test-cases/kmip-v1.4/TC-MDO-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-MDO-3-14.xml)

### TC-NP-1-14

A client performs a create request triggering the server sending a Put message to the client.

See [test-cases/kmip-v1.4/TC-NP-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-NP-1-14.xml)

### TC-NP-2-14

A client performs a register request followed by an add attribute operation triggering the server sending a Put message and a Notify to the client.

See [test-cases/kmip-v1.4/TC-NP-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-NP-2-14.xml)

### TC-OFFSET-1-14

A client requests the server creates a number of symmetric keys and then uses the Offset parameter in Locate to return various items.

See [test-cases/kmip-v1.4/TC-OFFSET-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-OFFSET-1-14.xml)

### TC-OFFSET-2-14

A client requests the server creates a number of symmetric keys and then uses the Offset parameter in Locate to return various items.

See [test-cases/kmip-v1.4/TC-OFFSET-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-OFFSET-2-14.xml)

### TC-OTP-1-14

One-Time-Pad encryption - assuming pad has been setup

How the server sets up and operates the one time pad is outside of the scope of KMIP - this is just an example usage for testing the encrypt/decrypt mechanism.

A KMIP server can implement handling of the one-time-pad material via whatever approach makes sense in the context of a specific server implementation - all that is required is that both servers involved are in agreement about the one-time-pad.

See [test-cases/kmip-v1.4/TC-OTP-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-OTP-1-14.xml)

### TC-OTP-2-14

One-Time-Pad decryption - assuming pad has been setup

How the server sets up and operates the one time pad is outside of the scope of KMIP - this is just an example usage for testing the encrypt/decrypt mechanism.

A KMIP server can implement handling of the one-time-pad material via whatever approach makes sense in the context of a specific server implementation - all that is required is that both servers involved are in agreement about the one-time-pad.

See [test-cases/kmip-v1.4/TC-OTP-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-OTP-2-14.xml)

### TC-OTP-3-14

One-Time-Pad attempted get - assuming pad has been setup

Note: this example shows a server configured to return a Get without the key material present.

See [test-cases/kmip-v1.4/TC-OTP-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-OTP-3-14.xml)

### TC-OTP-4-14

One-Time-Pad attempted get - assuming pad has been setup

Note: this example shows a server configured to return denied for a Get request; the key material is never returned to the client in this configuration.

See [test-cases/kmip-v1.4/TC-OTP-4-14.xml](test-cases%5Ckmip-v1.4%5CTC-OTP-4-14.xml)

### TC-OTP-5-14

One-Time-Pad attempted get - assuming pad has been setup and supports multiple encrypt and decrypt operations.

See [test-cases/kmip-v1.4/TC-OTP-5-14.xml](test-cases%5Ckmip-v1.4%5CTC-OTP-5-14.xml)

### TC-PGP-1-14

Register a PGP public key block and private key block and add appropriate links between the managed objects.

See [test-cases/kmip-v1.4/TC-PGP-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-PGP-1-14.xml)

### TC-PKCS12-1-14

Register objects and then performs a Get returning in PKCS#12 format

See [test-cases/kmip-v1.4/TC-PKCS-12-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-PKCS12-1-14.xml)

### TC-PKCS12-2-14

Register objects in PKCS#12 format and then performs a Get returning the individual objects.

See [test-cases/kmip-v1.4/TC-PKCS-12-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-PKCS12-2-14.xml)

### TC-Q-CAP-1-14

Return a list of responses indicating the server does not want to provide details as to its specific capabilities.

See [test-cases/kmip-v1.4/TC-Q-CAP-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-CAP-1-14.xml)

### TC-Q-CAP-2-14

Return a list of responses indicating the server simply deletes key material on destroy.

See [test-cases/kmip-v1.4/TC-Q-CAP-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-CAP-2-14.xml)

### TC-Q-CAP-3-14

Return a list of responses indicating the server simply deletes key material on destroy.

See [test-cases/kmip-v1.4/TC-Q-CAP-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-CAP-3-14.xml)

### TC-Q-CREG-1-14

Return the list of client registration methods supported by a server. This example shows all four approaches are supported.

See [test-cases/kmip-v1.4/TC-Q-CREG-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-CREG-1-14.xml)

### TC-Q-PROF-1-14

Return details of the server claimed supported profiles.

See [test-cases/kmip-v1.4/TC-Q-PROF-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-PROF-1-14.xml)

### TC-Q-PROF-2-14

Return details of the server claimed supported profiles. This example shows a server claiming to support all profiles.

See [test-cases/kmip-v1.4/TC-Q-PROF-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-PROF-2-14.xml)

### TC-Q-PROF-3-14

 Return details of the server claimed supported profiles. This example shows a server returning Server URI and Port values for HTTPS usage

See [test-cases/kmip-v1.4/TC-Q-PROF-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-PROF-3-14.xml)

### TC-Q-RNGS-1-14

Return details of the supported RNGs where the server provides no actual information about the RNG (i.e. nothing is claimed).

See [test-cases/kmip-v1.4/TC-Q-RNGS-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-1-14.xml)

### TC-Q-RNGS-2-14

Return details of the supported RNGs where the server provides details of an ANSI X9.31 AES-256 based RNG. (e.g. RNGVAL 1202)

See [test-cases/kmip-v1.4/TC-Q-RNGS-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-2-14.xml)

### TC-Q-RNGS-3-14

Return details of the supported RNGs where the server provides details of an FIPS 186-2 x-Chagne Notice SHA-1 based RNG. (e.g. RNGVAL 1203)

See [test-cases/kmip-v1.4/TC-Q-RNGS-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-3-14.xml)

### TC-Q-RNGS-4-14

Return details of the supported RNGs where the server provides details of a DRBG HMAC based HMAC-SHA256 with prediction resistance RNG and a DRBG HMAC based HMAC-SHA1 with prediction resistance RNG and a DRBG Hash based SHA256 with prediction resistance RNG. (e.g. DRBGVAL 540)

See [test-cases/kmip-v1.4/TC-Q-RNGS-4-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-4-14.xml)

### TC-Q-RNGS-5-14

Return details of the supported RNGs where the server provides details of a DRBG Dual-EC based SHA-256 P-256 with prediction resistance RNG. (e.g. DRBGVAL 480)

See [test-cases/kmip-v1.4/TC-Q-RNGS-5-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-5-14.xml)

### TC-Q-RNGS-6-14

Return details of the supported RNGs where the server provides details of use of a plain AES-based DRBG

See [test-cases/kmip-v1.4/TC-Q-RNGS-6-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-RNGS-6-14.xml)

### TC-Q-S2C-1-14

Server to Client Server queries the client's capabilities Client returns what it supports and may elect to use on the client to server link. This example is for a client supporting only the required operations and object types in the Tape Library Profile.

See [test-cases/kmip-v1.4/TC-Q-S2C-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-S2C-1-14.xml)

### TC-Q-S2C-2-14

Server to Client Server queries what KMIP protocol versions it supports Client returns the protocol versions it may use on the client to server link.

See [test-cases/kmip-v1.4/TC-Q-S2C-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-S2C-2-14.xml)

### TC-Q-S2C-PROF-1-14

Return details of the client claimed supported profiles. This is server-to-client request. Client returns the profiles it may use on the client to server link.

See [test-cases/kmip-v1.4/TC-Q-S2C-PROF-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-S2C-PROF-1-14.xml)

### TC-Q-S2C-PROF-2-14

Return details of the client claimed supported profiles. This is server-to-client request. Client returns the profiles it may use on the client to server link.

See [test-cases/kmip-v1.4/TC-Q-S2C-PROF-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-S2C-PROF-2-14.xml)

### TC-Q-VAL-1-14

Return details of the server claimed validation information. Example is for NIST CMVP FIPS140-2

See [test-cases/kmip-v1.4/TC-Q-VAL-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-VAL-1-14.xml)

### TC-Q-VAL-2-14

Return details of the server that does not claim any validations.

See [test-cases/kmip-v1.4/TC-Q-VAL-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-Q-VAL-2-14.xml)

### TC-REKEY-1-10

Create a key and perform multiple rekey operations.

See [test-cases/kmip-v1.4/TC-REKEY-1-10.xml](test-cases%5Ckmip-v1.4%5CTC-REKEY-1-10.xml)

### TC-REKEY-1-11

Create a key and perform multiple rekey operations.

See [test-cases/kmip-v1.4/TC-REKEY-1-11.xml](test-cases%5Ckmip-v1.4%5CTC-REKEY-1-11.xml)

### TC-REKEY-1-12

Create a key and perform multiple rekey operations.

See [test-cases/kmip-v1.4/TC-REKEY-1-12.xml](test-cases%5Ckmip-v1.4%5CTC-REKEY-1-12.xml)

### TC-REKEY-1-13

Create a key and perform multiple rekey operations.

See [test-cases/kmip-v1.4/TC-REKEY-1-13.xml](test-cases%5Ckmip-v1.4%5CTC-REKEY-1-13.xml)

### TC-REKEY-1-14

Create a key and perform multiple rekey operations.

See [test-cases/kmip-v1.4/TC-REKEY-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-REKEY-1-14.xml)

### TC-RNG-ATTR-1-14

A client registers a symmetric key including details of the RNG that the client is claiming was used to generate the symmetric key.

See [test-cases\kmip-v1.4\TC-RNG-ATTR-1-14.xml](test-cases/kmip-v1.4/TC-RNG-ATTR-1-14.xml)

### TC-RNG-ATTR-2-14

A client requests the server creates a symmetric key and it does and also includes the required details of the RNG that was used to generate the symmetric key.

See [test-cases\kmip-v1.4\TC-RNG-ATTR-2-14.xml](test-cases/kmip-v1.4/TC-RNG-ATTR-2-14.xml)

### TC-RSA-SIGN-DIGESTEDDATA 1-14

RSA Signing with the digested data provided by the client.

See [test-cases/kmip-v1.4/TC-RSA-SIGN-DIGESTEDDATA-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-RSA-SIGN-DIGESTEDDATA-1-14.xml)

### TC-SJ-1-14

Create a symmetric key and perform split and join in various combinations.

See [test-cases/kmip-v1.4/TC-SJ-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-SJ-1-14.xml)

### TC-SJ-2-14

Register a symmetric key and perform split and join in various combinations.

See [test-cases/kmip-v1.4/TC-SJ-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-SJ-2-14.xml)

### TC-SJ-3-14

Register split keys and perform join in various combinations.

See [test-cases/kmip-v1.4/TC-SJ-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-SJ-2-14.xml)

### TC-SJ-4-14

Create a symmetric key and perform split and join in various combinations using the XOR method.

See [test-cases/kmip-v1.4/TC-SJ-4-14.xml](test-cases%5Ckmip-v1.4%5CTC-SJ-4-14.xml)

### TC-STREAM-ENC-1-14

Create a symmetric key and perform encrypt with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-ENC-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-ENC-1-14.xml)

### TC-STREAM-ENC-2-14

Register a symmetric key and perform encrypt and decrypt with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-ENC-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-ENC-2-14.xml)

### TC-STREAM-ENCDEC-1-14

Register a symmetric key and perform encrypt with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-ENCDEC-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-ENCDEC-1-14.xml)

### TC-STREAM-HASH-1-14

Hash operation for data 'abc' in a single request followed immediately by a streaming equivalent for which the result must be identical.

Note: - test vector data from http://csrc.nist.gov/groups/ST/toolkit/documents/Examples/SHA\_All.pdf

See [test-cases/kmip-v1.4/TC-STREAM-HASH-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-HASH-1-14.xml)

### TC-STREAM-HASH-2-14

Hash operation for data 'abc' in a single request followed immediately by a streaming equivalent for which the result must be identical.

Note: - test vector data from http://csrc.nist.gov/groups/ST/toolkit/documents/Examples/SHA\_All.pdf

See [test-cases/kmip-v1.4/TC-STREAM-HASH-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-HASH-2-14.xml)

### TC-STREAM-HASH-3-14

Hash operation for data 'abc' in a single request followed immediately by a streaming equivalent for which the result must be identical.

Note: - test vector data from http://csrc.nist.gov/groups/ST/toolkit/documents/Examples/SHA\_All.pdf

See [test-cases/kmip-v1.4/TC-STREAM-HASH-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-HASH-3-14.xml)

### TC-STREAM-SIGN-1-14

Sign with a known asymmetric key with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-SIGN-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-SIGN-1-14.xml)

### TC-STREAM-SIGNVFY-1-14

Sign and Signature Verify with a known asymmetric key with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-SIGNVFY-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-SIGNVFY-1-14.xml)

### TC-WRAP-1-14

Show usage of Key Wrap Type As Registered.

See [test-cases/kmip-v1.4/TC-WRAP-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-WRAP-1-14.xml)

### TC-WRAP-2-14

Show usage of Key Wrap Type Not Wrapped.

See [test-cases/kmip-v1.4/TC-WRAP-2-14.xml](test-cases%5Ckmip-v1.4%5CTC-WRAP-2-14.xml)

### TC-WRAP-3-14

Show usage of returning wrapped key wrapped with a different wrapping key.

See [test-cases/kmip-v1.4/TC-WRAP-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-WRAP-3-14.xml)

### TC-SENSITIVE-1-14

Show usage of Sensitive and Always Sensitive

See [test-cases/kmip-v1.4/TC-SENSITIVE-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-SENSITIVE-1-14.xml)

### TC-EXTRACTABLE-1-14

Show usage of Extractable and Never Extractable

See [test-cases/kmip-v1.4/TC-EXTRACTABLE-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-EXTRACTABLE-1-14.xml)

### TC-STREAM-MAC-1-14

MAC and MACVerify with streaming.

See [test-cases/kmip-v1.4/TC-STREAM-MAC-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-STREAM-MAC-1-14.xml)

1. KMIP Test Cases Setup

The test cases defined in the previous section all operate independent and assume that the other end of the KMIP connection has been configured to match the assumptions in the test case.

The following scripts allow for setting up the pre-conditions for a number of the test cases and for cleaning up after the test cases have executed – via KMIP operations. A server is not required to use KMIP or to use these scripts for this purpose – they are provided simply because they are useful for some implementations.

## KMIP 1.4 Test Cases Setup

### TC-CREG-1-14

This is used to set up the test data used in the client registration example. How the server sets up this in a normal context is outside of the scope of KMIP - this is just an example usage with configuration via KMIP with a pre-generated keypair and corresponding certificate.

A KMIP server can implement the equivalent capability via whatever approach makes sense in the context of a specific server implementation.

Register a public/private key pair in the PKCS\_1 key format and a corresponding X509 certificate. Add the appropriate links between the registered objects.

See [test-cases/kmip-v1.4/TC-CREG-1-14.xml](test-cases%5Ckmip-v1.4%5CTC-CREG-1-14.xml)

### TC-CREG-3-14

This is used to clean up the test data used in the client registration example. How the server sets up this in a normal context is outside of the scope of KMIP - this is just an example usage with configuration via KMIP with a pre-generated keypair and corresponding certificate.

A KMIP server can implement the equivalent capability via whatever approach makes sense in the context of a specific server implementation.

See [test-cases/kmip-v1.4/TC-CREG-3-14.xml](test-cases%5Ckmip-v1.4%5CTC-CREG-3-14.xml)

1. Acknowledgments

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

**Editors of the previous versions of this document:**

Mathias Björkqvist, [IBM](http://www.ibm.com) (v1.0 and v1.1)

Tim Hudson, [Cryptsoft](http://www.cryptsoft.com) (v1.1, v1.2 and v1.3)

René Pawlitzek, [IBM](http://www.ibm.com) (v1.0)

**Technical Committee Participants:**

Anthony Berglas, Cryptsoft

Justin Corlett, Cryptsoft

Tony Cox, Cryptsoft

Tim Hudson, Cryptsoft

Bruce Rich, Cryptsoft

Greg Scott, Cryptsoft

Magda Zdunkiewicz, Cryptsoft

Judith Furlong, Dell

Michael Phillips, Dell

Lina Baquero, Fornetix

Jeff Bartell, Fornetix

Stephen Edwards, Fornetix

Gary Gardner, Fornetix

Heather Stevens, Fornetix

Gerald Stueve, Fornetix

Charles White, Fornetix

Alex Downey, Futurex

Hannah Lee, Hancom Secure, Inc.

Indra Fitzgerald, Hewlett Packard Enterprise (HPE)

Christopher Hillier, Hewlett Packard Enterprise (HPE)

Matt Suh, Hewlett Packard Enterprise (HPE)

Nathan Turajski, Hewlett Packard Enterprise (HPE)

Steve Wierenga, Hewlett Packard Enterprise (HPE)

Rinkesh Bansal, IBM

Mathias Bjorkqvist, IBM

Kevin Driver, IBM

Prashant Mestri, IBM

Krishna Yellepeddy, IBM

Andre Bereza, KRYPTUS

Tim Chevalier, NetApp

Hai-May Chao, Oracle

Valerie Fenwick, Oracle

Susan Gleeson, Oracle

Hal Lockhart, Oracle

Saikat Saha, Oracle

Radhika Siravara, Oracle

Mark Joseph, P6R, Inc

Jim Susoy, P6R, Inc

John Leiseboer, QuintessenceLabs Pty Ltd.

David Featherstone, SafeNet, Inc.

Joseph Brand, Semper Fortis Solutions

Chris Skiscim, Semper Fortis Solutions

Kathy Kriese, Symantec Corp.

Robert Lockhart, Thales e-Security

Steve He, Vormetric, Inc.

Peter Tsai, Vormetric, Inc.

Joshua Zhu, Vormetric, Inc.

1. Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Editor** | **Changes Made** |
| **wd04** | **23-Mar-2017** | **Tim Hudson** | **Incorporated feedback from RSA 2017 interop testing and test cases fixes from interop. Added Sensitive and Extractable test cases.** |
| wd03 | 28-Nov-2016 | Tim Hudson / Mark Joseph | Corrected errors and added additional test cases for digested data and internationalization. |
| wd02 | 19-Nov-2016 | Tim Hudson | Corrected typographical errors and added test case for create secret data |
| wd01 | 17-Nov-2016 | Tim Hudson | Initial draft. |