This is a Non-Standards Track Work Product. The patent provisions of the OASIS IPR Policy do not apply.

Key Management Interoperability Protocol Test Cases Version 1.3

Committee Note Draft 01

07 April 2016

Specification URIs

This version:

http://docs.oasis-open.org/kmip/testcases/v1.3/cnd01/kmip-testcases-v1.3-cnd01.docx (Authoritative)

http://docs.oasis-open.org/kmip/testcases/v1.3/cnd01/kmip-testcases-v1.3-cnd01.html

http://docs.oasis-open.org/kmip/testcases/v1.3/cnd01/kmip-testcases-v1.3-cnd01.pdf

Previous version:

N/A

Latest version:

http://docs.oasis-open.org/kmip/testcases/v1.3/kmip-testcases-v1.3.docx (Authoritative)

http://docs.oasis-open.org/kmip/testcases/v1.3/kmip-testcases-v1.3.html

http://docs.oasis-open.org/kmip/testcases/v1.3/kmip-testcases-v1.3.pdf

Technical Committee:

OASIS Key Management Interoperability Protocol (KMIP) TC

Chairs:

Tony Cox (tony.cox@cryptsoft.com), Cryptsoft Pty Ltd.

Saikat Saha (saikat.saha@oracle.com), Oracle

Editors:

Tim Hudson (tjh@cryptsoft.com), Cryptsoft Pty Ltd.

Mark Joseph (mark@p6r.com), P6R, Inc

Additional artifacts:

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

- Test cases: http://docs.oasis-open.org/kmip/testcases/v1.3/cnd01/test-cases/kmip-v1.3/

Related work:

This document replaces or supersedes:

- *Key Management Interoperability Protocol Test Cases Version 1.1*. Edited by Mathias Björkqvist and Tim Hudson. 27 January 2012. OASIS Committee Note 01. http://docs.oasis-open.org/kmip/testcases/v1.1/kmip-testcases-v1.1.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.

This document is related to:

- *Key Management Interoperability Protocol Specification Version 1.3*. Edited by Kiran Thota and Tony Cox. Latest version: http://docs.oasis-open.org/kmip/spec/v1.3/kmip-spec-v1.3.html.
- *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 Usage Guide Version 1.3*. Edited by Judith Furlong. Latest version: http://docs.oasis-open.org/kmip/ug/v1.3/kmip-ug-v1.3.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" 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-v1.3]

Key Management Interoperability Protocol Test Cases Version 1.3. Edited by Tim Hudson and Mark Joseph. 07 April 2016. OASIS Committee Note Draft 01. http://docs.oasis-open.org/kmip/testcases/v1.3/cnd01/kmip-testcases-v1.3-cnd01.html. Latest version: http://docs.oasis-open.org/kmip/testcases/v1.3/kmip-testcases-v1.3.html.

Table of Contents

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.

1.1 References (non-normative)

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

[KMIP-PROFILES]
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.

[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.

2 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.

2.1 KMIP 1.3 Test Cases

2.1.1 TC-CREG-2-13

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-v1.3/TC-CREG-2-13.xml]

2.1.2 TC-OFFSET-1-13

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-v1.3/TC-OFFSET-1-13.xml

2.1.3 TC-OFFSET-2-13

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-v1.3/TC-OFFSET-2-13.xml

2.1.4 TC-OTP-1-13

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-v1.3/TC-OTP-1-13.xml

2.1.5 TC-OTP-2-13

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-v1.3/TC-OTP-2-13.xml

2.1.6 TC-OTP-3-13

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-v1.3/TC-OTP-3-13.xml

2.1.7 TC-OTP-4-13

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-v1.3/TC-OTP-4-13.xml

2.1.8 TC-OTP-5-13

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

See test-cases/kmip-v1.3/TC-OTP-5-13.xml

2.1.9 TC-Q-CAP-1-13

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

See test-cases/kmip-v1.3/TC-Q-CAP-1-13.xml

2.1.10 TC-Q-CAP-2-13

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

See test-cases/kmip-v1.3/TC-Q-CAP-2-13.xml

2.1.11 TC-Q-CREG-1-13

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

See test-cases/kmip-v1.3/TC-Q-CREG-1-13.xml

2.1.12 TC-Q-PROF-1-13

Return details of the server claimed supported profiles.

See test-cases/kmip-v1.3/TC-Q-PROF-1-13.xml

2.1.13 TC-Q-PROF-2-13

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

See test-cases/kmip-v1.3/TC-Q-PROF-2-13.xml

2.1.14 TC-Q-PROF-3-13

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-v1.3/TC-Q-PROF-3-13.xml

2.1.15 TC-Q-RNGS-1-13

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-v1.3/TC-Q-RNGS-1-13.xml

2.1.16 TC-Q-RNGS-2-13

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-v1.3/TC-Q-RNGS-2-13.xml

2.1.17 TC-Q-RNGS-3-13

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-v1.3/TC-Q-RNGS-3-13.xml

2.1.18 TC-Q-RNGS-4-13

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-v1.3/TC-Q-RNGS-4-13.xml

2.1.19 TC-Q-RNGS-5-13

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-v1.3/TC-Q-RNGS-5-13.xml

2.1.20 TC-Q-RNGS-6-13

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

See test-cases/kmip-v1.3/TC-Q-RNGS-6-13.xml

2.1.21 TC-Q-S2C-1-13

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-v1.3/TC-Q-S2C-1-13.xml

2.1.22 TC-Q-S2C-2-13

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-v1.3/TC-Q-S2C-2-13.xml

2.1.23 TC-Q-S2C-PROF-1-13

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-v1.3/TC-Q-S2C-PROF-1-13.xml

2.1.24 TC-Q-S2C-PROF-2-13

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-v1.3/TC-Q-S2C-PROF-2-13.xml

2.1.25 TC-Q-VAL-1-13

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

See test-cases/kmip-v1.3/TC-Q-VAL-1-13.xml

2.1.26 TC-Q-VAL-2-13

Return details of the server that does not claim any validations

See test-cases/kmip-v1.3/TC-Q-VAL-2-13.xml

2.1.27 TC-RNG-ATTR-1-13

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-v1.3/TC-Q-RNG-ATTR-1-13.xml

2.1.28 TC-RNG-ATTR-2-13

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-v1.3/TC-Q-RNG-ATTR-2-13.xml

2.1.29 TC-STREAM-HASH-1-13

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-v1.3/TC-STREAM-HASH-1-13.xml

2.1.30 TC-STREAM-HASH-2-13

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-v1.3/TC-STREAM-HASH-2-13.xml

2.1.31 TC-STREAM-HASH-3-13

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-v1.3/TC-STREAM-HASH-3-13.xml

2.1.32 TC-STREAM-ENC-1-13

Create a symmetric key and perform encrypt using the symmetric key.

Variation on CS-BC-M-1-13 in [KMIP-PROFILES]

See test-cases/kmip-v1.3/TC-STREAM-ENC-1-13.xml

2.1.33 TC-STREAM-ENC-2-13

Create a symmetric key and perform encrypt using the symmetric key. Cryptographic Parameters are provided in first Encrypt operation rather than specified against the managed object.

Variation on CS-BC-M-4-13 in [KMIP-PROFILES]

See test-cases/kmip-v1.3/TC-STREAM-ENC-2-13.xml

2.1.34 TC-STREAM-ENCDEC-1-13

Register a symmetric key and perform encrypt using the symmetric key followed by decrypt. The input data is non-block size.

Variation on CS-BC-M-10-13 in [KMIP-PROFILES]

See test-cases/kmip-v1.3/TC-STREAM-ENC-3-13.xml

3 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.

3.1 KMIP 1.3 Test Cases Setup

3.1.1 TC-CREG-1-13

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-v1.3/TC-CREG-1-13.xml

3.1.2 TC-CREG-3-13

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-v1.3/TC-CREG-3-13.xml

3.1.3 TC-OTP-SETUP-1-13

One-Time-Pad setup - for testing purposes only

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 where the one-time-pad has been set up with a simple value for testing rather than actually hooked into a real secure one-time-pad.

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-v1.3/TC-OTP-SETUP-1-13.xml

3.1.4 TC-OTP-SETUP-2-13

One-Time-Pad setup - for testing purposes only

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 where the one-time-pad has been set up with a simple value for tetsing rather than actually hooked into a real secure one-time-pad.

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.

This setup allows for specification of the details without provision of the key material.

See test-cases/kmip-v1.3/TC-OTP-SETUP-2-13.xml

3.1.5 TC-OTP-CLEANUP-1-13

One-Time-Pad cleanup - for testing purposes only

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 where the one-time-pad has been set up with a simple value for tetsing rather than actually hooked into a real secure one-time-pad.

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-v1.3/TC-OTP-CLEANUP-1-13.xml

Appendix A. 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 (v1.0 and v1.1)
Tim Hudson, Cryptsoft (v1.1)
René Pawlitzek, IBM (v1.0)

Technical Committee Participants:

Warren Armstrong, QuintessenceLabs Pty Ltd.
Rinkesh Bansal, IBM
Lina Baquero, Fornetix
Jeff Bartell, Fornetix
Tom Benjamin, IBM
Anthony Berglas, Cryptsoft Pty Ltd.
Mathias Bjorkqvist, IBM
Todd Bottger, Oracle
Joseph Brand, Semper Fortis Solutions
Alan Brown, Thales e-Security
Robert Burns, Thales e-Security
Andrew Byrne, EMC
Hai-May Chao, Oracle
Chye-Lin Chee, Hewlett Packard Enterprise (HPE)
Tim Chevalier, NetApp
Kenli Chong, QuintessenceLabs Pty Ltd.
Justin Corlett, Cryptsoft Pty Ltd.
Tony Cox, Cryptsoft Pty Ltd.
DINESH DIALANI, SafeNet, Inc.
Michael Dong, Hewlett Packard Enterprise (HPE)
Alex Downey, Futurex
Kevin Driver, IBM
Stephen Edwards, Fornetix
James Espinoza, Futurex
Faisal Faruqui, Thales e-Security
Stan Feather, Hewlett Packard Enterprise (HPE)
Indra Fitzgerald, NetApp
Judith Furlong, EMC
Michael Gardiner, SafeNet, Inc.
Jonathan Geater, Thales e-Security
Susan Gleeson, Oracle
Saheem Granados, IBM
John Green, QuintessenceLabs Pty Ltd.
Robert Griffin, EMC
Robert Haas, IBM
Steve He, Vormetric, Inc.
Christopher Hiller, Hewlett Packard Enterprise (HPE)
Hao Hoang, Hewlett Packard Enterprise (HPE)
Tim Hudson, Cryptsoft Pty Ltd.
Michael Jenkins, National Security Agency

Elysa Jones, Individual
Mark Joseph, P6R, Inc
Mahadev Karadigudda, NetApp
Jason Katonica, IBM
Tim Kelsey, Hewlett Packard Enterprise (HPE)
Hyun-jin Kim, Hancom Secure, Inc.
Stephen Kingston, SafeNet, Inc.
Kathy Kriese, Symantec Corp.
Leonardo Ladeira, SafeNet, Inc.
Sun-ho Lee, Hancom Secure, Inc.
John Leiseboer, QuintessenceLabs Pty Ltd.
Hal Lockhart, Oracle
Robert Lockhart, Thales e-Security
Martin Luther, Hewlett Packard Enterprise (HPE)
Jane Melia, QuintessenceLabs Pty Ltd.
Prashant Mestri, IBM
Trisha Paine, SafeNet, Inc.
Incheon Park, Hancom Secure, Inc.
John Peck, IBM
Michael Phillips, Dell
Stefan Pingel, EMC
Ajai Puri, SafeNet, Inc.
Saravanan Ramalingam, Thales e-Security
Bruce Rich, Cryptsoft Pty Ltd.
Warren Robbins, Dell
Peter Robinson, EMC
Rick Robinson, IBM
Saikat Saha, Oracle
Boris Schumperli, Cryptomathic
Greg Scott, Cryptsoft Pty Ltd.
Amit Sinha, SafeNet, Inc.
Radhika Siravara, Oracle
Curtis Smith, Futurex
Ryan Smith, Futurex
Amruta Soman, Cryptsoft Pty Ltd.
Gerald Stueve, Fornetix
Jim Susoy, P6R, Inc
Kiran Thota, VMware, Inc.
Peter Tsai, Vormetric, Inc.
Nathan Turajski, Hewlett Packard Enterprise (HPE)
Charles White, Fornetix
Steve Wierenga, Hewlett Packard Enterprise (HPE)
Thomas Xuelin, Watchdata Technologies Pte Ltd.
Krishna Yellepeddy, IBM
Magda, Zdunkiewicz, Cryptsoft Pty Ltd.
yuan zhang, Watchdata Technologies Pte Ltd.
Joshua Zhu, Vormetric, Inc.

Appendix B. Revision History

Revision	Date	Editor	Changes Made
wd03	8-Jan-2016	Tim Hudson	Reformatted to have test cases in external files.
wd02	26-Jan-2015	Tim Hudson / Mark Joseph	Updates to move setup related items in to a separate section.
wd01	22-Jan-2015	Tim Hudson / Mark Joseph	Initial draft.