PKCS #11 Cryptographic Token Interface Profiles Version 3.0

OASIS Standard

15 June 2020

This stage:
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/os/pkcs11-profiles-v3.0-os.docx (Authoritative)
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/os/pkcs11-profiles-v3.0-os.html
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/os/pkcs11-profiles-v3.0-os.pdf

Previous stage:
N/A

Latest stage:
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/pkcs11-profiles-v3.0.docx (Authoritative)
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/pkcs11-profiles-v3.0.html
https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/pkcs11-profiles-v3.0.pdf

Technical Committee:
OASIS PKCS 11 TC

Chairs:
Tony Cox (tony.cox@cryptsoft.com), Cryptsoft Pty Ltd
Robert Relyea (rrelyea@redhat.com), Red Hat

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

Additional artifacts:
This prose specification is one component of a Work Product that also includes:

- PKCS #11 header files:
  https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/os/include/pkcs11-v3.0/
- ALERT: Due to a clerical error when publishing the Committee Specification, the header files listed above are outdated and may contain serious flaws. The TC is addressing this in the next round of edits. Meanwhile, users of the standard can find the correct header files at https://github.com/oasis-tcs/pkcs11/tree/master/working/3-00-current.

Related work:
This specification replaces or supersedes:

- *PKCS #11 Cryptographic Token Interface Profiles Version 2.40*. Edited by Tim Hudson. Latest stage. http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/pkcs11-profiles-v2.40.html.

This specification is related to:

- *PKCS #11 Cryptographic Token Interface Base Specification Version 3.0.* Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-base/v3.0/pkcs11-base-v3.0.html.
- *PKCS #11 Cryptographic Token Interface Current Mechanisms Specification Version 3.0.* Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-curr/v3.0/pkcs11-curr-v3.0.html.

- *PKCS #11 Cryptographic Token Interface Historical Mechanisms Specification Version 3.0*. Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-hist/v3.0/pkcs11-hist-v3.0.html.

Abstract:
This document is intended for developers and architects who wish to design systems and applications that conform to the PKCS #11 Cryptographic Token Interface standard.

The PKCS #11 Cryptographic Token Interface standard documents an API for devices that may hold cryptographic information and may perform cryptographic functions.

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 stage" 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=pkcs11#technical.

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/pkcs11/.

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

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

[PKCS11-Profiles-v3.0]

PKCS #11 Cryptographic Token Interface Profiles Version 3.0. Edited by Tim Hudson. 15 June 2020. OASIS Standard. https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/os/pkcs11-profiles-v3.0-os.html. Latest stage: https://docs.oasis-open.org/pkcs11/pkcs11-profiles/v3.0/pkcs11-profiles-v3.0.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

This document intends to meet this OASIS requirement on conformance clauses for providers and consumers of cryptographic services via PKCS#11 ([PKCS11-Base] Section 6 - PKCS#11 Implementation Conformance) through profiles that define the use of PKCS#11 data types, objects, functions and mechanisms within specific contexts of provider and consumer interaction. These profiles define a set of normative constraints for employing PKCS#11 within a particular environment or context of use. They may, optionally, require the use of specific PKCS#11 functionality or in other respects define the processing rules to be followed by profile actors.

For normative definition of the elements of PKCS#11 specified in these profiles, see the PKCS#11 Cryptographic Token Interface Base Specification ([PKCS11-Base]).and the PKCS#11 Cryptographic Token Interface Current Mechanisms ([PKCS11-Curr]). Illustrative guidance for the implementation of providers and consumers of PKCS#11 is provided in the PKCS#11 Cryptographic Token Interface Usage Guide ([PKCS11-UG]).

1.1 IPR Policy

This specification is provided under the RF on RAND Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. 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 TC's web page (https://www.oasis-open.org/committees/pkcs11/ipr.php).

1.2 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].

1.3 Normative References

[PKCS11-Base] *PKCS #11 Cryptographic Token Interface Base Specification Version 3.0.* Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-base/v3.0/pkcs11-base-v3.0.html.

[PKCS11-Curr] *PKCS #11 Cryptographic Token Interface Current Mechanisms Specification Version 3.0.* Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-curr/v3.0/pkcs11-curr-v3.0.html.

[PKCS11-Hist] *PKCS #11 Cryptographic Token Interface Historical Mechanisms Specification Version 3.0*. Edited by Chris Zimman and Dieter Bong. Latest stage. https://docs.oasis-open.org/pkcs11/pkcs11-hist/v3.0/pkcs11-hist-v3.0.html.

[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.

1.4 Non-Normative References

[PKCS11-UG] *PKCS #11 Cryptographic Token Interface Usage Guide Version 3.0*. Work in progress.

2 Profiles

2.1 PKCS #11 Profiles

This document defines a selected set of conformance clauses which form PKCS #11 Profiles. The PKCS 11 TC also welcomes proposals for new profiles. PKCS 11 TC members are encouraged to submit these proposals to the PKCS 11 TC for consideration for inclusion in a future version of this TC-approved document. However, some OASIS members MAY simply wish to inform the committee of profiles or other work related to PKCS #11.

2.2 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 **[PKCS11-Base] Section 6** (PKCS#11 Implementation Conformance)
2. Specify the list of additional data types that SHALL be supported
3. Specify the list of additional attributes that SHALL be supported
4. Specify the list of additional objects that SHALL be supported
5. Specify the list of additional functions that SHALL be supported
6. Specify the list of additional mechanisms that SHALL be supported

2.3 Guidelines for Validating Conformance to PKCS #11 Profiles

A PKCS #11 provider implementation SHALL claim conformance to a specific provider profile only if it instruments all required data types, attributes, objects, functions and mechanisms of that profile

- All data types specified as required in that profile
- All attributes specified as required in that profile
- All objects specified as required in that profile
- All functions specified as required in that profile
- All mechanisms specified as required in that profile

A PKCS #11 consumer implementation SHALL claim conformance to a specific consumer profile only if it instruments all required data types, attributes, objects, functions and mechanisms of that profile

- All data types specified as required in that profile
- All attributes specified as required in that profile
- All objects specified as required in that profile
- All functions specified as required in that profile
- All mechanisms specified as required in that profile

Note: items may be specified either directly in a profile or by reference to other profiles. Where another profile is referenced as required, the combination of the requirements of all referenced required profiles (directly or indirectly) SHALL apply.

2.4 Defined Profile Identifiers

Profile objects (object class CKO_PROFILE) describe which PKCS #11 profiles the token implements.

The **CKA_PROFILE** attribute identifies a profile that the token implements.

Attribute	Data type	Meaning
CKA_PROFILE_ID	CK_PROFILE_ID	ID of the supported profile.

The following table defines the **CK_PROFILE_ID** values:

Constant	Meaning
CKP_INVALID_ID	Invalid profile
CKP_BASELINE_PROVIDER	Baseline Provider
CKP_EXTENDED_PROVIDER	Extended Provider
CKP_AUTHENTICATION_TOKEN	Authentication Token
CKP_PUBLIC_CERTIFICATES_TOKEN	Public Certificates Token
CKP_VENDOR_DEFINED	Vendor defined

3 Conformance

3.1 Purpose of this Section

The following subsections describe currently-defined profiles related to the use of PKCS #11. The profiles define classes of PKCS #11 functionality to which an implementation can declare conformance.

3.2 Baseline Consumer Clause

A PKCS #11 consumer calls a PKCS #11 provider implementation of the PKCS #11 API in order to use the cryptographic functionality from that provider.

This profile specifies the most basic functionality that would be expected of a conformant PKCS #11 consumer – the ability to consume information via the cryptographic services offered by a provider.

3.2.1 Implementation Conformance

An implementation is a conforming Baseline Consumer Clause if it meets the conditions as outlined in the following section.

3.2.2 Conformance of a PKCS #11 Baseline Consumer

An implementation conforms to this specification as a Baseline Consumer if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the following data types:
    a. CK_VERSION ([PKCS11-Base] 3.1)
    b. CK_INFO ([PKCS11-Base] 3.1)
    c. CK_SLOT_ID ([PKCS11-Base] 3.2)
    d. CK_SLOT_INFO ([PKCS11-Base] 3.2)
    e. CK_TOKEN_INFO ([PKCS11-Base] 3.2)
    f. CK_SESSION_HANDLE ([PKCS11-Base] 3.3)
    g. CK_USER_TYPE ([PKCS11-Base] 3.3)
    h. CK_SESSION_INFO ([PKCS11-Base] 3.3)
    i. CK_OBJECT_HANDLE ([PKCS11-Base] 3.4)
    j. CK_OBJECT_CLASS ([PKCS11-Base] 3.4)
    k. CK_ATTRIBUTE_TYPE ([PKCS11-Base] 3.4)
    l. CK_ATTRIBUTE ([PKCS11-Base] 3.4)
    m. CK_RV ([PKCS11-Base] 3.6)
    n. CK_FUNCTION_LIST ([PKCS11-Base] 3.6)
    o. CK_C_INITIALIZE_ARGS ([PKCS11-Base] 3.7)
3. Supports the following attributes:
    a. CKA_CLASS ([PKCS11-Base] 4.2)
    b. CKA_VALUE ([PKCS11-Base])
4. Supports the following objects:
    a. None specified
5. Supports the following functions:
    a. C_GetFunctionList ([PKCS11-Base] 5.4)
    b. C_Initialize ([PKCS11-Base] 5.4)
    c. C_Finalize ([PKCS11-Base] 5.4)
    d. C_GetInfo ([PKCS11-Base] 5.4)
    e. C_GetSlotList ([PKCS11-Base] 5.5)

f. C_GetSlotInfo ([PKCS11-Base] 5.5)
g. C_GetTokenInfo ([PKCS11-Base] 5.5)
h. C_OpenSession ([PKCS11-Base] 5.6)
i. C_CloseSession ([PKCS11-Base] 5.6)

6. Supports the following mechanisms:
    a. None specified
7. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
8. Optionally supports any clause within [PKCS11-Base] that is not listed above
9. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.3 Baseline Provider Clause

A PKCS #11 provider makes cryptographic functionality available to a consuming application in terms of the PKCS #11 API.

This profile specifies the most basic functionality that would be expected of a conformant PKCS #11 provider – the ability to provide information about the capabilities of the cryptographic services provided.

3.3.1 Implementation Conformance

An implementation is a conforming Baseline Provider if it meets the conditions as outlined in the following section.

3.3.2 Conformance of a PKCS #11 Baseline Provider

An implementation conforms to this specification as a Baseline Provider if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the following data types:
    a. CK_VERSION ([PKCS11-Base] 3.1)
    b. CK_INFO ([PKCS11-Base] 3.1)
    c. CK_SLOT_ID ([PKCS11-Base] 3.2)
    d. CK_SLOT_INFO ([PKCS11-Base] 3.2)
    e. CK_TOKEN_INFO ([PKCS11-Base] 3.2)
    f. CK_SESSION_HANDLE ([PKCS11-Base] 3.3)
    g. CK_USER_TYPE ([PKCS11-Base] 3.3)
    h. CK_SESSION_INFO ([PKCS11-Base] 3.3)
    i. CK_OBJECT_HANDLE ([PKCS11-Base] 3.4)
    j. CK_OBJECT_CLASS ([PKCS11-Base] 3.4)
    k. CK_ATTRIBUTE_TYPE ([PKCS11-Base] 3.4)
    l. CK_ATTRIBUTE ([PKCS11-Base] 3.4)
    m. CK_PROFILE_ID ([PKCS11-Base] 3.4)
    n. CK_RV ([PKCS11-Base] 3.6)
    o. CK_FUNCTION_LIST ([PKCS11-Base] 3.6)
    p. CK_INTERFACE ([PKCS11-Base] 3.6)
    q. CK_C_INITIALIZE_ARGS ([PKCS11-Base] 3.7)
3. Supports the following attributes:
    a. CKA_CLASS ([PKCS11-Base] 4.2)
    b. CKA_TOKEN ([PKCS11-Base] 4.2)
    c. CKA_VALUE ([PKCS11-Base])
    d. CKA_ID ([PKCS11-Base])
    e. CKA_PRIVATE ([PKCS11-Base] 4.4)
    f. CKA_MODIFIABLE ([PKCS11-Base)

g. CKA_LABEL ([PKCS11-Base)
h. CKA_UNIQUE_IDENTIFIER ([PKCS11-Base] 4.4)
i. CKA_PROFILE_ID ([PKCS11-Base] 4.13)

4. Supports the following objects:
    a. CKO_PROFILE ([PKCS11-Base] 4.13) with value CKP_BASELINE_PROVIDER
5. Supports the following functions:
    a. C_GetFunctionList ([PKCS11-Base] 5.4)
    b. C_GetInterfaceList ([PKCS11-Base] 5.4)
    c. C_GetInterface ([PKCS11-Base] 5.4)
    d. C_Initialize ([PKCS11-Base] 5.4)
    e. C_Finalize ([PKCS11-Base] 5.4)
    f. C_GetInfo ([PKCS11-Base] 5.4)
    g. C_GetSlotList ([PKCS11-Base] 5.5)
    h. C_GetSlotInfo ([PKCS11-Base] 5.5)
    i. C_GetTokenInfo ([PKCS11-Base] 5.5)
    j. C_OpenSession ([PKCS11-Base] 5.6)
    k. C_CloseSession ([PKCS11-Base] 5.6)
    l. C_GetSessionInfo ([PKCS11-Base] 5.6)
    m. C_FindObjectsInit ([PKCS11-Base] 5.6)
    n. C_FindObjects ([PKCS11-Base] 5.6)
    o. C_FindObjectsFinal ([PKCS11-Base] 5.6)
    p. C_GetAttributeValue ([PKCS11-Base] 5.7)
6. Supports the following mechanisms:
    a. None specified
7. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
8. Optionally supports any clause within [PKCS11-Base] that is not listed above
9. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.4 Extended Consumer Clause

This profile builds on the PKCS#11 Baseline Consumer profile to add support for mechanism-based usage.

3.4.1 Implementation Conformance

An implementation is a conforming Extended Consumer if it meets the conditions as outlined in the following section.

3.4.2 Conformance of a PKCS #11 Extended Consumer

An implementation conforms to this specification as Extended Consumer if it meets the following conditions:

1. Supports the conditions required by the PKCS11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the conditions required by the PKCS11 Baseline Consumer clauses section 3.2
3. Supports the following data types:
    a. CK_MECHANISM_TYPE ([PKCS11-Base] 3.4)
    b. CK_MECHANISM ([PKCS11-Base] 3.4)
4. Supports the following attributes:
    a. None specified
5. Supports the following objects:

a. None specified

6. Supports the following functions:
    a. C_GetMechanismList ([PKCS11-Base] 5.5)
    b. C_GetMechanismInfo ([PKCS11-Base] 5.5)
7. Supports the following mechanisms:
    a. None specified
8. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
9. Optionally supports any clause within [PKCS11-Base] that is not listed above
10. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.5 Extended Provider Clause

This profile builds on the PKCS#11 Baseline Provider to add support for mechanism-based usage.

3.5.1 Implementation Conformance

An implementation is a conforming Extended Provider if it meets the conditions as outlined in the following section.

3.5.2 Conformance of a PKCS #11 Extended Provider

An implementation conforms to this specification as Extended Provider if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the conditions required by the PKCS #11 Baseline Provider clauses section 3.3.
3. Supports the following data types:
    a. CK_MECHANISM_TYPE ([PKCS11-Base] 3.4)
    b. CK_MECHANISM ([PKCS11-Base] 3.4)
4. Supports the following attributes:
    a. None specified
5. Supports the following objects:
    a. CKO_PROFILE ([PKCS11-Base] 4.13) with value CKP_EXTENDED_PROVIDER
6. Supports the following functions:
    a. C_GetMechanismList ([PKCS11-Base] 5.5)
    b. C_GetMechanismInfo ([PKCS11-Base] 5.5)
    c. C_Login ([PKCS11-Base] 5.6)
    d. C_LoginUser ([PKCS11-Base] 5.6)
    e. C_Logout ([PKCS11-Base] 5.6)
7. Supports the following mechanisms:
    a. None specified
8. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
9. Optionally supports any clause within [PKCS11-Base] that is not listed above
10. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.6 Authentication Token Clause

This profile builds on the PKCS #11 Baseline Provider and/or Baseline Consumer profiles to provide for use in the context of an authentication token.

3.6.1 Implementation Conformance

An implementation is a conforming Authentication Token if it meets the conditions as outlined in the following section.

3.6.2 Conformance of an Authentication Token

An implementation conforms to this specification as an Authentication Token if it meets the following conditions:

1. If the implementation is a consumer then it SHALL support the conditions required by the PKCS #11 Baseline Consumer Clause (Section 3.2)
2. If the implementation is a provider then it SHALL support the conditions required by the PKCS #11 Baseline Provider Clause (Section 3.3)
3. Supports the following data types:
    a. None specified
4. Supports the following attributes:
    a. None specified
5. Supports the following objects:
    a. CKO_PRIVATE_KEY ([PKCS11-Base] 4.9)
    b. CKO_PUBLIC_KEY ([PKCS11-Base] 4.8)
    c. CKO_PROFILE ([PKCS11-Base] 4.13) with value CKP_AUTHENTICATION_TOKEN
6. Supports the following functions:
    a. C_Login ([PKCS11-Base] 5.6)
    b. C_LoginUser ([PKCS11-Base] 5.6)
    c. C_Logout ([PKCS11-Base] 5.6)
    d. C_SignInit ([PKCS11-Base] 5.13)
    e. C_Sign and/or C_SignUpdate and C_SignFinal ([PKCS11-Base] 5.13)
7. Supports the following mechanisms:
    a. None specified
8. Optionally supports any clause within [PKCS11-Base] that is not listed above
9. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements.

3.7 Public Certificates Token Clause

This profile builds on the PKCS #11 Baseline Provider and/or Baseline Consumer profiles to provide for use in the context of a public certificates token.

3.7.1 Implementation Conformance

An implementation is a conforming Public Certificates Token if it meets the conditions as outlined in the following section.

3.7.2 Conformance of a Public Certificates Token

An implementation conforms to this specification as Public Certificates Token if it meets the following conditions:

1. If the implementation is a consumer then it SHALL support the conditions required by the PKCS #11 Baseline Consumer Clause (Section 3.2)

2. If the implementation is a provider then it SHALL support the conditions required by the PKCS #11 Baseline Provider Clause (Section 3.3)
3. Supports the following data types:
    a. None specified
4. Supports the following attributes:
    a. None specified
5. Supports the following objects:
    a. CKO_CERTIFICATE ([PKCS11-Base] 4.6)
    b. CKO_PROFILE ([PKCS11-Base] 4.13) with value CKP_PUBLIC_CERTIFICATES_TOKEN
6. Supports the following functions:
    a. None specified
7. Supports the following mechanisms:
    a. None specified
8. Supports the following object location requirements:
    a. All certificates are publicly readable, able to be found on the token without a login having been performed
    b. All certificates for which a matching private key also exists on the token must have a matching CKA_ID attribute for the certificate and private key
    c. One or more of the following conditions must be met:
        i. The matching private key for a certificate can be found via C_FindObjects using the matching CKA_ID value without a login having been performed;
        ii. The matching public key for a certificate can be found via C_FindObjects using the matching CKA_ID value without a login having been performed.
9. Optionally supports any clause within [PKCS11-Base] that is not listed above
10. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements.

Appendix A. Acknowledgments

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

Participants:

Benton Stark - Cisco Systems

Anthony Berglas - Cryptsoft Pty Ltd.

Justin Corlett - Cryptsoft Pty Ltd.

Tony Cox - Cryptsoft Pty Ltd.

Tim Hudson - Cryptsoft Pty Ltd.

Bruce Rich - Cryptsoft Pty Ltd.

Greg Scott - Cryptsoft Pty Ltd.

Jason Thatcher - Cryptsoft Pty Ltd.

Magda Zdunkiewicz - Cryptsoft Pty Ltd.

Andrew Byrne - Dell

David Horton - Dell

Kevin Mooney - Fornetix

Gerald Stueve - Fornetix

Charles White - Fornetix

Matt Bauer - Galois, Inc.

Wan-Teh Chang - Google Inc.

Patrick Steuer - IBM

Michele Drgon - Individual

Gershon Janssen - Individual

Oscar So - Individual

Michelle Brochmann - Information Security Corporation

Michael Mrkowitz - Information Security Corporation

Jonathan Schulze-Hewett - Information Security Corporation

Philip Lafrance - ISARA Corporation

Thomas Hardjono - M.I.T.

Hamish Cameron - nCipher

Paul King - nCipher

Sander Temme - nCipher

Chet Ensign - OASIS

Jane Harnad - OASIS

Web Master - OASIS

Dee Schur - OASIS

Xuelei Fan - Oracle

Jan Friedel - Oracle

Susan Gleeson - Oracle

Dina Kurktchi-Nimeh - Oracle

John Leser - Oracle

Darren Moffat - Oracle
Mark Joseph - P6R, Inc
Jim Susoy - P6R, Inc
Roland Bramm - PrimeKey Solutions AB
Warren Armstrong - QuintessenceLabs Pty Ltd.
Kenli Chong - QuintessenceLabs Pty Ltd.
John Leiseboer - QuintessenceLabs Pty Ltd.
Florian Poppa - QuintessenceLabs Pty Ltd.
Martin Shannon - QuintessenceLabs Pty Ltd.
Jakub Jelen - Red Hat
Chris Malafis - Red Hat
Robert Relyea - Red Hat
Christian Bollich - Utimaco IS GmbH
Dieter Bong - Utimaco IS GmbH
Chris Meyer - Utimaco IS GmbH
Daniel Minder - Utimaco IS GmbH
Roland Reichenberg - Utimaco IS GmbH
Manish Upasani - Utimaco IS GmbH
Steven Wierenga - Utimaco IS GmbH

Appendix B. Revision History

Revision	Date	Editor	Changes Made
wd06	28-May-2019	Tony Cox	Final cleanup of front introductory texts and links prior to CSPRD
wd05	18-Apr-2019	Tim Hudson	Remove CKA_USER reference
wd04	16-Apr-2019	Tim Hudson	Update given function name changes in specification
wd03	24-Sep-2018	Tim Hudson	Update based on list comments
wd02	19-Sep-2018	Tim Hudson	Update based on list comments
wd01	05-Sep-2018	Tim Hudson	Initial Draft