Release Notes

Specifications for the Java Card 3 Platform

Version 3.0.5, Classic Edition
November 2017

Table of Contents

1. Acknowledgment

Dedicated to Tanjore (Ravi) Ravishankar, 1958 - 2015. His leadership, passion and devotion inspired us all

Top

2. Overview

These release notes describe the specifications for the Java Card 3 Platform, Version 3.0.5, Classic Edition. The Java Card 3 platform consists of versions 3.0, 3.0.1, 3.0.4 and 3.0.5 of the specifications and versions 3.0.1, 3.0.2, 3.0.3, 3.0.4 and 3.0.5 of the development kit.

The Java Card platform is available in two editions, Classic and Connected. This Version 3.0.5 release of the specifications is for the Classic Edition; the Connected Edition specifications are not included in this release.

This release of the Classic Edition specifications is based on an evolution of Version 3.0.4 of the Classic Edition specifications, and targets more resource-constrained devices that support applet-based applications. This release introduces more security features/updates to protect against attacks and better alignment with other industry standards. Also included are bug fixes and clarifications against the Java Card Classic Edition specifications, Version 3.0.4, and new security algorithms.

CLASSIC EDITION SPECIFICATION	**DESCRIPTION**
Runtime Environment Specification, Java Card Platform, Classic Edition, Version 3.0.5	This specification describes the runtime environment (RE) for the Classic Edition of the Java Card Platform. This RE mirrors those REs found in previous releases of the Java Card platform, including v3.0.4, v3.0.1 and v2.2.2.
Application Programming Interface, Java Card Platform, Classic Edition, Version 3.0.5	This API defines a set of classes upon which Java Card technology-based applets can be constructed. This API mirrors those APIs found in previous releases of the Java Card platform, including v3.0.4, v3.0.1 and v2.2.2.
Virtual Machine Specification, Java Card Platform, Classic Edition, Version 3.0.5	This specification describes the virtual machine for the Classic Edition of the Java Card Platform. This VM mirrors those VMs found in previous releases of the Java Card platform, including v3.0.4, v3.0.1 and v2.2.2.

Top

3. Supported Platforms

The documents are accessible on any computer system with an unzip utility, Adobe Acrobat Reader (version 4.0 or later), and a CSS-compliant web browser.

HTML can be viewed with any CSS-compliant browser software, such as:

- Internet Explorer, version 5.0 or later
- Mozilla Firefox, version 11.0 or later

PDF files can be viewed in your web browser *with an appropriate plugin* or in Adobe® Acrobat Reader. Most recent browsers include the PDF reader plugin. If your browser does not, you can download the plugin from the browser vendor's web site or the Adobe web site at http://www.adobe.com/products/acrobat/readstep.html.

Top

4. Installation Instructions

Download and unzip the specifications bundle. The bundle unzips into the subdirectory javacard_specifications-3_0_5-RR, within which you will find the subdirectory classic.

Within the classic subdirectory you will find the specifications:

- api_classic - contains the Java Card API specification for the Classic Edition in Javadoc™ tool HTML format. These can be viewed in most browsers but do not render well in Mozilla Firefox 3.0.10.
- jcre_classic - contains the Java Card runtime environment specification for the Classic Edition in PDF format (JCREspecCLASSIC-3_0_5-RR.pdf).
- jcvm_classic - contains the Java Card virtual machine specification for the Classic Edition in PDF format (JCVMspecCLASSIC-3_0_5-RR.pdf).

Top

5. Changes in the Classic Edition Specifications Since Version 3.0.4

The following sections describe the changes to the Classic Edition specifications for the Java Card platform since the Version 3.0.4 release.

Application Programming Interface, Version 3.0.5, Classic Edition

This section describes the changes to the *Application Programming Interface Specification, Java Card Platform, Version 3.0.5, Classic Edition* since the Version 3.0.4 release.

The export files associated with the API packages of the Java Card Platform, Classic Edition, are included in the reference implementation bundle. The export files are subject to change until the final release. The new package version numbers are:

- javacard.framework - version 1.6
- javacard.security - version 1.6
- javacardx.crypto - version 1.6
- javacardx.apdu.util - version 1.0
- javacardx.biometry1toN - version 1.0
- javacardx.security -version 1.0

Summary

Updates to the API specification, Version 3.0.5, since Version 3.0.4 include:

- **javacard.framework**
  - **javacard.framework.APDU**
    - Added the new constants - PROTOCOL_MEDIA_CONTACTLESS_TYPE_F and PROTOCOL_MEDIA_HCI_APDU_GATE to support JIS X 6319-4:2010 transport protocol Type F and Transport protocol Media - APDU over HCI defined for the APDU gate in ETSI TS 102 622 respectively.
  - **javacard.framework.OwnerPINBuilder**
    - New class - OwnerPINBuilder. This class is factory for Owner PIN objects.
  - **javacard.framework.OwnerPINx**

- New interface - OwnerPINx. This interface represents an Owner PIN, extends Personal Identification Number functionality as defined in the PIN interface, and provides the ability to update the PIN, update the try limit and try counter and thus owner functionality.

- **javacard.framework.OwnerPINxWithPredecrement**
  - New interface - OwnerPINxWithPredecrement. This interface extends the OwnerPINx interface, to support the decrementing of the tries counter before any PIN validation attempts.
- **javacard.framework.PINException**
  - Added new constant ILLEGAL_STATE to indicate a method has been invoked at an illegal or inappropriate time.
- **javacard.framework.SensitiveArrays**
  - New class - SensitiveArrays. This class provides methods for creating and handling integrity-sensitive array objects.
- **javacard.framework.Util**
  - Added method arrayFill which fills an array in an atomic way.

- **javacard.security**
  - **Diffie-Hellman modular exponentiation**
    - New interfaces - DHKey, DHPrivateKey, and DHPublicKey**.** These interfaces support Diffie-Hellman modular exponentiation.
  - **Domain Data Conservation**
    - Added constants ALG_TYPE_DH_PARAMETERS, ALG_TYPE_DSA_PARAMETERS, ALG_TYPE_EC_F2M_PARAMETERS, ALG_TYPE_EC_FP_PARAMETERS and method buildKeyWithSharedDomain(byte algorithmicKeyType, byte keyMemoryType, Key domainParameters, boolean keyEncryption) to class javacard.security.KeyBuilder to support Domain Data Conservation for Diffie-Hellman, Elliptic Curve and DSA keys
  - **OneShot inner classes**
    - Added classes javacard.security.Signature.OneShot, javacard.security.RandomData.OneShot, javacard.security.InitializedMessageDigest.OneShot and javacard.security.MessageDigest.OneShot to support efficient one-shot, signing, verification, random number generation and hashing operations.
  - **RSA 3K support**
    - Added new constant LENGTH_RSA_3072 to support 3K length for RSA keys.
  - **javacard.security.RandomData**
    - Added new constants ALG_FAST, ALG_KEY_GENERATION, ALG_PRESEEDED_DRBG ALG_TRNG and methods getAlgorithm() and nextBytes(byte[] buffer, short offset, short length) to class javacard.security.RandomData.
  - **javacard.security.MessageDigest**
    - Added new constants ALG_SHA3_224, ALG_SHA3_256, ALG_SHA3_384, ALG_SHA3_512, LENGTH_SHA3_224, LENGTH_SHA3_256, LENGTH_SHA3_384 and LENGTH_SHA3_512 to support SHA3
  - **javacard.security.Signature**

- Added method verifyPreComputedHash(byte[] hashBuff, short hashOffset, short hashLength, byte[] sigBuff, short sigOffset, short sigLength) to support verification of pre-computed hash.
- Added constants SIG_CIPHER_ECDSA_PLAIN to support plain ECDSA.
- Added new constants ALG_AES_CMAC_128 and SIG_CIPHER_AES_CMAC128 to support AES CMAC signature algorithm

- **javacardx.apdu.util**
    - Extension package javacardx.apdu.util added that contains APDUUtil class which contains utility functions to parse CLA byte from a command APDU.
- **javacardx.biometry1toN**
    - Extension package javacardx.biometry1toN added that contains functionality for implementing a 1:N biometric framework on the Java Card platform.
- **javacardx.crypto**
    - **javacardx.crypto.Cipher**
        - Added new constants ALG_SHA3_224, ALG_SHA3_256, ALG_SHA3_384, ALG_SHA3_512, LENGTH_SHA3_224, LENGTH_SHA3_256, LENGTH_SHA3_384 and LENGTH_SHA3_512 to support SHA3
        - Added new extension class javacardx.crypto.Cipher.OneShot to support efficient one-shot cryptographic operations.
        - Added new constant ALG_AES_CTR to support a cipher using AES in counter (CTR) mode.
        - Added new constants PAD_PKCS1_OAEP_SHA224, PAD_PKCS1_OAEP_SHA256, PAD_PKCS1_OAEP_SHA384 and PAD_PKCS1_OAEP_SHA512 to extend PKCS#1-OAEP scheme support to SHA224, SHA256, SHA384 and SHA512
    - **javacardx.crypto.AEADCipher**
        - New abstract class javacardx.crypto.AEADCipher to support Authenticated Encryption with Associated Data (AEAD) ciphers. Only GCM and CCM modes of operation for AES are supported in this version of the spec.
- **javacardx.security**
    - Extension package javacardx.security added that contains functionality, for implementing security countermeasures to protect security relevant applet assets on the Java Card platform. The package contains SensitiveResult class which provides methods for asserting results of sensitive functions. List of API functions that have been updated to support usage of this functionality can be found in the specification for this class
- **javacardx.framework.util**
    - Added methods arrayFillGeneric and arrayFillGenericNonAtomic to class ArrayLogic.

Runtime Environment Specification, Version 3.0.5, Classic Edition

This section describes the changes to the *Runtime Environment Specification, Java Card Platform, Version 3.0.5, Classic Edition* since the Version 3.0.4 release.

Summary

Updates to the Runtime Environment specification, Version 3.0.5, since Version 3.0.4 include:

- Reformatting and bug fixes

Virtual Machine Specification, Version 3.0.5, Classic Edition

Tools related to CAP file generation and verification in the Java Card Development Kit by Oracle, have been updated to support JDK 7 class format. Virtual Machine Specification, Java Card Platform, Classic Edition, however, were not changed to provide this support. The changes to the *Virtual Machine Specification, Java Card Platform, Version 3.0.5, Classic Edition* since the Version 3.0.4 release include:

- Reformatting and bug fixes

Top

6. Bugs Fixed in Version 3.0.5

This section describes the bugs that have been fixed in this release of the Java Card specifications, Classic Edition, Version 3.0.5.

BUG ID	DESCRIPTION
JCCL-2873	SPEC: javadoc for javacard.security.KeyAgreement.ALG_EC_SVDP_DH_PLAIN_XY should be clarified
JCCL-2746	JavaDoc: JCSystem.getPreviousContextAID()
JCCL-2692	[SC] It's not clear in JCVM that defining an initialized non-primitive final static field in an interface is prohibited
JCCL-2678	Key check requirements in API spec for the init methods of Signature, Cipher, KeyAgrement class and KeyPair.genKeyPair method unclear
JCCL-2873	SPEC: javadoc for javacard.security.KeyAgreement.ALG_EC_SVDP_DH_PLAIN_XY should be clarified
JCCL-2657	bertlv spec clarification needed

Top

7. Product Information

The Java Card technology web site is http://www.oracle.com/technetwork/java/javacard.

Visit this website to access most up-to-date information on the following:

- Product news and reviews
- Release notes and product documentation

Top

8. Specification Errata

This section contains a list of important reported errata for the specifications included in the specification bundle of the Java Card Platform, Version 3.0.5, Classic Edition that was released in June 2015.

BUG ID	DESCRIPTION	CORRECTION
JCCL-3066	Incorrect value of APDU.PROTOCOL_MEDIA_HCI_APDU_GATE/ PROTOCOL_MEDIA_CONTACTLESS_TYPE_F	In Java Card specification 3.0.5, the constant javacard.framework.APDU.PROTOCOL_MEDIA_HCI_APDU_GATE has an incorrect value of (byte)0xB0. Developers are advised to use the correct value (byte)0xC0 instead.
JCCL-3075	verifyPrecomputedHash method spec needs update	Specification for method verifyPrecomputedHash method in class javacard.security.Signature is missing HMAC from the list of algorithms, which if used, result in this method throwing an exception. Inclusion of HMAC in the list along with DES, AES and Korean Seed will be considered for the future versions of the specification.

JCCL-3085	javacard.security.Key.getType is undefined for domainParameters objects	New constants will be considered for domainParameters type of objects in future versions of the specification. Candidate values are as follows:
		TYPE_DH_KEY_PARAMETER=38,
		TYPE_DSA_KEY_PARAMETER=39,
		TYPE_EC_F2M_KEY_PARAMETER=40,
		TYPE_EC_FP_KEY_PARAMETER=41

JCCL-3090	Add reference to OwnerPINx.getTryLimit to SensitiveResult "See also" section	SensitiveResult class specification contains a list of all methods that may set the internal state which can be checked via methods in SensitiveResult class. This list is missing the method OwnerPINx.getTryLimit. Note that this method may also set the internal state if supported by the platform which can be checked using methods in SensitiveResult class. For details refer to specification for OwnerPinx.getTryLimit.
JCCL-3108	Define the relationship between Signature.OneShot and SignatureMessageRecovery	Note that Signature.OneShot cannot implement SignatureMessageRecovery. Future versions of the specification may clarify this relationship.

Top