EMV®

Secure Remote Commerce

Version Management for SRC API and JavaScript SDK Specifications

Version 1.0

June 2021

Legal Notice

The EMV® Specifications are provided “AS IS” without warranties of any kind, and EMVCo neither assumes nor accepts any liability for any errors or omissions contained in these Specifications. EMVCO DISCLAIMS ALL REPRESENTATIONS AND WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AS TO THESE SPECIFICATIONS.

EMVCo makes no representations or warranties with respect to intellectual property rights of any third parties in or in relation to the Specifications. EMVCo undertakes no responsibility to determine whether any implementation of the EMV® Specifications may violate, infringe, or otherwise exercise the patent, copyright, trademark, trade secret, know-how, or other intellectual property rights of third parties, and thus any person who implements any part of the EMV® Specifications should consult an intellectual property attorney before any such implementation.

Without limiting the foregoing, the Specifications may provide for the use of public key encryption and other technology, which may be the subject matter of patents in several countries. Any party seeking to implement these Specifications is solely responsible for determining whether its activities require a license to any such technology, including for patents on public key encryption technology. EMVCo shall not be liable under any theory for any party’s infringement of any intellectual property rights in connection with the EMV® Specifications.

Revision Log—Version 1.0

This is the first published version of this document.

Contents

1 Introduction

The EMV Secure Remote Commerce (SRC) specifications describe the functional requirements for the SRC ecosystem as well as the interactions and data exchanges among SRC Participants in order to prepare and assert Payment Data to a merchant or commerce provider using a payment-enabled application for a remote commerce payment interaction. Common messages structure, data fields and systems processes provide consistency and reduce integration complexity by SRC System Participants for the enablement of a remote commerce payment interaction. As the set of SRC specification will be enhanced a proper version management of the revisions needs to be established.

1.1 Purpose

This document, the EMVCo Secure Remote Commerce Version Management for SRC API and JavaScript SDK Specifications, describes how EMV Secure Remote Commerce (SRC) Specification Versions are assigned and incremented. This is based on, but not necessarily limited to, pre-existing widespread common practices used within the software industry.

This is specific to the following documents:

- SRC API
- SRC JavaScript SDK

1.2 Audience

This document is intended for use by implementers of the EMV SRC specifications.

1.3 Terminology and Conventions

The terminology used in this document and their specific meaning is described in the following sections.

1.3.1 Specification Version

The Specification Version refers to the EMV SRC specification version that SRC Participants support (e.g. 2.1).

Specification Version format is: MAJOR.MINOR.

- A MAJOR version for significant and non-backward compatible specification changes (e.g. a new architecture). A MAJOR version increase may include additional changes

- A MINOR version for added functionality and deprecated functionality

Note: A bug fix will not contain a version numbering for immediate specification fixes as it is published in a bulletin only. A subsequent minor or major release must include those changes.

1.3.2 Definitions

For the definition of the terms used in this document, refer to the EMV Core Specification, Table 1.3.

1.4 References

The document is specific to the SRC API and SRC Java Script SDK, which are listed in Table 1.1, along with the SRC Core Specification, and are located at www.emvco.com. They should be used in conjunction with this document.

The latest version of any reference, including all published amendments, shall apply unless a publication date is explicitly stated.

Table 1.1: EMVCo References

Reference	Publication Name
SRC Core Specification	EMV® Secure Remote Commerce Specification
SRC API	EMV® Secure Remote Commerce Specification – API
SRC JavaScript SDK	EMV® Secure Remote Commerce Specification – JavaScript SDK

2 Specification Version Management

This Section provides an overview of SRC specification change management. Specification changes can include specification releases, updates, and clarifications.

EMV SRC specifications and SRC components identify the specification version using a two-part version number.

- Specification Version number format: MAJOR.MINOR
- Specification Version number example: 1.0

In case bug fixes to published documents are required, a bulletin release can be made without increasing the version numbering of the specification, however the APIs needs to reflect those changes in subsequent versions.

2.1 Major Releases

A major release occurs when EMVCo has made non-backward compatible changes and uplifts to the SRC specifications and in this way signals to the industry that the specification has gone through significant changes. It contains modifications with large impact on existing system due to impact of change or better technology, completely different architecture, large extensions to existing architectures, security breaches, destructive modifications or completely new technology concepts requiring different design paradigms. For example, the Specification Version number would be updated from v2.1 to v3.0.

2.1.1 Numbering

Specification Version impact:

- Position 1 is incremented
- Position 2 is reset to “0”

Example:

- Previous Specification Version number: 2.5
- New Specification Version number: 3.0

2.1.2 Process

2.1.2.1 Technical Associate (TA) and Subscriber Input and Feedback

All major release changes to the SRC specifications(s) would allow EMVCo Technical Associates and Subscribers to have an opportunity to provide input and feedback on the draft specifications. Once the input and feedback has been finalised, the specifications would be published as public to the industry.

2.1.2.2 Documentation

SRC specification-related documentation will be updated to the new Specification Version number and will be published to the EMVCo website. These documentation updates include:

- SRC API
- SRC JavaScript SDK

These documents will carry the same Specification Version number, regardless whether changes were made within the document. If changes were not made to the document, it will be noted under the document's change log. All other SRC documentation will be updated if changes are deemed necessary for that Major Release.

2.1.3 Communication

EMVCo will publish a notification announcing the scope of the new features for implementation. The notification would also indicate the target date as to when the new specification would be published.

2.2 Minor Releases

A minor release indicates that EMVCo has made additive changes to the SRC specifications. Changes introduced by adding new components to an existing specification may enhance existing or provide new functionality in the specifications. A minor release may include deprecated content to manage non-backward compliant changes. A minor release signals to the industry that the SRC specifications has added new features for implementation. For example, the Specification Version number would be updated from v2.1 to v2.2.

2.2.1 Numbering

Specification Version number impact:

- Position 1 has no change
- Position 2 is incremented

Example:

- Previous Specification Version number: 2.1
- New Specification Version number: 2.2

2.2.2 Process

2.2.2.1 Technical Associate (TA) and Subscriber Input and Feedback

All minor release changes to the SRC specifications(s) would allow EMVCo Technical Associates and Subscribers to have an opportunity to provide input and feedback on the draft documentation. Once the input and feedback has been finalized, the documentation would be published as public to the industry.

2.2.2.2 Documentation

SRC specification-related documentation will be updated to the new Specification Version number and will be published to the EMVCo website. These documentation updates may include:

- SRC API
- SRC JavaScript SDK

The SRC API and SRC JavaScript SDK may increment Specification Versions independently.

2.2.3 Communication

EMVCo will publish a notification announcing the scope of the new optional features for implementation. The notification would also indicate the target date as to when the new specification would be published.

2.3 Deprecation Process for Backward Incompatible Changes

Deprecation defines the process to manage content changes, which may impact backward compatibility. Deprecated content remains in the specification, it signals content which may get deleted over time and may recommend alternative approaches.

As per definition of minor release, the updates could be backward incompatible changes as shown in the following examples:

- A data element conditionality is updated e.g. from "optional" to "required"
- An entire complex data element undergoes changes to the structure and conditionality
- A service, method or operation is deleted as it is found not to be useful
- An entire section in the document is deleted as the contents are outdated or do not serve the purpose

For the changes above, a deprecation process will be followed, where the breaking change content is not deleted, rather the specification would use “deprecated” keyword to indicate the

deprecated elements and optionally “replaced by” syntax to highlight the new elements replacing the deprecated one.

Example 1: srcCorrelationId is updated from "optional" to "conditional"

{

required String srcClientId;
conditional String srcDpaId;
DEPRECATED ~~*optional String srcCorrelationId;*~~
replaced by *conditional String srcCorrelationId2;*
optional String serviceId;
optional String srciTransactionId;
required Address billingAddress;
optional Boolean setAsShippingAddress;

}

Example 2: Backward incompatible updates to Assurance data structure (excerpt)

AssuranceData

Data Element	R/C/O	Constraints	Description
verificationData Type: List<VerificationData>	O	See VerificationData	Set of verification data structures relating to different types of assurance.
~~**cardVerificationEntity**~~ ~~Type: String (Numeric)~~ DEPRECATED	O	Length = 2	Entity performing card verification. Valid values are: • 01 SRC Initiator • 02 SRC System • 03 SRCPI • 04 DCF • 05 DPA • 06 - 99 Others

2.3.1.1 Deprecation policy

The following policy is applied for deprecation:

- Till and when deprecated content is not removed in a next minor version update, existing implementations are not impacted by backward incompatibility
- EMVCo may remove deprecated content within the next major version publication.

2.4 Bug fixes

A bug fix indicates that EMVCo has made required specification fixes and would signal to the industry to immediately update their SRC software. This change gets introduced when minor updates to the specifications need to be done due to oversight, errata, etc. A bug fix may or may not be backward compatible. Bug fixes may only be done in urgent cases which require immediate action.

2.4.1 Process

2.4.1.1 Documentation

All bug fixes to the SRC specifications(s) would require that a bulletin to be published immediately to the industry as public documents. This bulletin will be published as a separate document and it will not increase the versioning number of the underlying specification. The bulletin will provide a change log from its document of origin.

An EMVCo Technical Associates and Subscriber draft review is not required prior to a bulletin release.

2.4.1.2 Communication

EMVCo will publish a Bulletin announcing the scope of the bug fix changes for implementation.

In addition to the EMVCo Bulletin process, push notifications are recommended to the Payments Systems to ensure their customers are aware of the upcoming changes.

The Bulletin would be published immediately without EMVCo Technical Associates and Subscribers draft review. Subsequently, the Bulletin updates will be incorporated into the next major or minor revision of the specification as well.

***** END OF DOCUMENT *****