



**EMV® Specification Bulletin No. 205 v1**  
**August 2018**

---

**EMV 3-D Secure SDK and Device Information Updates,  
Clarifications & Errata**

*This Draft Specification Bulletin No. 205 provides updates, clarifications and errata incorporated into the EMV 3-D Secure—SDK Specification and EMV 3-D Secure—Device Information documents since the October 2017 v2.1.0 publication.*

---

**Applicability**

*This Specification Bulletin applies to:*

- *EMV 3-D Secure—SDK Specification, Version 2.1.0*
- *EMV 3-D Secure—Device Information*

*Updates are provided by document, in the order in which they appear in the specification. Deleted text is identified using ~~strike~~through, and **red** font is used to identify changed text. Unedited text is provided only for context.*

**Effective Date**

*August 2018*

---



# Contents

EMV 3-D Secure SDK and Device Information Updates, Clarifications & Errata .....	1
Applicability.....	1
Effective Date .....	1
<b>EMV 3-D Secure SDK—Device Information v2.1.0.....</b>	<b>5</b>
Chapter 2 Device Identification Parameters .....	5
2.2 Minimum Supported Platform Versions.....	5
Table 2.2: Common Parameters Available in Android, iOS and Windows 10 Mobile Platforms...	5
2.7 Windows 10 Mobile Specific Device Parameters.....	5
Table 2.5: Windows 10 Mobile-Specific Device Parameters .....	5
2.9 Device Information JSON Data .....	5
Table 2.7: Device Parameters JSON Structure.....	5
<b>EMV 3-D Secure SDK Specification v2.1.0 .....</b>	<b>5</b>
Chapter 1 Introduction.....	5
1.3 Normative References.....	5
Chapter 3 Getting Started with the EMV 3-D Secure Mobile SDK .....	6
3.3.2 Challenge Flow.....	6
24 [Req 10] .....	6
Chapter 4 Code Elements of the EMV 3-D Secure Mobile SDK .....	6
4.1 initialize.....	6
Table 4.2 initialize Parameters .....	6
4.5 Class UiCustomization .....	6
4.2 Class Config Parameters.....	6
Table 4.9: ConfigParameters Class Methods.....	6
4.2.1 addParam .....	6
Table 4.10: addParamParameters.....	6
Table 4.11: addParam Exceptions.....	6
4.2.2 getParamValues .....	6
Table 4.12: getParamValue Parameters .....	7
getParamValue Return Value.....	7
4.2.3 removeParam .....	7
Table 4.14: removeParam Parameters.....	7
removeParam Return Value.....	7
4.5 Class UiCustomization .....	7
4.5.2 setButtonCustomization .....	7
Table 4.28: setButtonCustomization Parameters .....	7
4.5.6 getButtonCustomization .....	8
getButtonCustomization Return Value .....	8
4.5.7 getButtonCustomization .....	8
getButtonCustomization Return Value .....	8

4.5.8 getToolbarCustomization .....	8
getToolbarCustomization Return Value .....	8
4.5.9 getLabelCustomization .....	8
getLabelCustomization Return Value .....	8
4.5.10 getTextBoxCustomization .....	8
getTextBoxCustomization Return Value .....	8
4.10.4 getBorderColor .....	8
4.23 Enum Button Type.....	8
Table 4.100: ButtonType Enum .....	8
Chapter 7 User Interface.....	9
7.2.1 Input and Output Formats for Native UI.....	9
Chapter 8 SDK Security .....	9
[Req 68] .....	9
Annex D .....	9
D.1 Code Sample for iOS, D.2 Code Sample for Android, D.3 Code Sample for Windows Phone...	9

## Chapter 2 Device Identification Parameters

### 2.2 Minimum Supported Platform Versions

**Table 2.2: Common Parameters Available in Android, iOS and Windows 10 Mobile Platforms**

Identifier	Parameter	Description	Permissions
C007		<del>For example, in Windows 10 Mobile, AdvertisingManager.AdvertisingId retrieves a unique ID used to provide more relevant advertising.</del>	

### 2.7 Windows 10 Mobile Specific Device Parameters

**Table 2.5: Windows 10 Mobile-Specific Device Parameters**

Group or Identifier	Attribute	Description
W025	<del>PublisherHostId</del>	<del>A string that uniquely identifies a device. This string is unique per device and per publisher, which means that no two publishers will receive the same value for the same device.</del>

### 2.9 Device Information JSON Data

**Table 2.7: Device Parameters JSON Structure**

Updated all instances of:

en\_US to en-US

## Chapter 1 Introduction

### 1.3 Normative References

Updated URL links:

- RFC 7159      *The JavaScript Object Notation (JSON) Data Interchange Format*  
<https://tools.ietf.org/html/rfc7159>
- RFC 7518      *JSON Web Algorithms (JWA)*  
<https://tools.ietf.org/html/rfc7518>

---

## Chapter 3 Getting Started with the EMV 3-D Secure Mobile SDK

### 3.3.2 Challenge Flow

#### 24 [Req 10]

Call one of the methods (`completed`, `cancelled`, `protocolError` or `runtimeError`) of the `ChallengeStatusReceiver` callback object to return the result of the challenge process to the 3DS Requestor App. **After the method is run, the 3DS SDK gets back control** and cleans up resources that are held by the `Transaction` object (**implementation specific**).

---

## Chapter 4 Code Elements of the EMV 3-D Secure Mobile SDK

### 4.1 initialize

#### Table 4.2 initialize Parameters

`locale` Description: ~~en\_US~~ to **en-US**

### 4.5 Class UiCustomization

```
public enum ButtonType {VERIFYSUBMIT, CONTINUE, NEXT, CANCEL, RESEND}
```

### 4.2 Class Config Parameters

- Group names and parameter names are case-insensitive.**

#### Table 4.9: ConfigParameters Class Methods

Method	Description
<code>removeParam</code>	Removes a configuration parameter either from the specified group or from the default group. It should return the <del>name</del> <b>value</b> of the parameter that it removes.

#### 4.2.1 addParam

The `addParam` method shall add a configuration parameter either to the specified group or to the default group, **if the group is not specified**.

#### Table 4.10: addParamParameters

Parameters	Mandatory?	Description
<code>paramValue</code>	<del>No</del> <b>Yes</b>	<b>Note: The value cannot be null.</b>

#### Table 4.11: addParam Exceptions

Method	Description
<code>InvalidInputException</code>	This exception shall be thrown if <code>paramName</code> is null <b>or if the parameter in the group is duplicate</b> .

#### 4.2.2 getParamValues

The `getParamValue` method shall return a configuration parameter's value either from the specified group or from the default group, **if the group is not specified**.

**Table 4.12: getParamValue Parameters**

Parameters	Mandatory?	Description
group	No	<del>Note: If the group is null, then the default group shall be used.</del>

**getParamValue Return Value**

The `getParamValue` method returns the value of the specified configuration parameter as a string. **If the parameter is not found in the specified group, then this method returns null.**

**Note: If the group is null, then the default group shall be used for lookup.**

**4.2.3 removeParam**

The `removeParam` method shall remove a configuration parameter either from the specified group or from the default group, **if the group is not specified**. It should return the **namevalue** of the parameter that it removes.

**Table 4.14: removeParam Parameters**

Parameters	Mandatory?	Description
group	No	<del>Note: If the group is null, then the default group shall be used.</del>

**removeParam Return Value**

The `removeParam` method returns the **namevalue** of the parameter that it removes. **If the parameter is not found in the specified group, then this method returns null.**

**Note: If the group is null, then the default group shall be used for lookup.**

**4.5 Class UiCustomization****[Req 69]**

The `UICustomization` object and the objects held by it, such as `ButtonCustomization`, `ToolbarCustomization`, `LabelCustomization`, and `TextBoxCustomization`, shall be immutable. In other words, after the customization data is set, the 3DS SDK shall not allow the data to be modified. To enforce this, SDK implementers can choose any mechanism such as delinking references from the 3DS Requestor App by creating copies of the customization objects and so on.

**4.5.2 setButtonCustomization**

This method is a variation of the `setButtonCustomization` method.

**Note: This method shall be used when the SDK implementer wants to use a button type that is not included in the predefined Enum `ButtonType`. If the button type that is specified already exists in the Enum `ButtonType` (case-insensitive match), then the button type available in the Enum is used.**

**Table 4.28: setButtonCustomization Parameters**

Parameters	Mandatory?	Description
buttontype	Yes	Implementer-specific button type. <b>The value of the <code>ButtonType</code> is case-insensitive.</b>

## 4.5.6 getButtonCustomization

### getButtonCustomization Return Value

This method returns a `ButtonCustomization` object. If the `ButtonCustomization` object was not set, then this method returns null.

## 4.5.7 getButtonCustomization

### getButtonCustomization Return Value

This method returns a `ButtonCustomization` object. If the `ButtonCustomization` object was not set, then this method returns null.

## 4.5.8 getToolbarCustomization

### getToolbarCustomization Return Value

This method returns a `ToolbarCustomization` object. If the `ToolbarCustomization` object was not set, then this method returns null.

## 4.5.9 getLabelCustomization

### getLabelCustomization Return Value

This method returns a `LabelCustomization` object. If the `LabelCustomization` object was not set, then this method returns null.

## 4.5.10 getTextBoxCustomization

### getTextBoxCustomization Return Value

This method returns a `TextBoxCustomization` object. If the `TextBoxCustomization` object was not set, then this method returns null.

## 4.10.4 getBorderColor

```
Public intString getBorderColor()
```

## 4.23 Enum Button Type

The usage of these button types is shown in section 4.2.2 Native UI Templates in the EMV 3DS Protocol Specification.

```
public enum ButtonType {VERIFYSUBMIT, CONTINUE, NEXT, CANCEL, RESEND}
```

Table 4.1: ButtonType Enum

Button Type	Description
<del>VERIFY</del>	Verify button
SUBMIT	Submit button



---

## Chapter 7 User Interface

### 7.2.1 Input and Output Formats for Native UI

#### Single Text Input

Challenge Data Entry field: ~~432525~~

```
"challengeDataEntry": "432525"
```

#### Single Select

Challenge Selection Information field (JSON):

```
{"challenge-selection-information": [
  {"mobilephone": "Mobile **** * 3291"},
  {"email": "Email sa*****tg**@g***.com"}
]
```

Challenge Data Entry field: ~~mobile~~

```
"challengeDataEntry": "phone"
```

#### Multi Select (Checkbox)

Challenge Selection Information field (JSON):

```
{"challenge-selection-information": [
  {"chicago_illinois": "Chicago, Illinois"},
  {"st_louis_missouri": "St Louis, Missouri"},
  {"portland_oregon": "Portland, Oregon"}
]
```

Challenge Data Entry field (comma-separated):

```
"challengeDataEntry": "chicago_illinois,portland_oregon"
chicago_illinois,portland_oregon
```

**Note: For Text, Single Select, and Multi Select challenge types, the Challenge Data Entry field is not included in the CReq message when challenge data has not been entered.**

---

## Chapter 8 SDK Security

### [Req 68]

The SDK shall maintain a lookup of the protocol versions that it supports and use this lookup to identify the latest **protocol** version.

## Annex D

### D.1 Code Sample for iOS, D.2 Code Sample for Android, D.3 Code Sample for Windows Phone

Within code samples, updated all instances of:

~~en-US~~ to **en-US**

# 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