EMV® Specification Bulletin No. 285

May 2023

EMV® 3-D Secure SDK—Device Information Data Version 1.6

This Specification Bulletin No. 285 describes changes incorporated into the EMV® 3-D Secure SDK—Device Information Data Version 1.6, May 2023.

Applicability

This Specification Bulletin applies to:

- *EMV® 3-D Secure SDK—Device Information Data Version 1.6*

Updates are provided in the order in which they appear in the specification. Deleted text is identified using strikethrough, and red font is used to identify added text. Unedited text is provided only for context.

Effective Date

- *May 2023*

Contents

Throughout Specification

- Revisions added to improve grammar, consistency, clarity and readability without any effect on the meaning or interpretation of the specification are not included in this specification bulletin, except when provided as context for other, more substantive revisions.
- Instances of SDK have been replaced with 3DS SDK.
- To ensure consistency with other recently published 3-D Secure documents, the *EMV 3-D Secure Protocol and Core Functions Specification* is referred to as the *Core Specification*.
- References and bookmarks have been updated to accommodate deleted/added sections and tables.

Chapter 1 Introduction

The 3-D Secure protocol is aimed at securing authentication in Browser-based and ~~mobile~~ aApp-based transactions. The *EMV 3-D Secure—Protocol and Core Functions Specification* (hereinafter also referred to as the *Core Specification*) describes the 3-D Secure protocol and core functions.

The 3DS ~~Mobile~~ SDK is the ~~mobile~~ device-side component of 3-D Secure. The *EMV 3-D Secure—SDK Specification* describes the specification for the 3DS SDK.

Device identification is used to uniquely identify ~~mobile~~ platform devices in the 3-D Secure ecosystem.

1.4 Abbreviations

Table 1.1 Abbreviations

Abbreviation	Description
ABI	Application Binary Interface
API	Application Programming Interface
LoA	Letter of Approval
MCC	Mobile Country Code
MNC	Mobile Network Code
NITZ	Network Identity and Time Zone

Chapter 2 Device Information Parameters

This chapter describes the device identification parameters that shall be collected by the 3DS SDK from all ~~mobile~~ device platforms. These parameters are categorised as ~~device platform~~ Common parameters, Platform-specific parameters and Platform Provider-specific parameters ~~that are common to all device platforms~~.

The 3DS SDK shall collect ~~and provide to the 3DS Server~~ either ~~the~~:

- for the Default-SDK – the Common parameters (see Section ~~2.4~~2.5) and one set of ~~the Device~~ Platform-specific parameters (See Section ~~2.5~~2.6 for Android~~,~~ and Section ~~2.6~~2.7 for iOS~~, Section 2.7 for Windows~~),

  OR

- for the Split-SDK – the Platform Provider-specific parameters (see Section 2.8)~~.~~,

then prepare and encrypt the Device Information as defined in Requirements 2 through 5 in the Core Specification.

The ACS uses the Device Information for device identification and risk analysis.

~~All parameters shall be encoded as String or Array of String.~~

The 3DS SDK shall collect all the parameters listed in the applicable tables: Table 2.2, Table 2.3, Table 2.4 OR Table 2.5, unless the parameter cannot be collected for any of the reasons stated in Table 2.6.

~~**Note: The availability of a higher number of device parameters improves the effectiveness of risk-based decision-making by the ACS. This, in turn, increases the probability of applying a Frictionless Flow.**~~

Note: If collecting Platform Provider-specific parameters from Section 2.8, the parameters defined in Sections 2.5, 2.6 and 2.7 are not collected.

2.1 Data Version

The Data Version may change when, for example, there are parameter changes in future ~~mobile~~device OS versions, an existing parameter is deprecated, etc.

The device identification parameters that are described in this document constitute Data Version 1.~~5~~6.

2.2 Minimum Supported Platform Versions

Table 2.1 Minimum Supported Platform Versions

Platform	Minimum Version
Android	Android ~~8~~10 (API version ~~26~~29)
iOS	~~12~~15

Note: The 3DS SDK can elect to support older OS versions ~~of the OS if the versions are supported by OS providers~~as long as the OS vendors provide security fixes for these versions.

[Section 2.4 is an entirely new section, added in Version 1.6]

2.4 Data Format and Representation

All parameters shall be encoded as String or Array of String.

Boolean device parameters are coded as String of “false” or “true”.

Integer parameters are coded as String with the format (-) integer part.

- Integer may only have a negative sign (-), the positive sign (+) is not allowed.
- Integer has no leading zeros.

Examples: 2, -3

Floating-point and Double Floating-point number parameters are represented as String with the format (-) integer par, dot (.) decimal part.

- It may only have a negative sign (-), the positive sign (+) is not allowed.
- The integer part has no leading zeros.
- The decimal part has no trailing zeros.
- If the integer part is null, then it is represented by 0 (zero).
- If the decimal part is null, then the dot and decimal part are not present.

Examples: 0, 1.2, -1.2, 0.12, 12

~~2.4~~2.5 Common Device Identification Parameters Available in All ~~Mobile~~ Device Platforms

~~In addition to the platform-specific device parameters discussed later in this document,~~ Table 2.2 lists the ~~device~~ Common parameters that the 3DS SDK shall ~~be collected~~ collect from all ~~mobile platforms~~ (Android~~,~~ and iOS ~~and Windows) for Device ID validation and risk analysis~~ platforms. ~~For Platform Provider-specific parameters, refer to Section 2.8.~~

The availability of these parameters is subject to change in future OS versions.

~~**Note: Each parameter listed in this table shall be collected by the SDK unless the parameter cannot be collected for any of the reasons stated in Table 2.7.**~~

Table 2.2 Common Parameters Available in Android~~,~~ and iOS ~~and Windows 10~~ Platforms

Identifier	Parameter	Description	Permissions
C001		Platform that the device is using. JSON Data Type: String Values accepted: • "Android" • "iOS" • ~~"Windows 10"~~	
C002		Mobile device manufacturer and model. JSON Data Type: String Values accepted: • Android: `Build.MANUFACTURER` + " " "||" + `Build.MODEL` returns the mobile device manufacturer and model. Example: "~~s~~Samsung||SM-G960U1" • iOS: `utsname.machine` returns the device model. Example: "iPhone10.4" Note: Apple as a manufacturer is not included because it is the same for all iOS devices.	

Identifier	Parameter	Description	Permissions
C003		Operating system name. JSON Data Type: String Values accepted: • Android: “Android” + “ ” + (Build.Version.SDK_INT equivalent field name from Build.Version.VERSION_CODES) + “ ” + Build.Version.RELEASE + “ API ” + Build.Version.SDK_INT returns the name of the operating system and the API level.~~, for e~~Example~~, the following format~~: “Android Q 10 API 29”	
C004		Operating system version. JSON Data Type: String Values accepted: • Android: Build.VERSION.RELEASE returns the version of the operating system. Example: “8.1.0” • iOS: the systemVersion property of the UIDevice class returns the version of the operating system. Example: “14.2”	
C005		Device locale set by the user. For more information, refer to IETF BCP 47. JSON Data Type: String Values accepted: • Android: the device locale.Language() + “-” + locale.getCountry() returns the device locale.~~, for e~~Example~~, the following format~~: “en-US”. • iOS: the device currentLocale.languageCode + “-” + currentLocale.countryCode returns the device locale.~~, for e~~Example~~, the following format~~: “en-US”.	

Identifier	Parameter	Description	Permissions
C006	Time Zzone	Time zone offset in minutes between UTC and the device local time JSON Data Type: String Length: Variable, 1–4 characters Value accepted: • Integer in the range of -720 to 840, coded as a string If UTC -5 hours: • "300" • ~~"+300"~~	
C008		Pixel width and pixel height.~~, i.e. "1080x1920".~~ JSON Data Type: String Length: Variable, maximum 13 characters Both width and height: Integer in the range of 0 to 999999 Expressed as width x height, for example: "1080x1920".	
C009		User-assigned device name. JSON Data Type: String	On Android, this parameter requires ~~Bluetooth permission during installation~~ Installation-time permissions AND Run-time permissions. No permissions required on iOS ~~or Windows 10.~~
C010		Local IP address of the 3DS SDK in IPv4 or IPv6 format. JSON Data Type: String Length: Variable, maximum 45 characters Values accepted: • IPv4 address. Refer to RFC 791. • IPv6 address. Refer to RFC 4291.	No permissions required on iOS ~~or Windows 10~~.

Identifier	Parameter	Description	Permissions
C011		Device physical location latitude. JSON Data Type: String Value accepted: • Double Floating-point number coded as a string Range: -90 to 90	Run-time permissions required on Android API level 23 and later, and iOS ~~and Windows 10~~. Installation-time permissions required on Android API level 22 and earlier.
C012		Device physical location longitude. JSON Data Type: String Value accepted: • Double Floating-point number coded as a string Range: -180 to 180	Run-time permissions required on Android API level 23 and later, and iOS ~~and Windows 10~~. Installation-time permissions required on Android API level 22 and earlier.
C013		The unique package name/bundle identifier of the application in which the 3DS SDK is embedded. JSON Data Type: String	
C014		Universally unique ID that is created for each installation of the 3DS Requestor App on a Consumer Device. JSON Data Type: String Length: 36 characters Note: This should be the same ID that is passed to the 3DS Requestor App in the `AuthenticationRequestParameters` object (refer to Section 4.12.1 in the *EMV ~~3DS~~ 3-D Secure—SDK Specification*).	
C015		3DS SDK version as applied by the implementer and stored securely in the 3DS SDK (refer to Req 58 in the *EMV ~~3DS~~ 3-D Secure—SDK Specification*). JSON Data Type: String	

Identifier	Parameter	Description	Permissions
C016		Identifies the vendor and version of the 3DS SDK that is used for a specific transaction. The value is assigned by EMVCo when the Letter of Approval (LoA) of the specific 3DS SDK is issued. JSON Data Type: String Length: Variable, maximum 32 characters Note: The ACS should verify that this value matches the SDK Reference Number present in the AReq message.	
C017		Date and time when the 3DS SDK gathers the Device Information converted into UTC. Refer to the *Core Specification* for the definition of UTC. JSON Data Type: String Length: 14 characters Format accepted: YYYYMMDDHHMMSS	No permissions required
C018		Universally unique transaction identifier assigned by the 3DS SDK to identify a single transaction. Refer to the *Core Specification* for the definition of the SDK Transaction ID. JSON Data Type: String Length: 36 characters Note: The `sdkTransID` is added to the Device Information before the 3DS SDK encrypts the data. It is updated every time the `createTransaction` method is invoked.	No permissions required

~~2.5~~2.6 Android-specific Device Parameters

~~Table 2.3 provides information about~~ Table 2.3 lists the ~~device~~ Platform-specific parameters that ~~shall be collected by~~ the 3DS SDK shall collect from the Android ~~mobile~~ device platform ~~for risk analysis by the ACS~~. The Group or Identifier column contains the name of the parameter group or parameter identifier.

Table 2.3 Android-specific Parameters

Group or Identifier	Element	Description	Comments	Permissions
A001		Unique identifier of the device. Example: • IMEI for GSM phones ~~and~~ • MEID or ESN for CDMA phones. JSON Data Type: String		
A002		Unique subscriber ID. JSON Data Type: String		
A003		IMEI software version. JSON Data Type: String		
A004		Group identifier level 1 for a GSM phone. JSON Data Type: String		
A005		Phone number string for line 1. JSON Data Type: String		
A006		MMS user agent profile URL. JSON Data Type: String		
A007		MMS user agent. JSON Data Type: String		
A008		ISO country code equivalent of the current registered operator's Mobile Country Code (MCC). Length: 2 characters JSON Data Type: String Value accepted: • The ISO-3166-1 alpha-2 country code equivalent of the MCC.		

Group or Identifier	Element	Description	Comments	Permissions
A009		Numeric name ~~(mobile country code + mobile network code)~~ (MCC + Mobile Network Code (MNC)) of the current registered operator. JSON Data Type: String		
A010		Alphabetic name of the current registered operator. JSON Data Type: String		
A011		NETWORK_TYPE_xxxx for the current data connection. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Integer coded as a string		
A012		Number of phones available. Returns 1 for single standby mode (single SIM functionality). Returns 2 for dual standby mode (dual SIM functionality). Length: 1 character JSON Data Type: String Value accepted: • Integer in the range of 0 to 5, coded as a string	~~Use~~From API level 30 onwards use getActiveModemCount() ~~for API level 30 and higher~~.	
A013		Constant that indicates the device phone type. This indicates the type of radio used to transmit voice calls. JSON Data Type: String		

Group or Identifier	Element	Description	Comments	Permissions
A014		ISO country code equivalent for the SIM provider's country code. Length: 2 characters JSON Data Type: String Value accepted: • The ISO-3166-1 alpha-2 country code equivalent of the SIM provider's country code		
A015		MCC+MNC ~~(mobile country code + mobile network code)~~ of the SIM provider. JSON Data Type: String Length: Variable, maximum 6 numeric characters		
A016		Service Provider Name (SPN). JSON Data Type: String		
A017		Serial number of the SIM, if applicable. JSON Data Type: String		
A018		Constant that indicates the state of the default SIM card. Length: 1 character JSON Data Type: String Value accepted: • Integer in the range of 0 to 9, coded as a string		
A019		Alphabetic identifier associated with the voice mail number. JSON Data Type: String		

Group or Identifier	Element	Description	Comments	Permissions
A020		Voice mail number. JSON Data Type: String		
A021		Returns true if an Integrated Circuit Card (ICC ~~card~~) is present. JSON Data Type: String Values accepted: • “false” • “true”		
A022		Indicates whether the phone supports hearing aid compatibility. JSON Data Type: String Values accepted: • “false” • “true”		
A023		Determines if the device is considered roaming on the current network, for GSM purposes. JSON Data Type: String Values accepted: • “false” • “true”		
A024		Determines if the current device supports SMS service. JSON Data Type: String Values accepted: • “false” • “true”		

Group or Identifier	Element	Description	Comments	Permissions
A025		Determines whether the phone supports TTY mode. JSON Data Type: String Values accepted: • “false” • “true”	From API level 28 onwards use TelecomManager.isTtySupported() ~~. to be used instead from API level 28 onwards~~.	
A026		Determines if the current device is “voice capable”. JSON Data Type: String Values accepted: • “false” • “true”		
A027		Determines whether the device is a world phone. JSON Data Type: String Values accepted: • “false” • “true”		
A138		Provides a platform-wide unique identifier for each carrier. Length: Variable, maximum 11 characters JSON Data Type: String Values accepted: • Integer coded as a string		
A139		Provides user-facing name of the specific carrier ~~id~~ID JSON Data Type: String		

Group or Identifier	Element	Description	Comments	Permissions
A140		Provides the Manufacturer code from the Mobile Equipment Identifier JSON Data Type: String		
A141		Provides carrier ID of the current subscription Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Integer coded as a string		
A142		Provides the user-facing name of the specific carrier ID JSON Data Type: String		
A143		~~Returns~~ Indicates if the ability to register multiple SIM cards simultaneously on the network is supported by the device and by the carrier. Length: 1 character JSON Data Type: String Values accepted: • “0” • “1” • “2”	Available only for API 29 or higher	~~No permissions required~~ Installation-time permissions
~~A144~~	~~networkCountryIso~~	~~Returns the ISO-3166-1 alpha-2 country code equivalent of the Mobile Country Code (MCC) of the current registered operator.~~	~~Available only for API 30 or higher~~	~~No permissions required~~

Group or Identifier	Element	Description	Comments	Permissions
A145		Returns the subscription ID for the given phone account. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Integer coded as a string		
WiFiManager				
A028		Returns the wireless MAC address of the device. JSON Data Type: String	For API level 31 or higher, set to RE04 in the DPNA.	
A029		Returns the ~~basic service set identifier~~Basic Service Set Identifier (BSSID) of the current access point. JSON Data Type: String		
A030		Returns the ~~service set identifier~~Service Set Identifier (SSID) of the current 802.11 network. JSON Data Type: String		
A031		Each configured network has a unique small integer ID, used to identify the network when performing operations on the supplicant. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A032		Determines if this adapter supports the 5 GHz band. JSON Data Type: String Values accepted: • "false" • "true"		
A033		Determines if this adapter supports Device-to-AP RTT. JSON Data Type: String Values accepted: • "false" • "true"		
A034		Determines if this adapter supports advanced power and performance counters. JSON Data Type: String Values accepted: • "false" • "true"		
A035		Determines if this adapter supports WifiP2pManager. JSON Data Type: String Values accepted: • "false" • "true"		
A036		Determines if this adapter supports offloaded connectivity scan. JSON Data Type: String Values accepted: • "false" • "true"		

Group or Identifier	Element	Description	Comments	Permissions
A037		Determines if scanning is always available. JSON Data Type: String Values accepted: • "false" • "true"		
A038		Determines if this adapter supports Tunnel Directed Link Setup. JSON Data Type: String Values accepted: • "false" • "true"		
A146		Returns a Boolean ~~(coded as a string "0" or "1")~~ if 6GHz band is supported. JSON Data Type: String Values accepted: • "false" • "true"		
A147		Returns the ~~Fully Qualified Domain Name~~ fully qualified domain name of the network if it is a Passpoint network. JSON Data Type: String		
A148		Returns the Provider Friendly Name of the network if it is a Passpoint network. JSON Data Type: String		
Bluetooth Manager				

Group or Identifier	Element	Description	Comments	Permissions
A039		Hardware MAC address of the local Bluetooth adapter. Example: 00:00:56:B1:C0:6E JSON Data Type: String Values accepted: • 48 bits represented as 6 hexadecimal bytes separated by ":" (colon hexadecimal notation)		Installation-time permissions AND Run-time permissions
A040		Returns the array of BluetoothDevice MAC address ~~coded as string~~ that are bonded (paired) to the local adapter. ~~For e~~Example: :["48:F0:7B:61:DD:D4","ED:90:C2:3D:E8:14"] JSON Data Type: Array of String Values accepted: • 48 bits represented as 6 hexadecimal bytes separated by ":" (colon hexadecimal notation)		Installation-time permissions AND Run-time permissions
A149		Returns the array of BluetoothDevice alias coded as a string that are bonded (paired) to the local adapter. JSON Data Type: Array of String		Installation-time permissions AND Run-time permissions

Group or Identifier	Element	Description	Comments	Permissions
A041		Returns true if Bluetooth is currently enabled and ready for use. JSON Data Type: String Values accepted: • "false" • "true"		
Build				
A042-A052		[*added at the end for each*] JSON Data Type: String		
A053		JSON Data Type: String	Deprecated in API level 26.~~; getSerial() to be used instead from~~ From API level 26 onwards use getSerial().	
A153	SKU	The SKU of the hardware (from the kernel command line). JSON Data Type: String	Available only for API level 31 or higher.	No permissions required
A154	SOC_MANUFACTURER	The manufacturer of the device's primary system-on-chip. JSON Data Type: String	Available only for API level 31 or higher.	No permissions required
A155	SOC_MODEL	The model name of the device's primary system-on-chip. JSON Data Type: String	Available only for API level 31 or higher.	No permissions required
A054		Ordered list of 32-bit Application Binary Interfaces (ABIs) supported by this device. The most preferred ABI is the first element in the list. JSON Data Type: Array of String		

Group or Identifier	Element	Description	Comments	Permissions
A055		Ordered list of 64-bit ABIs supported by this device. The most preferred ABI is the first element in the list. JSON Data Type: Array of String		
A056		Comma-separated tags describing the build, such as "unsigned,debug". JSON Data Type: String		
A057		Build time. JSON Data Type: String Length: Variable, maximum 20 characters Value accepted: • Positive long integer coded as a string		
A058		Type of build, such as "user" or "eng". JSON Data Type: String		
A059		JSON Data Type: String		
Build.VERSION				
A060		The current development codename, or the string "REL" if this is a release build. JSON Data Type: String		
A061		The internal value used by the underlying source control to represent this build. JSON Data Type: String		

Group or Identifier	Element	Description	Comments	Permissions
A062		The developer preview revision of a pre-release SDK. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A063		The user-visible SDK version of the framework; its possible values are defined in Build.VERSION_CODES. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A064		The user-visible security patch level. JSON Data Type: String		
Settings Secure				
A065		Specifies whether display colour inversion is enabled. JSON Data Type: String Values accepted: • “false” • “true”		

Group or Identifier	Element	Description	Comments	Permissions
A066		Specifies whether accessibility is enabled. JSON Data Type: String Values accepted: • "false" • "true"		
A067		Specifies whether to speak passwords while in accessibility mode. JSON Data Type: String Values accepted: • "false" • "true"		
A068		Origins for which browsers should allow geolocation by default. The value is a space-separated list of origins. JSON Data Type: String		
A069		A 64-bit number (expressed as a hexadecimal string) ~~that is randomly generated when the end~~, unique to each combination of app-signing key, user ~~first sets up the,~~ and device. ~~This number should remain constant for the lifetime of the end user's device.~~ Values of ANDROID_ID are scoped by signing key and user. The value may change if a factory reset is performed on the device or if an APK signing key changes. JSON Data Type: String Value accepted: • 8-byte hexadecimal [0-9,a-f,A-F]		

Group or Identifier	Element	Description	Comments	Permissions
A071		Setting to record the input method used by default. JSON Data Type: String		
A073		List of enabled accessibility providers. JSON Data Type: Array of String		
A074		List of input methods that are currently enabled. JSON Data Type: Array of String		
A075		Setting to record the visibility of the input method selector. JSON Data Type: String		
A076		Specifies whether applications can be installed for this user via the system's ACTION_INSTALL_PACKAGE mechanism. ~~Encoded as either~~JSON Data Type: String Values accepted: • "false" • "true"	From API level 29 onwards use PackageManager.canRequestPackageInstalls() Note: canRequestPackageInstalls() needs Run-time permissions (android.permission.REQUEST_INSTALL_PACKAGES). If no permission, set to ~~be used instead from API level 29 onwards~~ RE03 in the DPNA.	
A077		Degree of location access enabled by the end user. JSON Data Type: String	From API level 28 onwards use LocationManager.isLocationEnabled() ~~to be used instead from API level 28 onwards.~~	

Group or Identifier	Element	Description	Comments	Permissions
A078		If enabled, apps should try to skip any introductory hints on first launch. JSON Data Type: String Values accepted: • "false" • "true"		
A079		Secure system settings, containing system preferences that applications can read but are not allowed to write. JSON Data Type: String		
A080		Default text-to-speech engine pitch. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A081		Default text-to-speech engine speech rate. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A082		Default text-to-speech engine. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A083		Space-delimited list of plugin packages that are enabled. JSON Data Type: String		
A150		User-selected Real Time Text (RTT) mode. ~~Boolean coded as a string "0" or "1".~~ JSON Data Type: String Values accepted: • "false" • "true"		
A151		Indicates whether the device is ~~under~~ in restricted secure Factory Reset Protection (FRP) mode. ~~Boolean coded as a string "0" or "1".~~ JSON Data Type: String Values accepted: • "false" • "true"		
Settings Global				
A084		Specifies whether Android Debug Bridge (ADB) is enabled. JSON Data Type: String Values accepted: • "false" • "true"		
A085		Comma-separated list of radios that need to be disabled when airplane mode is on. JSON Data Type: String		

Group or Identifier	Element	Description	Comments	Permissions
A086		If 1, the activity manager will aggressively finish activities and processes as soon as they are no longer needed. JSON Data Type: String Values accepted: • “false” • “true”		
A087		Scaling factor for animator-based animations. JSON Data Type: String Value accepted: • Floating-point number coded as a string		
A088		Value to specify whether the user prefers the date, time and time zone to be automatically fetched from the network. Refer to Network Identity and Time Zone (NITZ). JSON Data Type: String Values accepted: • “false” • “true”		
A089		Value to specify whether the user prefers the time zone to be automatically fetched from the network. ~~(~~Refer to NITZ~~).~~ JSON Data Type: String Values accepted: • “false” • “true”		

Group or Identifier	Element	Description	Comments	Permissions
A070		Determines whether ~~or not~~ data roaming is enabled. JSON Data Type: String Values accepted: • “false” • “true”	Available only for API level 17 or higher. Note: `TelephonyManager.isDataRoamingEnabled()` may also be used.	
A090		Determines whether the end user has enabled development settings. JSON Data Type: String Values accepted: • “false” • “true”		
A072		Determines whether the device has been provisioned. JSON Data Type: String Values accepted: • “false” • “true”		
A091		Host name and port for global ~~http~~HTTP proxy. JSON Data Type: String		
A092		User preference for which networks should be used. JSON Data Type: String		
A093		Determines whether the device must remain switched on while it is plugged in. JSON Data Type: String Value accepted: • Integer in the range of 0 to 15, coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A094		Scaling factor for activity transition animations. JSON Data Type: String Value accepted: • Floating-point number coded as a string		
A095		Indicates whether USB mass storage is enabled. JSON Data Type: String Values accepted: • “false” • “true”		
A096		If this setting is set (to anything), then all references to Gmail on the device must change to Google Mail. JSON Data Type: String		
A097		If 1, when launching DEBUG_APP, it will wait for the debugger before starting user code. JSON Data Type: String Values accepted: • “false” • “true”		
A098		Determines whether the end user should be notified of open networks. JSON Data Type: String Values accepted: • “false” • “true”		

Group or Identifier	Element	Description	Comments	Permissions
A152		~~Returns~~ Indicates if ramping ringer is enabled on incoming call ringtone~~. Boolean coded as a string "0" or "1".~~ JSON Data Type: String Values accepted: • "false" • "true"	Available only for API level 29 or higher, deprecated in API level 33. From API level 33 onwards, use `AudioManager.isRampingRingerEnabled()` "	
Settings System				
A099		Control whether the accelerometer will be used to change screen orientation. JSON Data Type: String Values accepted: • "false" • "true"		
A100		Determines whether remote devices may discover and/or connect to this device. JSON Data Type: String Values accepted: • "0" • "1" • "2"		
A101		Bluetooth discoverability timeout. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A102		Date format~~: string~~. JSON Data Type: String Values accepted: • mm/dd/yyyy • dd/mm/yyyy • yyyy/mm/dd	Deprecated in API level 31 From API level 31 onwards, use A120 - TIME_12_24.	
A103		CDMA-only settings + DTMF tone type played by the dialler when dialling. JSON Data Type: String Values accepted: • “false” • “true”		
A104		Specifies w~~W~~hether the audible DTMF tones are played by the dialler when dialling. JSON Data Type: String Values accepted: • “false” • “true”		
A105		The behaviour when the user presses the end call button if they are not on a call. JSON Data Type: String Values accepted: • “0” • “1” • “2” • “3”		

Group or Identifier	Element	Description	Comments	Permissions
A106		Scaling factor for fonts~~, float~~. JSON Data Type: String Value accepted: • Positive Floating-point number coded as a string		
A107		Specifies w~~W~~hether the haptic feedback (long presses) is enabled. JSON Data Type: String Values accepted: • “false” • “true”	Deprecated in API level 33	
A108		Determines which streams are affected by ringer mode changes. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A109		Persistent store for the system-wide default notification sound. JSON Data Type: String		
A110		Determines which streams are affected by mute. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A111		Persistent store for the system-wide default ringtone URI. JSON Data Type: String		
A112		~~The screen~~Screen backlight brightness between 0 and 255. Length: Variable, maximum 3 characters JSON Data Type: String Value accepted: • Integer in the range of 0 to 255, coded as a string		
A113		Control whether to enable automatic brightness mode. JSON Data Type: String Values accepted: • “false” • “true”		
A114		The amount of time in milliseconds before the device goes to sleep or begins to dream after a period of inactivity. This value is also known as the user activity timeout period since the screen is not necessarily turned off when it expires. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A115		Specifies w~~W~~hether sound effects (key clicks, lid open) are enabled. JSON Data Type: String Values accepted: • “false” • “true”		
A116		Setting to enable Auto Caps in text editors. JSON Data Type: String Values accepted: • “false” • “true”		
A117		Setting to enable Auto Punctuate in text editors. JSON Data Type: String Values accepted: • “false” • “true”		
A118		Setting to enable Auto Replace (AutoText) in text editors. JSON Data Type: String Values accepted: • “false” • “true”		
A119		Setting to show password characters in text editors. JSON Data Type: String Values accepted: • “false” • “true”		

Group or Identifier	Element	Description	Comments	Permissions
A120		Display time in the 12-hour format or the 24-hour format. JSON Data Type: String Values accepted: • “12” • “24”		
A121		Default screen rotation when no other policy applies. JSON Data Type: String Values accepted: • “0” • “1” • “2” • “3”		
A122		Specifies w~~W~~hether vibrate is on for different events. JSON Data Type: String Values accepted: • “false” • “true”		
A123		Specifies w~~W~~hether the phone vibrates when it is ringing during an incoming call. JSON Data Type: String Values accepted: • “false” • “true”	Available only for API level 23 or higher. Deprecated in API level 33.	
Package Manager				

Group or Identifier	Element	Description	Comments	Permissions
A124		Returns whether the device has been booted into safe mode. JSON Data Type: String Values accepted: • "false" • "true"		
A125		Returns an array of non-system application packages that are installed on the device. JSON Data Type: Array of String		~~No~~Run-time permissions ~~required~~
A126		Retrieves the package name of the application that installed a package. This identifies which market the package came from. JSON Data Type: String	~~Starting~~ From API level 30 onwards, use getInstallSourceInfo().	
A127		Retrieves a list of features that are available on the device. The 3DS SDK shall share only the count of items in this list and not the full list itself. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A128		Retrieves a list of shared libraries that are available on the device. The 3DS SDK shall share only the count of items in this list and not the full list itself. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
Environment				
A129		Returns the current state of the primary shared/external storage media. JSON Data Type: String		
Locale				
A130		Returns the system's installed locales. The 3DS SDK shall share only the length of this list and not the full list itself. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
DisplayMetrics				

Group or Identifier	Element	Description	Comments	Permissions
A131		The logical density of the display. JSON Data Type: String Value accepted: • Floating-point number coded as a string		
A132		The screen density expressed as dots per inch. Length: Variable, maximum 11 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
A133		A scaling factor for fonts displayed on the display. JSON Data Type: String Value accepted: • Floating-point number coded as a string		
A134		The exact physical pixels per inch of the screen in the X dimension. JSON Data Type: String Value accepted: • Floating-point number coded as a string		

Group or Identifier	Element	Description	Comments	Permissions
A135		The exact physical pixels per inch of the screen in the Y dimension. JSON Data Type: String Value accepted: • Floating-point number coded as a string		
StatFs				
A136		The total number of bytes supported by the filesystem. Length: Variable, maximum 19 characters JSON Data Type: String Value accepted: • Positive integer coded as a string		
WebView				
A137		The default user agent of the WebView component during the App-based HTML flow. String `defaultUserAgent = android.webkit.WebSettings.getDefaultUserAgent(context);` JSON Data Type: String		

~~2.6~~2.7 iOS-specific Device Parameters

Table 2.4 ~~provides information about~~ lists the ~~device~~ Platform-specific parameters that ~~shall be collected by~~ the 3DS SDK shall collect from the iOS ~~mobile~~ device platform ~~for risk analysis by the ACS~~.

~~In this table, the~~ The Group or Identifier column contains the name of the parameter group or parameter identifier.

Note: The 3DS SDK does not require any permissions to collect these parameters.

~~**Note: Each parameter listed in this table shall be collected by the 3DS SDK unless the parameter cannot be collected for any of the reasons stated in Table 2.6.**~~

Table 2.4 iOS-specific Parameters

Group or Identifier	Attribute	Description
UIDevice		
I001		Alphanumeric string that uniquely identifies a device to the app's vendor. JSON Data Type: String
I002		Style of interface to use on the current device. JSON Data Type: String
UIFont		
I003		Returns an array of font family names available on the system. JSON Data Type: Array of String
I004		Returns an array of font names for all the font families listed in I003. JSON Data Type: Array of String
I005		~~System~~Returns the system font. JSON Data Type: String
I006		Returns the standard font size used for labels. JSON Data Type: String Value accepted: • Floating-point number ~~represented~~ coded as ~~a~~ string~~, e.g., '10.5'~~
I007		Returns the standard font size used for buttons. JSON Data Type: String Value accepted: • Floating-point number ~~represented~~ coded as ~~a~~ string~~, e.g. "18"~~.
I008		Returns the size of the standard small system font. JSON Data Type: String Value accepted: • Floating-point number coded as a string

Group or Identifier	Attribute	Description
I009		Returns the size of the standard system font. JSON Data Type: String Value accepted: • Floating-point number coded as a string
NSLocale		
I010		Formatted as the device locale language + "-" + device locale country. For example, "en-US". JSON Data Type: String
I011		Returns an array of string as provided by the OS method, each of which identifies a locale available on the system. JSON Data Type: Array of String
I012		Returns the user's language preference order as an array of string as provided by the OS method. JSON Data Type: Array of String
NSTimeZone		
I013		Returns the time zone offset in minutes between UTC and the default time zone for the current application. JSON Data Type: String Length: Variable, 1–4 characters Value accepted: • Integer in the range of -720 to 840, coded as a string Example time zone offset values in minutes: If UTC -5 hours: • "300" • ~~"+300"~~ If UTC +5 hours: • "-300"
NSBundle		

Group or Identifier	Attribute	Description
I014		The file URL for the main application bundle's App Store receipt. `[[NSBundle mainBundle] appStoreReceiptURL]` https://developer.apple.com/documentation/foundation/nsbundle/1407276-appstorereceipturl JSON Data Type: String
I015	~~Encoded as either "false" or "true"~~	Indicates whether the receipt file residing in the `appStoreReceiptURL` path exists and is non-empty. Indirectly, this field can be used to determine whether the application has been purchased from the ~~Apple~~ App ~~s~~Store. JSON Data Type: String Values accepted: • "false" • "true"

*[**Section 2.7 Windows 10-specific Device Parameters** has been deleted in its entirety. Subsequent sections, tables and references have been renumbered accordingly.]*

2.8 Platform Provider-specific Parameters

Table 2.5 ~~provides information about~~ lists the ~~device~~ Platform Provider-specific parameters that ~~shall be collected by~~ the 3DS SDK ~~for risk analysis by~~ shall collect from the ~~ACS~~ Platform Provider-specific platform.

If providing Platform Provider-specific parameters, the parameters defined in Sections 2.5, 2.6, ~~2.7~~ and ~~2.8~~2.7 shall not be provided.

Refer to the definition of the term "Platform Provider" in the *Core Specification*.

Table ~~2.6~~2.5 Platform Provider-specific Parameters

Group or Identifier	Parameter	Description	Comments
D001		Platform that the device is using~~, as a string.~~ JSON Data Type: String	
D002		Platform-defined device model~~, as a string.~~ JSON Data Type: String	
D003		Platform-defined OS name~~, as a string.~~ JSON Data Type: String	

Group or Identifier	Parameter	Description	Comments
D005		Device locale set by the user~~, as a string.~~, refer to IETF BCP 47 The Device Locale as set by the user ~~is made~~ consists of the device Language Code + "-" + current Country Code~~, for example, the following format~~: Example: "en-US". JSON Data Type: String	
D006		User-selected or platform-provisioned Time Zone for the user's device rendering the 3DS challenge. Time zone offset in minutes between UTC and the device local time as a string. JSON Data Type: String Length: Variable, 1–4 characters Value accepted: • Integer in the range of -720 to 840, coded as a string Example time zone offset values in minutes: If UTC -5 hours: • "300" • ~~"+300"~~	
D008		Pixel width and pixel height~~, as a string, i.e.,~~ JSON Data Type: String Length: Variable, maximum 13 characters Value accepted: • Both width and height: Integer in the range of 0 to 999999. Expressed as width x height, for example: "1080x1920".	
D013		The unique package name/bundle identifier of the application in which the 3DS SDK is embedded. JSON Data Type: String	
D015		3DS SDK version as applied by the implementer and stored securely in the 3DS SDK (refer to Requirement 58 in the *EMV 3-D Secure—SDK Specification*). JSON Data Type: String	

Group or Identifier	Parameter	Description	Comments
D016		Identifies the vendor and version of the 3DS SDK that is utilised for a specific transaction. The value is assigned by EMVCo when the ~~Letter of Approval (~~LoA~~)~~ of the specific 3DS SDK is issued and is provided as a string. JSON Data Type: String Note: The ACS should verify that this value matches the SDK Reference Number present in the AReq message.	
D017		Challenge window width and height in pixels~~, as a string, i.e.,~~. JSON Data Type: String Length: Variable, maximum 13 characters Value accepted: • Both width and height: Integer in the range of 0 to 999999. Expressed as width x height, for example: “500x600”.	
D021		Example: • Hardware Device ID • Platform-calculated device fingerprint JSON Data Type: String	
D022		Constant that indicates the device type. ~~Valid~~JSON Data Type: String Values accepted: • “01” = Desktop • “02” = TV-connected • “03” = Tablet/Mobile • “04” = Headless/Voice • “05” = Wearable • “06” = Internet of Things • “99” = Other	~~Not applicable~~

Group or Identifier	Parameter	Description	Comments
D023		List of cardholder input methods enabled on the device ~~as an array of string~~, i.e. [“01”, “02”]. ~~Valid values:~~ JSON Data Type: Array of String Values accepted: • “01” = Physical Keyboard • “02” = Touch Keyboard • “03” = TV-connected Onscreen Keyboard • “04” = Voice-activated • “05” = Gesture-activated • “99” = Other	
D024		List of output methods enabled on the device ~~as an array of string~~. ~~Valid values:~~ JSON Data Type: Array of String Values accepted: • “01” = Display • “02” = Audio • “03” = Monochrome Display • “99” = Other	
D025		Preferred network and issuer logo colour preference ~~provided as a string~~. ~~Valid values:~~ JSON Data Type: String Values accepted: • “01” = Full Colour • “02” = Monochrome White • “03” = Monochrome Black • “99” = Other	
D026		Identifier of the transacting user’s platform Account ID. This identifier is a unique immutable hash of the user’s account identifier for the given platform~~,~~ ~~provided as a string~~. JSON Data Type: String	

Group or Identifier	Parameter	Description	Comments
D027		Gets the set of languages preferred by the user, in order of preference provided as an array of string, as defined in IETF BCP 47. JSON Data Type: Array of String	
D028		The device identifier of the device where the transaction started before it was transferred to another device or method for/to complete authentication. JSON Data Type: String ~~Provided as a string, f~~For example:	
D029		External IP address of the device as collected by the 3DS SDK in IPv4 or IPv6 format~~, provided as string~~. JSON Data Type: String Length: Variable, maximum 45 characters Values accepted: • IPv4 address. Refer to RFC 791. • IPv6 address. Refer to RFC 4291.	
D030		Exact content of the HTTP Accept Headers as sent to the 3DS Requestor from the Cardholder Browser~~, provided as string~~. JSON Data Type: String	~~Browser-SDK only~~Only applicable to the Split-SDK/Browser. For other devices, return RE02.
D031		Exact content of the HTTP User-Agent header~~, provided as string~~. JSON Data Type: String	
D032		Information about the Device ID, for example: “03”. ~~Valid values:~~ JSON Data Type: String Values accepted: • “01” = Hardware-based identifier • “02” = Hardware fingerprint identifier • “03” = Key-based software identifier • “04” = Software fingerprint identifier • ~~Provided as a string, for example: “03”~~	

Group or Identifier	Parameter	Description	Comments
D033		Information about the Device ID, for example: “02”. ~~Valid values:~~ JSON Data Type: String Values accepted: • “01” = Hardware-based identifier • “02” = Hardware fingerprint identifier • “03” = Key-based software identifier • “04” = Software fingerprint identifier • ~~Provided as a string, for example: “02”~~	
D034	dateTime	Date and time when the 3DS SDK gathers the Device Information converted into UTC. Refer to the *Core Specification* for the definition of UTC. JSON Data Type: String Length: 14 characters Format accepted: YYYYMMDDHHMMSS	
D035	sdkTransID	Universally unique transaction identifier assigned by the 3DS SDK to identify a single transaction. Refer to the *Core Specification* for the definition of the SDK Transaction ID. JSON Data Type: String Length: 36 characters Note: The sdkTransID is added to the Device Information before the 3DS SDK encrypts the data. It is updated every time the createTransaction method is invoked.	

2.9 Reasons for Device Parameter Unavailability

[The following note was added at the end of section introduction.]

Note: The availability of a higher number of device parameters improves the effectiveness of risk-based decision-making by the ACS, which may increase the probability of applying a Frictionless Flow.

Table 2.6: Device Parameter Unavailability Reasons

Reason Code	Description
RE01	Market, regional or privacy restriction on the parameter.

2.10 Device Information JSON Data

Table ~~2.8~~2.7 Device Parameters JSON Structure

Platform	Device Information
Android	{"DV":"1.~~5~~6","DD":{"C001":"Android","C002":"HTC||One_M8","C004":"5.0.1","C005":"en-US","C006":"-300","C008":"400x800","C009":"John's Android Device",....},"DPNA":{"C010":"RE01","C011":"RE03"},"SW":["SW01","SW04"]}
iOS	{"DV":"1.~~5~~6","DD":{"C001":"iOS","C002":"iPhone6,1","C003":" iPhone OS ","C004":"9.2","C005":"en-US","C006":"360","C008":"800x2000","C009":"John's iPhone",....},"DPNA":{"C010":"RE01","C011":"RE03"},"SW":["SW01","SW04"]}
~~Windows 10~~	~~{"DV":"1.50","DD":{"C001":"Windows","C002":"NOKIA RM-984_1006","C003":"WindowPhone","C004":"10.0.10586.11","C005":"en-US","C006":"(UTC-06:00) Central Time (US & Canada)+120","C007":"1bbd95da4520a6dfc7b94480d69f3cbb","C008":"1280x720","C009":"My Phone",....},"DPNA":{"C010":"RE02","C011":"RE03"},"SW":["SW01","SW04"]}~~
Platform Provider-specific	{"DV":"1.~~5~~6","DD":{"D001":"Android","D002":"Personal device","D003":"aPhone",~~"D004":"13.0.186.11",~~"D005":"fr-FR","D006":"60","D008":"2340x1080","D009":"My Phone",....},"DPNA":{"D028":"RE02","D031":"RE03"},"SW":["SW01","SW04"]}

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