EMV® Specification Bulletin No. 196
October 2017

EMV 3-D Secure Updates, Clarifications & Errata

This Draft Specification Bulletin No. 196 provides updates, clarifications and errata incorporated into the EMV 3-D Secure Protocol and Core Functions Specification since the October 2016 v2.0.0 publication. This Bulletin No. 196 provides updates by date for the two draft iterations, and the October 2017 final version of the specification.

Applicability

This Specification Bulletin applies to:

- *EMV 3-D Secure Protocol and Core Functions Specifications*, *Version 2.0.0*

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

- *Please note that draft versions of the protocol specification released since the October 2016 v2.0.0 included the version number 2.0.1. The October 2017 version number has been updated to version 2.1.0.*

Related Documents

The following publication should also be referenced for specification updates:

- *Bulletin No .190: Requirement Numbering Scheme and Error Processing, December, 2016*

The following publications have been updated since their initial release:

- *EMV 3-D Secure SDK Specification*
- *EMV 3-D Secure SDK Technical Guide*
- *EMV 3-D Secure SDK—Device Information*

Effective Date

1 October 2017

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Contents

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

October Publication of EMV 3-D Secure Protocol and Core Functions Specifications, Version 2.1.0

Throughout specification:

Updated all instances of:

- “2.0.1” to “2.1.0”
- “base64” to “base64url”
- “standard TLS” to “TLS”
- “PCI DSS” to “Payment System security requirements”

Chapter 1 Introduction

Table 1.3: Definitions (New and Updated)

- 3DS Requestor Initiated (3RI)
- Ends processing
- Message Version
- Preparation Request (PReq) Message
- Preparation Response (PRes) Message
- Protocol Version
- Base64url

1.7 3-D Secure ~~Message~~ Protocol Version Number

The following table provides the ~~Message~~ Protocol Version Number status for the 3-D Secure Protocol and Core Functions Specification.

Table 1.5 ~~Message~~ Protocol Version Number

1.8 Supporting Documentation

These documents as well as EMV 3-D Secure FAQS are located on the EMVCo website under the 3-D Secure heading.

- *EMV 3-D Secure JSON Message Samples*
- *EMV 3-D Secure Best Practices—User Interface*

Chapter 2 EMV 3-D Secure Overview

2.2.1 Directory Server

- Maintaining ACS and DS Start and End Protocol Versions ~~versions~~ and 3DS Method URLs

2.2.2 Directory Server Certificate Authority

- Signing certificates used to sign messages passed from the ACS to the 3DS SDK ~~and from the ACS to the 3DS Server~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

2.4.7 Preparation Request Message (PReq)

The PReq message is sent from the 3DS Server to the DS to request information about the Protocol Version Number(s) ~~version~~ supported by available ACSs and the DS and if one exists, any corresponding 3DS Method URL.

2.4.8 Preparation Response Message (PRes)

The 3DS Server can utilise the PRes message to cache information about the Protocol Versions supported by available ACSs and the DS, and if one exists, about the corresponding 3DS Method URL.

2.4.9 Error Message

Error messages provide additional information about an error that occurred during message processing between the 3DS Server, the DS, the ACS, and the 3DS SDK.

2.5.3 Processing Payment EMV Payment Tokens

Added section number and title to existing Note.

2.6.2 3DS Requestor Environment—Browser-based

1.1 **3DS Requestor and 3DS Server**—The 3DS Requestor communicates with the 3DS Server. The 3DS Server determines the ACS and DS Start and End Protocol Versions and, if present obtains the 3DS Method URL for the requested card range and returns the information to the 3DS Requestor. The ACS ~~version~~ and DS Protocol Versions and 3DS Method URL data was previously received by the 3DS Server via a PRes message.

Chapter 3 EMV 3-D Secure Authentication Flow Requirements

3.1 App-based Requirements

Step 2 The 3DS Requestor App

The 3DS Requestor App provides to the SDK the:

- ~~the~~ Directory Server ID value (which is the Payment System’s RID), and
- Message version (obtained from the 3DS Requestor Environment)

and obtains the:

- ~~the~~ SDK Transaction ID,
- SDK App ID~~, and the~~
- Device Information, and.
- Protocol Version (Version used by the SDK for this transaction)

Note: If the message version is null, the SDK will utilise the highest version of the protocol that it supports. If the message version is present, the SDK will utilise that version of the protocol (assuming the SDK supports the version), otherwise the SDK will error.

Step 5 The 3DS Server

Note: The 3DS Server can use the ACS Start Protocol Version, ACS End Protocol Version, DS Start Protocol Version and DS End Protocol Version obtained from the PRes message to verify that the ACS and DS support the protocol version used by the 3DS Server.

[Req 18]

Check that the Message Version Number is supported by the DS and the ACS.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

- If not, the DS returns to the 3DS Server EITHER an:
- ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing, OR

[Req 23]

If the connection cannot be established with the ACS then the DS ~~returns to the 3DS Server a message as specified~~ proceeds as specified in [Req 233] and **ends processing**.~~EITHER an:~~

~~ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing.~~

~~Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 405 with detailed information about the issue in the Error Description and ends processing.~~

[Req 24]

If the DS does not receive an ARes message from the ACS (as defined in Section 5.5), the DS returns to the 3DS Server a message as specified in [Req 235] and **ends processing**.

Step 7 ACS

Note: The ACS uses the Device Information received in the AReq message to recognise the device, assess transaction risk, and determine if it can complete the authentication with this device.

[Req 318]

The ACS shall not initiate an interaction with the Cardholder as part of a Frictionless transaction. Cardholder interaction shall be done as part of a Challenge flow.

[Req 32]

If a challenge is deemed necessary (Transaction Status = C), the ACS determines whether an acceptable challenge method is supported by the 3DS SDK based in part on the following data elements received in the AReq message: Device Channel, Device Rendering Options Supported, and SDK Maximum Timeout.

[Req 305]

- If the ACS ~~Server~~ Reference Number does not represent a participating ACS ~~Server~~ the DS shall:
  - Return to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 and **ends processing**, OR
  - Send an ARes message to the 3DS server (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS.

[Req 40]

Evaluate based in part on the 3DS Requestor Challenge Indicator, the ACS Challenge Mandated Indicator and the ACS Rendering Type ~~preference~~whether to perform the requested challenge.

- If the 3DS Requestor continues without performing the requested challenge, receive the RReq message from the DS and Validate as defined in Section 5.9.9. If the message is in error, the 3DS Server **ends processing**. Format the RRes message as defined in Table B.9 and send to the DS. Further processing is outside the scope of 3-D Secure processing.

[Req 64]

If the Cardholder abandons the challenge during the processing of Step 16 and Step 17, or if the ACS receives an abandonment CReq message from the 3DS SDK (as defined in [Req 60]), then the ACS sets the Challenge Completion Indicator = Y in the CRes message and sets the Challenge Cancellation Indicator to the appropriate value in the RReq message.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

3.2 Challenge Flow with OOB Authentication Requirements

Step 17

If the ACS determines the cardholder did not authenticate, it can update the cardholder instructions through another CRes message.

When a Cardholder returns to the 3DS Requestor App from another app, the ACS can also utilise the Challenge Additional Info Text element (for the Native UI) or the ACS HTML Refresh element (for the HTML UI) to improve the UI experience. In this scenario, the SDK will automatically display these data elements when the 3DS Requestor App is moved to the foreground.

Note: The 3DS Requestor should consider that an OOB authentication can take longer for the Consumer to complete and therefore should adjust the SDK's challenge time-out accordingly.

3.3 Browser-based Requirements

Step 2 3DS Server/3DS Requestor

The 3DS Requestor uses the Cardholder Account Number and optionally other cardholder information to request the ACS Start Protocol Version, ACS End Protocol Version, DS Start Protocol Version, and DS End Protocol Version ~~Message Version Number~~ and if present, the 3DS Method URL for that BIN range from the 3DS Server.

[Req 80]

Retrieve the ACS Start Protocol Version and ACS End Protocol Version, DS Start Protocol Version and DS End Protocol Version ~~Message Version Number~~, and, if present, the 3DS Method URL (stored from a previously received PRes message) for that BIN range.

[Req 82]

Pass the 3DS Server Transaction ID, ACS Start Protocol Version and ACS End Protocol Version, DS Start Protocol Version and DS End Protocol Version ~~Message Version Number~~ and if present, the 3DS Method URL back through the 3DS Requestor Environment to the 3DS Requestor.

If the DS Start Protocol Version and DS End Protocol Version are not present for the BIN range, then the default values for the DS Start Protocol Version and DS End Protocol Version located in the PRes message shall be utilised.

[Req 95]

Check that the Message Version Number is supported by the DS and the ACS.

- If not, the DS returns to the 3DS Server EITHER an:
- ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing, OR

[Req 100]

If the connection cannot be established with the ACS then the DS ~~returns to the 3DS Server a message as specified~~ proceeds as specified in [Req 233] and **ends processing**.~~EITHER an:~~

~~ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing.~~

~~Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 405 with detailed information about the issue in the Error Description and ends processing.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

[Req 101]

If the DS does not receive an ARes message from the ACS (as defined in Section 5.5), the DS returns to the 3DS Server a message as specified in [Req 235] and **ends processing**~~EITHER an:~~

~~ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing, OR~~

~~Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 402 with detailed information about the issue in Error Description and ends processing.~~

[Req 319]

The ACS shall not initiate an interaction with the Cardholder as part of a Frictionless transaction. Cardholder interaction shall be done as part of a Challenge flow.

[Req 109]

If a challenge is deemed necessary, (Transaction Status = C) the ACS determines whether an acceptable challenge method is supported by the 3DS Server considering the 3DS Requestor Challenge Indicator ~~based on the Device Channel~~ data element received in the AReq message.

[Req 306]

- If the ACS ~~Server~~ Reference Number does not represent a participating ACS ~~Server~~ the DS shall:
  - Return to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 and **ends processing**, OR
  - Send an ARes message to the 3DS server (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS.

[Req 117]

a. Evaluate based in part on the 3DS Requestor Challenge Indicator, the ACS Challenge Mandated Indicator and the ACS Rendering Type ~~preference~~ whether to perform the requested challenge.

- If the 3DS Requestor continues without performing the requested challenge, receive the RReq message from the DS and Validate as defined in Section 5.9.9. If the message is in error, the 3DS Server **ends processing**. Format the RRes message as defined in Table B.9 and send to the DS.

d. Construct a form containing the CReq message, and ~~optionally~~ if provided by the 3DS Requestor, the 3DS Requestor Session Data (as defined in Table A.3).

New Note following [Req 118]

Note: ACS implementations that use JavaScript or redirection must also support a fall-back for environments that do not support JavaScript.

[Req 140]

Send the final CRes message via an HTTP POST (for example, utilising JavaScript) through the browser to the Notification URL that was sent in the initial ~~CReq~~ AReq message using the secure link established in Step 10.

Step 22

Note: The 3DS Requestor should notify their 3DS Server and DS if invalid CRes messages are being received.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

3.4 3RI-based Requirements

3RI-based implementations shall support only Non-Payment Authentication (NPA) transactions.

[Req 280]

Check that the Message Version Number is supported by the DS and the ACS.

- If not, the DS returns to the 3DS Server EITHER an:
- ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing, OR

[Req 285]

If the connection cannot be established with the ACS then the DS ~~returns to the 3DS Server a message as specified~~ proceeds as specified in [Req 233] and **ends processing**.~~EITHER an:~~

~~ARes message (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS and ends processing.~~

~~Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 405 with detailed information about the issue in the Error Description and ends processing.~~

[Req 286]

- If the DS does not receive an ARes message from the ACS (as defined in Section 5.5), the DS returns to the 3DS Server a message as specified in [Req 235] and **ends processing**.

[Req 291]

- not authenticated because the Issuer is rejecting authentication and requesting that authorisation not be attempted (Transaction Status = R)

[Req 292]

- ~~For a Payment Authentication (Message Category = 01-PA), the ECI value and Authentication Value may be generated and included in the ARes message as defined by the DS.~~

[Req 308]

- If the ACS ~~Server~~ Reference Number does not represent a participating ACS ~~Server~~ the DS shall:
  - Return to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 and **ends processing**, OR
  - Send an ARes message to the 3DS server (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS.

Chapter 4 EMV 3-D Secure UI Templates, Requirements, and Guidelines

4.1 3-D Secure User Interface Templates

Note

Activation During Shopping (ADS) is not a supported UI template within the 3-D Secure specification.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Note:

Cardholder Terms and Conditions is not supported within 3-D Secure UI templates unless regulatory requirements mandate such inclusion during cardholder payment authentication.

[Req 314]

All Device Rendering Options supported shall be supported by the SDK and ACS components.

4.2.1 Processing Screen Requirements

New Figure:

- **Figure 4.5: Sample OOB Template—App-based Processing Flow**
- **Figure 4.12: Sample Challenge Additional Information Text—PA**

*Subsequent Figures are renumbered.

Notes:

If the Cardholder has been authenticated, then the Cardholder is returned to the merchant. If the Cardholder has not authenticated, then the UI is updated to reinforce the expected Cardholder action.

4.2.2 Native UI Templates

Details of the UI rendering process through an SDK are separately described in the *EMV 3-D Secure SDK~~—Implementation~~Technical Guide*.

The use of a carriage return in any UI data element is permitted only as specified in Table A.1.

While the issuer provides the content for the UI, the SDK is responsible for rendering the content. As such, the SDK has the ability to fine tune the UI to best display on the cardholder device. With the SDK's knowledge of the device screen size, font size, etc., the SDK can optimise the content provided by the issuer (for example, by removing an extra line feed that would cause scrolling). Issuers should understand that the formatting provided in the CRes message may not be exactly what is displayed to the cardholder.

Updated Figures:

- **Figure 4.4: Sample OTP/Text Template—App-based Processing Flow**
- **Figure 4.6: Sample Native UI OTP/Text Template—PA**
- **Figure 4.8: Sample Native UI—Single-select Information—PA**
- **Figure 4.9: Sample Native UI—Multi-select Information—PA**
- **Figure 4.10: Sample OOB Native UI Template—PA**
- **Figure 4.11: Sample Challenge Information Text Indicator—PA**

4.2.3.1 3DS SDK

[Req 316]

When Challenge Additional Information Text is present, the SDK would replace the Challenge Information Text and Challenge Information Text Indicator with the Challenge Additional Information Text when the 3DS Requestor App is moved to the foreground.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

4.2.3.2 ACS

[Req 158]

Note: The CReq/CRes message exchange continues until the Challenge Completion Indicator in the CRes is set to Y by the ACS or until the SDK times out.

4.2.4 HTML UI Templates

The SDK will display the HTML exactly as provided by the issuer. As such, it's the issuer's responsibility to format the HTML to best display on the cardholder device. Unlike the Native UI where the SDK can adjust the content provided by the issuer, the HTML provided by the issuer will be exactly what is displayed to the cardholder.

Updated Figure:

- **Figure 4.14: Sample OOB HTML UI Template—PA**

4.2.4.1 HTML Other UI Template ~~ACS Rendering Type~~

The HTML Other UI Template ~~ACS Rendering Type~~ allows Issuers to perform authentication functionality other than the existing Native data element options standard templates.

4.2.5.3 3DS SDK

[Req 317]

When the ACS HTML Refresh element is present, the SDK replaces the ACS HTML with the contents of ACS HTML Refresh when the 3DS Requestor App is moved to the foreground.

Chapter 5 EMV 3-D Secure Message Handling Requirements

5.1.1 HTTP POST

[Req 185]

Messages exchanged between 3-D Secure components shall be in the JSON data interchange format as defined in RFC 7159, or JWE/JWS object format as defined in RFC 7516/RFC 7515.

[Req 186]

The body of the HTTP message shall contain the JSON message properly formatted utilising the JSON required UTF-8-character set as defined in RFC 7159, or JWE/JWS object format as defined in RFC 7516/RFC 7515.

[Req 313]

Cryptographic keys exchanged between 3-D Secure components shall be in the JWK object format as defined in RFC 7517.

5.1.3 Base 64/Base64url Encoding

[Req 192]

The methods of encoding shall follow of Base64url encoding shall follow~~ Section 6.8 of the IETF RFC 2045~~:

- IETF RFC 2045 for Base64
- IETF RFC 7515 for Base64url

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.1.4 Protocol and Message Version Numbers

[Req 194]

A 3-D Secure ~~Message~~ version numbers shall be in the format major.minor~~[patch]~~.patch for messaging (for example, 2.1.0).

[Req 320]

Message Version Numbers shall be validated to ensure that they are consistent across a 3-D Secure transaction. The 3-D Secure component that identifies a validation error shall return an Error Message with the applicable Error Component and Error Code = 203.

For example, when the DS receives an RReq message, the DS will validate that the Message Version Number matches the AReq message.

Note: For all 3-D Secure transactions, the 3DS Server sets the Message Version Number that all components will utilise.

[Req 311]

3-D Secure messages containing an active Message Version Number supported by the 3-D Secure component are processed according to the requirements of the specified protocol version (See Table 1.5).

[Req 320]

Message Version Numbers shall be validated to ensure that they are consistent across a 3-D Secure transaction (for example, when the DS receives an RReq message, the DS will validate that the Message Version Number matches the AReq message). The 3-D Secure component that identifies a validation error shall return an Error Message with the applicable Error Component and Error Code = 203.

5.1.5 Message Parsing

[Req 198]

3-D Secure message is properly formatted as a JSON message as defined in RFC 7159 or JWE/JWS object format as defined in RFC 7516/7515.

[Req 209]

If there are additional data elements received that are not specified for the Message Type, ~~Message Version Number,~~ Device Channel and Message Category but the message otherwise passes validation, the message shall be considered valid.

Alternatively, if the additional data elements in the AReq message do not pass validation criteria, the DS responds with an error message to the 3DS Server.

Note: The AReq message will have additional message content validation requirements based on the content of the data elements.~~: Device Channel and Message Category~~

5.4 Error Codes

[Req 218]

For Error Codes 201, 203, 204 and 304, the 3-D Secure component that identifies the error shall include the name(s) of the erroneous data element(s) in the Error Detail.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.5.1 Transaction Timeouts

[Req 221]

If the transaction reaches the 30-second timeout expiry, send an RReq message to the DS to be passed to the 3DS Server with Transaction Status = N, Transaction Status Reason = 14 (Challenge Transaction Timed Out), and Challenge Cancelation Indicator = 05 (Transaction timed out at the ACS—First CReq not received). Clear the ephemeral key generated and stored for use in the CReq/CRes message exchange for the current transaction.

[Req 227]

The ACS sends a CRes message with a Transaction Status = N to the Notification URL received in the initial ~~CReq~~ AReq message.

5.5.2.1 AReq/Ares Message Timeouts

[Req 233]

Send an ARes message to the 3DS Server (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS.

[Req 235]

Send an ARes message to the 3DS Server (as defined in Table B.2) with Transaction Status set to the appropriate response as defined by the specific DS.

[Req 236]

Any failure to complete the initial connection and TLS handshake to the ACS ~~Server~~ shall result in an immediate retry. Upon second failure, the 3DS SDK shall send an error to the 3DS Requestor App to complete the transaction.

5.5.2.2 CReq/CRes Message Timeouts

[Req 239]

If the ACS does not respond with the CRes message before the 3DS SDK 10-second read timeout expiry, the 3DS SDK ends 3-D Secure processing, passes an Error Message to the ACS with Error Component = C and Error Code = 402 and passes an error message to the calling app, ending the 3-D Secure transaction.

[Req 312]

The 3DS SDK shall set the SDK Maximum Timeout value from the time the TLS handshake has completed and the first CReq message is sent for processing to the ACS.

If the ACS does not respond with the final CRes message before the SDK Maximum Timeout expiry, the 3DS SDK ends 3-D Secure processing, passes an Error Message to the ACS with Error Component = C and Error Code = 402 and passes an error message to the calling app, ending the 3-D Secure transaction.

The SDK Maximum Timeout shall not have a value less than five minutes.

5.5.2.3 RReq/RRes Message Timeouts

[Req 242]

If the DS has not responded with the RRes message or an Error message before the 5-second read timeout expiry, the ACS shall return to the DS an Error Message (as defined in A.5.5) with Error Component = A and Error Code = 402 ~~close and re-establish a new connection and resend with Transaction Status = U~~.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.6 PReq/PRes Message Handling Requirements

The PReq/PRes messages are utilised by the 3DS Server to cache information about the ~~version~~ Protocol Version Numbers(s) supported by available ACSs, the DS and also any URL to be used for the 3DS Method call. The data will be organised by card range as configured by a DS. The information provided on the ~~version~~ Protocol Version Number(s) supported by ACSs and DSs can be utilised in ~~both~~ the App-based,~~ and ~~Browser-based, and 3RI flows.

5.7 App/SDK-based Message Handling

~~[Req 352]~~ Text is unchanged, but is no longer marked as a requirement.

5.8.1 3DS Method Handling

[Req 258]

If the 3DS Method URL does not exist, the 3DS Requestor will notify the 3DS Server to set the 3DS Method Completion Indicator = U.

[Req 261]

Render a hidden HTML iframe in the Cardholder browser and send a form with a field named `threeDSMethodData` containing the JSON Object via HTTP POST to the ACS 3DS Method URL obtained from the PRes message cache data.

[Req 263]

Complete the required action. Recall the 3DS Server Transaction ID received in the initial 3DS Method POST and send via a form with a field named `threeDSMethodData` in the Cardholder browser HTML iframe using an HTTP POST method to the 3DS Method Notification URL.

[Req 315]

If the 3DS Method completes within 10 seconds, then the 3DS Requestor will notify the 3DS Server to set the 3DS Method Completion Indicator = Y. If the 3DS Method does not complete in 10 seconds, set the 3DS Method Completion Indicator to = N.

5.9 Message Error Handling

The 3-D Secure component will receive and validate the message, ~~and in some instances,~~ will verify and decrypt the message before performing further processing.

5.9.5.1 Message in Error

- Sends to the DS an RReq message (as defined in Table B.8) with Transaction Status = U and Challenge Cancelation Indicator = ~~05~~06 using the secure link.

5.9.8.1 Message in Error

- Sends to the 3DS Server an RReq message (as defined in Table B.8) with Transaction Status ~~= U and Challenge Cancelation Indicator = 05 using the secure link~~set to the appropriate value as defined by the specific DS.
- Sends to the 3DS Server an ~~RReq~~ Error Message (as defined in Table B.10) with Error Component = D and the Error Code returned to the ACS. ~~Transaction Status = U and Challenge Cancelation Indicator = 5 using the secure link~~.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.9.11 ACS RRes Message Error Handling—01-APP

- If a specific transaction can be identified, the ACS:
  - Sends to the 3DS SDK an Error Message (as defined in Section A.5.5) with Error Component = A and Error Code = 101 using the secure link established in ***[Req 56]***.

Chapter 6 EMV 3-D Secure Security Requirements

6.1.4 Link d: 3DS ~~SDK~~Client—ACS

6.2.2.1 3DS SDK Encryption

- "alg": RSA-OAEP-256

6.2.3.1 3DS SDK Preparation for Secure Channel

- Generates a fresh ephemeral key pair (Q_C, d_C) as described in Annex C and provides Q_C as a JWK for inclusion in the AReq message as sdkEphemPubKey.

6.2.3.2 ACS Secure Channel Setup

- "epk": Q_C (received in the AReq message as sdkEphemPubKey)
- {"~~ACSPublicKey~~acsEphemPubKey": "Q_T", "sdkEphemPubKey~~threeDSSDKPublicKey~~":" Q_C", "ACSURL":"ACS URL" }

6.2.3.3 3DS SDK Secure Channel Setup

- "epk": Q_T (received in the ARes message as acsEphemPubKey which is part of ACS Signed Content)

Annex A 3-D Secure Data Elements

A.1 Missing Required Fields

- field name is present but the value is empty or null

A.4 EMV 3-D Secure Data Elements

Table A.1 EMV 3-D Secure Data Elements

Throughout Table:

Added the following to all values "Reserved for EMVCo future use":

Values invalid until defined by EMVCo.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
3DS Method Completion Indicator Field Name: `threeDSCompInd`	Indicates whether the 3DS Method successfully completed.	3DS Server	Length: 1 character JSON Data Type: String Values accepted: Y = Successfully completed N = Did not successfully complete U = Unavailable—3DS Method URL was not present in the PRes message data for the card range associated with the Cardholder Account Number.	02-BRW	01-PA 02-NPA	AReq = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
3DS Requestor Authentication Indicator Field Name: threeDSRequestorAuthenticationInd	Indicates the type of Authentication request. This data element provides additional information to the ACS to determine the best approach for handing an authentication request.	3DS Server	Length: 2 characters JSON Data Type: String Values accepted: 01 = Payment transaction 02 = Recurring transaction 03 = Instalment transaction 04 = Add card 05 = Maintain card 06 = Cardholder verification as part of EMV token ID&V 07–79 = Reserved for EMVCo future use (values invalid until defined by EMVCo) 80-99 = Reserved for DS use	01-APP 02-BRW	01-PA 02-NPA	AReq = R	
3DS Requestor Challenge Indicator			Note: If the element is not provided, the expected action is that the ACS would interpret as 01 = No preference.				
~~3DS Requestor Initiated Indicator~~ *entire row deleted							

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
~~3DS Requestor Non-Payment Authentication Indicator~~ *entire row deleted							
3DS Requestor Prior Transaction Authentication Information			JSON Data Type: Object ~~with multiple arrays~~				
3DS Requestor URL						~~02-NPA~~ ~~AReq = R~~	
3DS Server Operator ID						AReq = ~~O~~C PReq = ~~O~~C	Requirements for the presence of this field are DS specific.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
3RI Indicator Field Name: `threeRIInd`	Indicates the type of 3RI request. This data element provides additional information to the ACS to determine the best approach for handing a 3RI request.	3DS Server	Length: 2 characters JSON Data Type: String Values accepted: 01 = Recurring transaction 02 = Instalment transaction 03 = Add card 04 = Maintain card information 05 = Account verification 06–79 = Reserved for EMVCo future use (values invalid until defined by EMVCo) 80-99 = Reserved for DS use	03-3RI	02-NPA	AReq = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	**Description**	**Source**	**Length/Format/ Values**	**Device Channel**	**Message Category**	**Message Inclusion**	**Condition Inclusion**
ACS Ephemeral Public Key (Q_T)			Length: Variable, maximum ~~128~~256 characters JSON Data Type: ~~String~~Object ~~Value accepted: Binary data base64 encoded~~ ~~Generate value in accordance with FIPS 186-4~~				
ACS HTML	HTML provided by the ACS in the CRes message. Utilised ~~in~~when HTML is specified in the ACS UI Type during the Cardholder challenge.						Conditional ~~if ACS~~ upon selection of the ~~HTML~~ACS UI Type = 5 (HTML) by the ACS.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS HTML Refresh Field Name: acsHTMLRefresh	Optional HTML provided by the ACS in the CRes message to be utilised in the Out of Band flow when the HTML is specified in the ACS UI Type during the Cardholder challenge. If the ACS HTML Refresh is present in the CRes message, the SDK will display it when the app is moved to the foreground.	ACS	Length: Variable, maximum 100KB JSON Data Type: String Value accepted: Base64url encoded HTML This value will be Base64url encoded prior to being placed into the CRes message.	01-APP	01-PA 02-NPA	CRes = O	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS Operator ID						AReq = ~~O~~C	Requirements for the presence of this field are DS specific.
ACS Rendering Type	Identifies the ACS UI ~~type~~Template that the ACS will first present to the consumer~~will be utilised by the ACS to complete the Cardholder challenge. Provides additional information about the challenge to the 3DS Requestor~~.		JSON Data Type: Object ~~with multiple arrays~~				
ACS Signed Content			JSON Data Type: Object ~~with multiple arrays~~				
Address Match Indicator						AReq = ~~C~~O	~~Required in 01-PA if 3DS Requestor has Shipping Address information.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Authentication Value						~~02-NPA:~~ ~~ARes = C~~ ~~RReq = C~~ ~~03-3RI:~~ ~~ARes = C~~	~~03-3RI:~~ ~~Conditional based on DS rules.~~
Broadcast Information Field Name: `broadInfo`	Unstructured information sent between the 3DS Server, the DS and the ACS.	3DS Server DS ACS	Length: 4096 characters JSON Data Type: Object	01-APP 02-BRW 03-3RI	01-PA 02-NPA	AReq = C ARes = C	Requirements for the presence of this field are DS specific.
Card Range Data	Card range data from the DS indicating the most recent protocol versions~~Message Version Number~~ supported by the ACS, and optionally the DS that hosts that range, and if configured, the ACS URL for the 3DS Method.		JSON Data Type: ~~Object with multiple~~ Array ~~Note: Data will be formatted into a JSON object prior to being placed into the Card Range Data field of the message.~~				
Card/Token Expiry Date						AReq = ~~R~~C	The requirements for the presence of this field are DS specific.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Account Information			JSON Data Type: Object ~~with multiple arrays~~				
Cardholder Account Number	Account number that will be used in the authorisation request for payment transactions.		Format represented ISO ~~7813~~ 7812				
Cardholder Billing Address City							01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address Country							Required if Cardholder Billing Address State is present. 01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address Line 1							01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address Line 2							01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address Line 3							01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address Postal Code							01-PA: Required unless market or regional mandate restricts sending this information. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Billing Address State							01-PA: Required unless market or regional mandate restricts sending this information, or State is not applicable for this country. 02-NPA: Required (if available) unless market or regional mandate restricts sending this information, or State is not applicable for this country.
Cardholder Home Phone Number							Required (if available), unless market or regional mandate restricts sending this information

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Information Text Field Name: `cardholderInfo`	Text provided by the ACS/Issuer to Cardholder during a Frictionless transaction that was not authenticated by the ACS. The Issuer can optionally provide information to Cardholder. For example, “Additional authentication is needed for this transaction, please contact (Issuer Name) at xxx-xxx-xxxx.”	ACS	Length: Variable, maximum 128 characters JSON Data Type: String If field is populated this information shall optionally be displayed to the cardholder by the merchant.	01-APP 02-BRW	01-PA 02-NPA	ARes = O	
Cardholder Mobile Phone Number							Required (if available) unless market or regional mandate restricts sending this information

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Shipping Address City							Required (if available) unless market or regional mandate restricts sending this information

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Shipping Address Country							Required if Cardholder Shipping Address State is present. Required (if available) unless market or regional mandate restricts sending this information.
Cardholder Shipping Address Line 1							Required (if available) unless market or regional mandate restricts sending this information.
Cardholder Shipping Address Line 2							Required (if available) unless market or regional mandate restricts sending this information.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	**Description**	**Source**	**Length/Format/ Values**	**Device Channel**	**Message Category**	**Message Inclusion**	**Condition Inclusion**
Cardholder Shipping Address Line 3							Required (if available) unless market or regional mandate restricts sending this information.
Cardholder Shipping Address Postal Code							Required (if available) unless market or regional mandate restricts sending this information.
Cardholder Shipping Address State							Required (if available) unless market or regional mandate restricts sending this information, or State is not applicable for this country.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Challenge Additional Information Text	~~Additional t~~Text provided by the ACS/Issuer to Cardholder during OOB authentication to replace ~~the Challenge Message exchange that could not be accommodated in the~~ Challenge Information Text and Challenge Information Text Indicator in the OOB Template.		If field is populated this information is displayed to the Cardholder by the SDK when the 3DS Requestor App is brought to the foreground.				
Challenge Cancelation Indicator			04 = Transaction Timed Out at ACS—other timeouts 05 = Transaction Timed Out at ACS—First CReq not received by ACS ~~05~~06 = Transaction Error ~~06~~07 = Unknown ~~07~~08-79 = Reserved for future EMVCo use (values invalid until defined by EMVCo)				Required in CReq for 01-APP if the authentication transaction was cancelled by user interaction with the cancelation button in the UI or for other reasons as indicated.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	**Description**	**Source**	**Length/Format/ Values**	**Device Channel**	**Message Category**	**Message Inclusion**	**Condition Inclusion**
Challenge Data Entry							Required when ACS UI Type = 01, 02, or 03 ~~04~~ and challenge data has been entered into the UI.
Challenge Information Text			Length: ~~256 ~~350 characters Note: Carriage return is supported in this data element and is represented by an “\n”.				
Challenge Selection Information			JSON Data Type: ~~Object with multiple~~ Array ~~Note: Data will be formatted into a JSON object prior to being placed into the Challenge Selection Information field of the message.~~				
Device Information			Length: Variable, maximum ~~15360~~64000 characters JSON Data Type: Object ~~with multiple arrays~~				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Device Rendering Options Supported	Defines the SDK UI types that the device supports for displaying specific challenge user interfaces within the SDK. Note: As established in *[Req 314]*, ~~ACS~~ All Device Rendering Options must be supported by ~~all~~ the SDK and ACS components.		JSON Data Type: Object ~~with multiple arrays~~				
DS End Protocol Version Field Name: dsEndProtocolVersion	The most recent active protocol version that is supported for the DS. Note: Optional within the Card Range Data (as defined in Table A.6).	DS	Length: Variable, 5–8 characters JSON Data Type: String	N/A	01-PA 02-NPA	PRes = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
DS Start Protocol Version Field Name: dsStartProtocolVersion	The most recent active protocol version that is supported for the DS. Note: Optional within the Card Range Data (as defined in Table A.6).	DS	Length: Variable, 5–8 characters JSON Data Type: String	N/A	01-PA 02-NPA	PRes = R	
DS URL						AReq = ~~RC~~	Required between the DS and ACS but will not be present from 3DS Server to DS
Electronic Commerce Indicator (ECI)						02-NPA	
Expandable Information Label~~ 1~~ Fields Name: expandInfoLabel~~1~~						CRes = ~~C~~O	~~Required based upon the ACS UI Type selected.~~
Expandable Information Text~~ 1~~ Fields Name: expandInfoText~~1~~			Note: Carriage return is supported in this data element and is represented by an “\n”.			CRes = ~~C~~O	~~Required based upon the ACS UI Type selected.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	**Description**	**Source**	**Length/Format/ Values**	**Device Channel**	**Message Category**	**Message Inclusion**	**Condition Inclusion**
Instalment Payment Data				~~03-3RI~~			Required if the Merchant and Cardholder have agreed to instalment payments, i.e. if 3DS Requestor Authentication Indicator = 03.
Issuer Image			Format: JSON Object~~ with multiple arrays~~				~~Required on the initial CRes message from the ACS, omitted after.~~Presence of this field is Payment System specific. ~~Conditional for Native UI.~~
Merchant Category Code						02-NPA: AReq = ~~C~~O	Optional but strongly recommended to include ~~Required~~ for 02-NPA ~~and 03-3RI~~ if the merchant is also the 3DS Requestor.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Merchant Country Code			Note: The same value must be used in the authorisation request.			02-NPA: AReq = ~~C~~O ~~03-3RI:~~ ~~AReq = C~~	Optional but strongly recommended to include ~~Required~~ for 02-NPA ~~and~~ ~~03-3RI~~ if the merchant is also the 3DS Requestor.
Merchant Name						02-NPA: AReq = ~~C~~O ~~03-3RI:~~ ~~AReq = C~~	Optional but strongly recommended to include ~~Required~~ for 02-NPA ~~and~~ ~~03-3RI~~ if the merchant is also the 3DS Requestor.
Merchant Risk Indicator			JSON Data Type: Object ~~with multiple arrays~~				
Message Category						~~02-NPA:~~ ~~AReq = R~~ ~~RReq = R~~ ~~03-3RI~~ ~~AReq = O~~	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Message Extension			Length: maximum 819~~2~~0 bytes				
Message Version Number	~~Specification~~Protocol version identifier. This shall be the Protocol ~~version~~ Version ~~number~~ Number of the specification utilised by the system creating this message. The Message Version Number is set by the 3DS Server which originates the protocol with the AReq message. The Message Version Number does not change during a 3DS transaction.	~~DS~~ ~~ACS~~ ~~3DS SDK~~	Length: Variable, 5–8 characters Value accepted: See Table 1.5 ~~n.n.n where:~~ ~~“n” represents a numeric digit that relates to the major and minor of the specification version number.~~				
Notification URL	Fully qualified URL of the system that receives the CRes message or Error Message.				~~CReq~~ AReq = R		

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Payment System Image	Sent in the initial CRes message from the ACS to the 3DS SDK to provide the URL(s) of the DS or Payment System logo or image to be used in the Native UI. ~~Option 2: May also be "none" if no image is to be displayed.~~		JSON Data Type: Object ~~with multiple arrays~~				~~Required on the initial CRes message from the ACS, omitted after.~~Presence of this field is Payment System specific. ~~Conditional for ACS UI Type = 01–04~~
Purchase Amount							Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.
Purchase Currency			Numeric				Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Purchase Currency Exponent							Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.
Purchase Date & Time	Date and time of the purchase, expressed in ~~GMT~~ UTC ~~For instalment and reoccurring purchases, this will be the time of the original cardholder purchase agreement.~~						Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.
Recurring Expiry				~~03-3RI~~			Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Recurring Frequency				~~03-3RI~~			Required for 02-NPA if ~~Instalment, or Recurring transactions~~3DS Requestor Authentication Indicator = 02 or 03.
SDK Encrypted Data			Maximum ~~15360~~ 64000 JSON Data Type: Object ~~with multiple arrays~~				
SDK Ephemeral Public Key (Q_C)			Length: maximum ~~128~~256 characters JSON Data Type: ~~String~~Object ~~Base64url encoded binary key~~JWK				
SDK Maximum Timeout Field Name: `sdkMaxTimeout`	Indicates maximum amount of time (in minutes) for all exchanges.	3DS SDK	Length: 2 characters JSON Data Type: String Values accepted: Greater than or = 05	01-APP	01-PA 02-NPA	AReq = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Transaction Status	Indicates whether a transaction qualifies as an authenticated transaction or account verification.		Y = Authentication/Account Verification Successful; ~~All data needed for authorisation, including the Authentication Value, is included in the message for 01-PA~~ N = Not Authenticated/Account Not Verified; Transaction denied U = Authentication/Account Verification Could Not Be Performed; Technical or other problem, as indicated in ARes or RReq A = Attempts Processing Performed; Not authenticated/verified, but a proof of attempted authentication/verification is provided. ~~All data needed for authorisation including the Authentication Value is included in the message for 01-PA.~~ R = Authentication/Account Verification Rejected; Issuer is rejecting authentication/verification and request that authorisation not be attempted.			02-NPA: ARes = C RReq = C CRes = C	~~For 02-NPA optional as defined by DS~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Transaction Status Reason							~~For 03-3RI. Optional as defined by the DS~~
Why Information Label Field Name: whyInfoLabel~~1~~						CRes = ~~C~~O	~~Required based upon the ACS UI format selected~~
Why Information text Field Name: whyInfoText~~1~~			Note: Carriage return is supported in this data element and is represented by an “\n”.			CRes = ~~C~~O	~~Required based upon the ACS UI format selected~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.5.3 3DS Method Data

The HTTP field name is `threeDSMethodData`.

A.5.5 Error Code, Error Description, and Error Detail

Table A.4 Error Code, Error Description, and Error Detail

Value	Code	Description	Detail
103	Sent messages limit exceeded	Exceeded maximum number of PReq messages sent to the DS.	For example, the 3DS Server sends two PReq messages to the DS within one hour

201			Parent Example:
			`messageType`
			Parent/Child Example:
			`acctInfo.chAccAgeInd`

203		Data element not in the required format or value is invalid as defined in Table A.1, ,or Message Version Number does not match the value set by the 3DS Server in the AReq message.	

A.5.4 Browser CReq and CRes POST

Table A.3 3DS CReq/CRes POST Data

Data Element/Field Name	Description	Length/Format/Value	Inclusion
3DS Requestor Session Data	3DS Requestor session data that is returned by the ACS in the CRes message POST to the 3DS Requestor.		

A.5.7 Card Range Data

It also may optionally contain the ACS URL for the 3DS Method if supported by the ACS and the DS Start and End Protocol Versions which support the card range.

Table A.6 Card Range Data

Data Element/Field Name	Description	Length/Format/Value	Inclusion
Action Indicator	Note: If the Serial Number is not included in the PReq message, then the action is A = Add for all card ranges returned (the Action Indicator is ignored in the PRes message).		

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

ACS End Protocol Version Field Name: `acsEndProtocolVersion`	The most recent active protocol version that is supported for the ACS URL.	Field Length: Variable 5–8 characters ~~n.n.n where:~~ ~~"n" represents a numeric digit that relates to the major and minor of the specification version number.~~	
ACS Start Protocol Version Field Name: `acsStartProtocolVersion`	The earliest (i.e. oldest) active protocol version that is supported by the ACS.	Field Length: Variable 5–8 characters ~~n.n.n where:~~ ~~"n" represents a numeric digit that relates to the major and minor of the specification version number.~~	
DS End Protocol Version Field Name: dsEndProtocolVersion	The most recent active protocol version that is supported for the DS.	Length: Variable, 5–8 characters JSON Data Type: String	O
DS Start Protocol Version Field Name: dsStartProtocolVersion	The earliest (i.e. oldest) active protocol version that is supported by the DS.	Length: Variable, 5–8 characters JSON Data Type: String	O

A.6 Message Extension Data

A maximum of 10 extensions (objects) are supported within the Message Extension data element, totalling a maximum of 81920 bytes.

A.6.1 Message Extension Attributes

Table A.7 Message Extension Attributes

Attribute Name	Description	Length/Format/Value	Inclusion
data		JSON Data Type: ~~Sting~~Object	
id	Note: Payment System Registered Application Provider Identifier (RID) is required as prefix of the ID. ID If the extension is EMVCo-assigned, the prefix of the ID will be "EMVCO".		

JSON example replaced

A.7.1 Cardholder Account Information

The Cardholder Account Information contains optional information about the Cardholder Account. The detailed data elements, which are optional are outlined in Table A.8.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.2 Merchant Risk Indicator

Table A.9 Merchant Risk Indictor

Data Element/Field Name	Description	Length/Format/Value
Shipping Indicator		01 ~~(addrMatch = Y)~~ 02 ~~(addrMatch = N)~~ 03 ~~(addrMatch = N)~~

A.7.3 3DS Requestor Authentication Information

The 3DS Requestor Authentication Information contains optional information about how the cardholder authenticated during login to their 3DS Requestor account.

Table A.10 3DS Requestor Authentication Information

Data Element/Field Name	Description	Length/Format/Value
3DS Requestor Authentication Method		04 = Login to the cardholder account at the 3DS Requestor system using issuer credentials 05 = Login to the cardholder account at the 3DS Requestor system using third-party authentication ~~04~~06 = Login to the cardholder account at the 3DS Requestor system using FIDO Authenticator ~~05~~07–79 = Reserved for EMVCo future use

A.7.4 3DS Requestor Prior Transaction Authentication Information

The 3DS Requestor Prior Transaction Authentication Information contains optional information about a 3DS cardholder authentication that occurred prior to the current transaction.

Table A.1: 3DS Requestor Prior Transaction Authentication Information

Data Element/Field Name	Description	Length/Format/Value
3DS Requestor Prior Transaction Authentication Method		03 = AVS verified 04 = Other issuer methods ~~03~~05–79 = Reserved for EMVCo future use

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.5 ACS Rendering Type

Table A.12 ACS Rendering Type

Data Element/Field Name	Description	Length/Format/Value
ACS Interface Field Name: acsInterface	This the ACS interface that the challenge will present to the cardholder.	
ACS UI Template~~ Type~~ Field Name: ~~uiType~~acsUITemplate	~~This is the UI Type that the ACS will be presenting.~~ Valid values for each interface: Native UI = 01–04 HTML UI = 01–05	

JSON Example—Updated to reflect Table A.12 updates

A.7.6 Device Rendering Options Supported

Table A.13 Device Rendering Options Supported

Data Element/Field Name	Description	Length/Format/Value
SDK Interface Field Name: sdkInterface	Lists all of the SDK Interface types that the device supports for displaying specific challenge user interfaces within the SDK.	
SDK UI Type Field Name: sdkUi~~ui~~Type	~~This is the UI Type that the ACS will be presenting.~~ Valid values for each interface: Native UI = 01–04 HTML UI = 01–05 Note: Currently, all SDKs need to support all UI Types. In the future, however, this may change (for example, smart watches may support a UI Type not yet defined by this specification).	JSON Data Type: Array of String

JSON Example—Updated to reflect Table A.13 updates

A.7.7 Issuer Image

Table A.14 Issuer Image

Data Element/Field Name	Description	Length/Format/Value

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

~~No Issuer Image~~
~~Field Name: none~~
	~~Option 1—~~Images to display:
	“issuerImage” :{
	“medium”: “http://acs.com/medium_image.svg”,
	”high”: “http://acs.com/high_image.svg”,
	“extraHigh”: “http://acs.com/extraHigh_image.svg”_
	}
		~~Option 2—No Image to display:~~
		~~The value of “none” should be passed~~

A.7.8 Payment System Image

Table A.15 Payment System Image

Data Element/Field Name	Description	Length/Format/Value

Medium Density Image
~~No Payment System Image~~
~~Field Name: none~~
	~~Option 1—~~Images to display:
	“psImage” :{
	“medium”: “http://ds.com/medium_image.svg”,
	”high”: “http://ds.com/high_image.svg”,
	“extraHigh”: “http://ds.com/extraHigh_image.svg”_
	}
		~~Option 2—No Image to display:~~
		~~The value of “none” should be passed~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Annex B Message Format

B.1 AReq Message Data Elements

Table B.1 AReq Data Elements

Data Element	Field Name
3DS Method Completion Indicator	threeDSCompInd
3DS Requestor Authentication Indicator	threeDSRequestorAuthenticationInd
~~3DS Requestor Initiated Indicator~~	~~threeDSRequestor3RIInd~~
~~3DS Requestor Non-Payment Authentication Indicator~~	~~threeDSRequestorNPAInd~~
3RI Indicator	threeRIInd
Broadcast Information	broadInfo
Notification URL	notificationURL
SDK Maximum Timeout	sdkMaxTimeout

B.2 ARes Message Data Elements

Table B.2 ARes Data Elements

Data Element	Field Name
Broadcast Information	broadInfo
Cardholder Information text	cardholderInfo

B.3 CReq Message Data Elements

Table B.3 CReq Data Elements

Data Element	Field Name
~~Notification URL~~	~~notificationURL~~

B.4 CRes Message Data Elements

Table B.4 CRes Data Elements

Data Element	Field Name
Expand Information Label ~~1~~	expandInfoLabel~~1~~
Expand Information Text ~~1~~	expandInfoText~~1~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element	Field Name
Why Information Label	whyInfoLabel~~1~~
Why Information Text	whyInfoText~~1~~

B.5 Final CRes Message Data Elements

Data Element	Field Name
ACS Counter ACS to SDK	acsCounterAtoS

B.7 PRes Message Data Elements

Table B.7 PRes Data Elements

Data Element	Field Name
DS End Protocol Version	dsEndProtocolVersion
DS Start Protocol Version	dsStartProtocolVersion

Annex D Approved Transport Layer Security Versions

For establishing the links secured by TLS between the DS, 3DS Server, ACS and SDK, the version number shall be V1.2 or higher.

D.1 ~~TLS~~ Cipher Suites for TLS 1.2

D.1.1 Supported ~~TLS~~ Cipher Suites

- TLS_~~ECHDE~~ECDHE_RSA_WITH_AES_128_GCM_SHA256

3DS Server, ACS and SDK shall be able to support both cipher suites.

DS shall be able to support at least one of the cipher suites.

DS CA shall provide client and server certificates for the cipher suite(s) supported by the DS.

D.1.2 Other Cipher Suites

If required for any other reason or for legacy purposes, additional cipher suites may be supported. Interoperability testing is at the implementer's discretion.

~~D.2.1 Implementation Notes~~

~~The use of 3DES and SHA (SHA-1) are to be phased out. They may be deprecated in future versions of this specification.~~

D.2 Not Supported

- Any cipher suite incorporating any of the algorithms 'RC2', 'RC4', 'DES', 'IDEA', KRB5', ~~'SEED',~~ 'ARIA', ~~'CHACHA20'~~ or 'MD5'

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

June Publication of EMV 3-D Secure Protocol and Core Functions Specifications, Version 2.0.1.Draft.June

Chapter 1 Introduction

1.5 Definitions

Table 1.3—Definitions

- Base64
- Directory Server ID
- Fully Qualified URL

1.7 3-D Secure Version Number

Table 1.5 Version Number

Chapter 3 EMV 3-D Secure Authentication Flow Requirements

3.1 App-based Requirements

Step 2: The 3DS Requestor App

Note: As described in the EMV 3DS SDK Specification, the 3DS SDK encrypts the Device Information by using the DS public key. This key is identified based on the DirectoryServerID that is passed to the createTransaction method. ~~DirectoryServerID is defined in the EMV 3-D Secure SDK Specification.~~

Step 6: The DS

[Req 17]

Decrypt the SDK Encrypted Data data element of the AReq message as defined in Section 6.2.2.2 and base 64 encode resulting content and move base64 encoded content to ~~move the SDK Encrypted Data data content to~~ the Device Information data element of the AReq message for the ACS.

[Req 19]

Check the data elements in the AReq message as follows.

- If either the:
  - 3DS Server Reference Number does not represent a participating 3DS Server, OR
  - SDK Reference Number does not represent a participating 3DS SDK, OR
  - Acquirer BIN does not represent a participating Acquirer, OR
  - Acquirer Merchant ID is not related to the Acquirer BIN

Then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code 303 and ends processing.

Step 7: The ACS

[Req 26]

Check whether the Consumer Device is supported.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

If not, the ACS returns to the DS an ARes message with a Transaction Status = U and Transaction Status Reason Code = 03 ~~and Error Message (as defined in Section A.5.5) with Error Component = A and Error Code = 401~~ and **ends processing**.

Step 8: The DS

[Req 305]

Check the data elements in the ARes message as follows.

If the ACS Server Reference Number does not represent a participating ACS Server then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 and **ends processing**.

Step 9: The 3DS Server

[Req 39]

For an authenticated transaction (Transaction Status = Y or A):

a. For Payment Authentication (Message Category = 01-PA), ensure that the Transaction Status, ECI value, and Authentication Value as generated by the ACS are provided for the authorisation process.

Step 17: The ACS

[Req 310]

If Challenge Cancelation Indicator has a value, then continue with Step 18.

[Req 61]

Check the authentication data entered by the Cardholder:

- If correct, then the ACS:
  - Increments the Interaction Counter

Step 18: The ACS

[Req 64]

If the Cardholder abandons the challenge during the processing of Step 16 and Step 17, or if the ACS receives an abandonment CReq message from the 3DS SDK (as defined in ***[Req 60]***), then the ACS sets the Challenge Completion Indicator = ~~N~~Y in the CRes message and sets the Challenge Indicator to the appropriate value in the RReq message. Refer to Annex A for the specific values.

3.3 Browser-based Requirements

Step 2: 3DS Server/3DS Requestor

The 3DS Requestor uses the Cardholder Account Number and optionally other cardholder information to request the ACS Message Version Number and if present, the 3DS Method URL for that BIN range from the 3DS Server.

The 3DS Server shall:

[Req 80]

Retrieve the ACS Message Version Number, and, if present, the 3DS Method URL (stored from a previously received PRes message) for that BIN range.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

[Req 82]

Pass the 3DS Server Transaction ID, ACS Message Version Number and if present, the 3DS Method URL back through the 3DS Requestor Environment to the 3DS Requestor.

Step 7: The ACS

[Req 96]

Further check the data elements in the AReq message as follows:

- If the 3DS Server Reference Number does not represent a participating 3DS Server, ~~OR,~~ then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 then ends processing.
    - ~~Acquirer BIN does not represent a participating Acquirer, OR~~
    - ~~Acquirer Merchant ID is not related to the Acquirer BIN~~

~~Then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 then ends processing.~~

Step 8: The ACS

[Req 103]

Check whether the Consumer Device on which the authentication is being requested is supported.

If not, the ACS returns to the DS an ARes message with a Transaction Status = U and Transaction Status Reason Code = 03~~Error Message (as defined in Section A.5.5) with Error Component = A and Error Code = 401~~ and **ends processing**.

Step 9: The DS

The DS shall:

[Req 306]

Check the data elements in the ARes message as follows.

If the ACS Server Reference Number does not represent a participating ACS Server then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code 303 and **ends processing**.

Step 15: The ACS

The ACS shall:

[Req 307]

Embed all resources in the ACS-provided HTML and not fetched via external URLs.

[Req 123]

Check the authentication data entered by the Cardholder:

- If correct, then the ACS:
    - Increments the Interaction Counter

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

3.4 3RI-based Requirements

Step 3: The DS

[Req 281]

Further check the data elements in the AReq message as follows:

- If the 3DS Server Reference Number does not represent a participating 3DS Server, then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 then **ends processing**.
    - ~~o 3DS Server Reference Number does not represent a participating 3DS Server, OR~~
    - ~~o Acquirer BIN does not represent a participating Acquirer, OR~~
    - ~~o Acquirer Merchant ID is not related to the Acquirer BIN~~

  ~~Then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 303 then **ends processing**.~~

[Req 308]

Check the data elements in the ARes message as follows.

If the ACS Server Reference Number does not represent a participating ACS Server then the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code 303 and **ends processing**.

Chapter 5 EMV 3-D Secure Message Handling Requirements

5.1.4 Message ~~Protocol~~ Version Numbers

[Req 194]

A 3-D Secure Message Version Number shall be in the format major.minor~~[update]~~.patch for messaging (for example, 2.1.0).

[Req 195]

Any Message Version Number not indicated as active in Table 1.5 ~~field value less than 2.0.0~~ shall be returned as an error. The 3-D Secure component shall return an Error Message with the applicable Error Component and an Error Code = 102.

5.1.6 Message Content Validation

[Req 309]

Unless explicitly noted, if a conditionally optional or optional field is sent as empty or null, the receiving component returns an Error Message as defined in Table B.10 with the applicable Error Component and Error Code = 203.

~~Table 5.1: AReq Message Criteria~~—Table Deleted

5.4 Error Codes

[Req 218]

For Error Codes 201, ~~202,~~ 203 and 204, the 3-D Secure component that identifies the error shall include the name(s) of the erroneous data element(s) in the Error Detail.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.5.2.1 AReq/ARes Message Timeouts

[Req 228]

The 3DS Server and ACS shall set appropriate AReq and ARes message timeout values, as set by DS requirements when communicating with each DS separately.

5.5.2.3 RReq/RRes Message Timeouts

Note: No further **processing shall occur between the DS and 3DS Server as the SDK has timed out.**

5.6 PReq/PRes Message Handling Requirements

If the Serial Number has not changed, the DS would not provide back the Card Range Data element in the PRes message (i.e., it should be absent). Card Range data should not be in the PRes, however Serial Number should be included.

[Req 249]

Upon the second failure to complete the TCP/IP connection and TLS handshake to the DS, end the transaction, and periodically retry within the 24-hour window at 60-second intervals until the transaction completes successfully.

[Req ~~998~~303]

Receive and validate the PReq as defined in Table B.6:

[Req 251]

Send the PRes message containing only information about card account ranges that are participating in EMV 3-D Secure ~~2.0 or greater~~ and are registered with the DS that is responding to the request.

[Req ~~999~~304]

Receive and validate the PRes message as defined in Table B.7:

5.9.12 ACS RRes Message Error Handling—02-BRW

- For a message that cannot be recognised, the ACS:
    - If a specific transaction can be identified, the ACS sends to the 3DS Server (via Browser) a ~~CReq~~ CRes message.

Chapter 6 EMV 3-D Secure Security Requirements

6.2.2.1 3DS SDK Encryption

- If $P_{DS,}$ is an RSA key:
    - Encrypt the JSON object according to JWE (RFC 7516) using the CEK and JWE Compact Serialization. The parameter values supported in this version of the specification are:
    - “alg”:dir
    - “epk”: QSDK~~, in JSON Web Key (JWK) format~~
      {"kty": "EC",
      "crv": "P-256"}
    - “enc”:either “A128CBC-HS256” or “A128GCM”
    - ~~A128CBC-HS256 using the full CEK or~~
    - ~~A128GCM using the leftmost 128 bits of CEK and the IV~~
    - ~~“kid”:ACS Transaction ID~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

- All other parameters: not present
- If the algorithm is A128CBC-HS256 use the full CEK or if the algorithm is A128GCM use the leftmost 128 bits of the CEK.

- Makes the resulting JWE ~~object~~ available to the 3DS Server as SDK Encrypted Data.

6.2.2.2 DS Decryption

- Conducts a Diffie-Hellman key exchange process according to JWA (RFC 7518) in Direct Key Agreement mode using curve P-256, $Q_{SDK,}$ and d_{DS} to produce a CEK. The parameter values supported in this version of the specification are:
  - “alg”:ECDH-ES
  - “apv”: DirectoryserverID
  - “epk”: Q_{SDK} ~~in JSON Web Key (JWK) format~~
    {“kty”:”EC”
    “crv”:“P-256”}

6.2.3.1 3DS SDK Preparation for Secure Channel

- Generates a fresh ephemeral key pair (Q_C, d_C) as described in Annex C and provides QC as a JWK for inclusion in the AReq message.

6.2.3.2 ACS Secure Channel Set-Up

- Generates a fresh ephemeral key pair (Q_T, d_T) as described in Annex C~~and makes Q_T available for Inclusion in the ARes message~~.
- Completes the Diffie-Hellman key exchange process as a local mechanism according to JWA (RFC 7518) in Direct Key Agreement mode using curve P-256, d_T and Q_C to produce a pair of CEKs (one for each direction) which are identified by the ACS Transaction ID. The parameter values supported in this version of the specification are:
  - “alg”: ECDH-ES
  - “apv”: SDK Reference Number
  - “epk”: Q_C ~~in JSON Web Key (JWK) format~~
- Generates a digital signature of the full JSON object according to JWS (RFC 7515) using JWS Compact Serialization. The parameter values supported in this version of the specification are:
  - “alg”: PS256 or ES256
  - “x5c”: X.5C v3: Cert(Pb_{ACS}) ~~plus optionally~~and chaining certificates if present

6.2.3.3 3DS SDK Secure Channel Set-Up

- Completes the Diffie-Hellman key exchange process according to JWA (RFC 7518) in Direct Key Agreement mode using curve P-256, d_C and Q_T to produce a pair of CEKs (one for each direction), which are identified to the ACS Transaction ID received in the ARes message. The parameter values supported in this version of the specification are:
  - “alg”: ECDH-ES
  - “apv”: SDK Reference Number
  - “epk”: Q_T ~~in JSON Web Key (JWK) format~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

6.2.4.1 3DS SDK—CReq

- Encrypts the JSON object according to JWE (RFC 7516) using the $\text{CEK}_{\text{S-A}}$ obtained in Section 6.2.3.3 and JWE Compact Serialization. The parameter values supported in this version of the specification are:
  - "alg": dir
  - "enc": either:
    - A128CBC- ~~HS256 using the full CEKS-A and a fresh 128-bit random data as IV~~
    - A128GCM ~~using the leftmost 128 bits of~~ ~~$\text{CEK}_{\text{S-A}}$~~ ~~with SDKCounterStoA (padded to the left with '00' bytes) as the IV~~
  - "kid":ACS Transaction ID
  - All other parameters: not present

  If the algorithm is A128CBC-HS256 use the full $\text{CEK}_{\text{S-A}}$ and a fresh 128-bit random data as IV or if the algorithm is A128GCM use the leftmost 128 bits of $\text{CEK}_{\text{S-A}}$ with SDKCounterStoA (padded to the left with '00' bytes) as the IV.

6.2.4.4 ACS—CRes

- Encrypts the JSON object according to JWE (RFC 7516) using the same "enc" algorithm used by the 3DS SDK for the CReq message, the $\text{CEK}_{\text{A-S}}$ obtained in Section 6.2.3.2 identified by "kid" and JWE Compact Serialization. The parameter values supported in this version of the specification are:
  - "alg": dir
  - "enc": either
    - A128CBC-HS256~~using the full~~ ~~$\text{CEK}_{\text{A-S}}$~~ ~~and a fresh 128-bit random data as IV~~
    - A128GCM~~using the rightmost 128 bits of~~ ~~$\text{CEK}_{\text{A-S}}$~~ ~~with ACSCounterAtoS (padded to the left with 'FF' bytes) as the IV~~
  - "kid": ACS Transaction ID
  - All other parameters: not present

  If the algorithm is A128CBC-HS256 use the full $\text{CEK}_{\text{A-S}}$ and a fresh 128-bit random data as IV or if the algorithm is A128GCM use the leftmost 128 bits of $\text{CEK}_{\text{A-S}}$ with SDKCounterStoA (padded to the left with 'FF' bytes) as the IV.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

- **Device Channel**
  - **03-3RI**—Verification of Account

A.4 EMV 3-D Secure Data Elements

Table A.1 EMV 3-D Secure Data Elements

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS UI Type Field name: acsU~~ui~~Type							
~~Action Indicator~~ ~~Field Name: actionInd~~	~~Indicates the action to take with the card range.~~ ~~The card ranges must be processed in the order returned.~~ ~~Note: If the serial number was not included in the PReq the action is Add for all ranges returned.~~	~~DS~~	~~Length:~~ ~~1 character~~ ~~JSON Data Type: String~~ ~~Values accepted:~~ ~~A = Add the card range to the cache (default value)~~ ~~D = Delete the card range from the cache.~~	~~01-APP~~ ~~02-BRW~~	~~01-PA~~ ~~02-NPA~~	~~PRes = O~~	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Card Range Data	Card range data from the DS indicating the most recent ~~EMV 3-D Secure~~ Message Version Number supported by the ACS that hosts that range, and if configured, the ACS URL for the 3DS Method.					PRes = ~~R~~O	
Challenge Data Entry	Note: ACS UI Type = 05 is not supported.						
Challenge HTML Data Entry	Note: ACS UI Types 01, 02, 03, and 04 are not supported.						
Challenge Window Size			Length: ~~1~~ 2 character JSON Data Type: String Value accepted: 01 = 250 x 400 02 = 390 x 400 03 = 500 x 600 04 = 600 x 400 05 = Full screen				
Device Rendering Options	Note: ACS All Device Rendering Options must be supported by all components.						

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
EMV Payment Token Indicator	A value of True ~~Indicates~~ indicates that the transaction was de-tokenised prior to being received by the ACS~~sending the message~~. This data element will be populated by the system residing in the 3-D Secure domain where the de-tokenisation occurs (i.e., the 3DS Server or the DS).						
Interaction Counter			~~Begin at 1 and increment by 1 until the exchange is complete.~~				
Purchase Date & Time			Length: ~~17~~ 14 characters				
SDK Encrypted Data	Note: ~~Device Information~~This element is the only field encrypted in this version of the EMV 3-D Secure specification.						
Transaction Status Reason			22–~~50~~ 79 = Reserved for EMVCo future use ~~51~~80–99 = Reserved for DS use~~DS specific~~				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.5.3 3DS Method Data

The following table defines the data elements sent in the 3DS Method. The data is exchanged between the 3DS Requestor via the cardholder browser. ~~3DS Method Data applies to both 01-PA and 02-NPA.~~

Table A.2 3DS Method Data

Added Recipient, Message Category, and Message Inclusion columns.

A.5.5 Error Code, Error Description, and Error Detail

The following table identifies the Error Code values and specifies the associated content for Error Description and Error Detail. The information provided in the Error Description and Error Detail are guidelines on the expected content. In Error Detail if there is a mention of listing required elements the expectation is that those data elements will be listed appropriately.

Table A.4 Error Code, Error Description, and Error Detail

Value	Error Code	Error Description	Error Detail
102	Message Version Number not supported	Message Version Number received is not valid for the receiving component	All supported Protocol Version Numbers in comma a delimited list
202	Critical message ~~Data element~~extension not recognised	Critical message ~~element~~ extension not recognised	ID of ~~Name of~~ critical Message Extension(s) that was not recognised; if more than one ~~elelment~~extension is detected, this is a comma delimited list of message identifiers that were not recognised
306		~~Invalid MCC received in the AReq~~ Merchant Category Code (MCC) not valid for Payment System	~~None required.~~ For example, Invalid MCC received in the AReq
307		~~Invalid Serial number in the PReq/PRes message (too old, not found)~~ Serial Number not valid	~~None required.~~ For example, Invalid Serial number in the PReq/PRes message (too old, not found)
~~401~~	~~Unsupported device~~	~~Device type is not supported~~	~~None Required~~
402		Transaction Timed- ~~Timeout expiry reached for the transaction as defined in Section 5.5~~	For example, Timeout expiry reached for the transaction as defined in Section 5.5

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Value	Error Code	Error Description	Error Detail
403		Transient system failure ~~For example, a slowly processing back-end system~~	For example, a slowly processing back-end system
404		Permanent system failure ~~For example, a critical database cannot be accessed.~~	For example, a critical database cannot be accessed.
405		System Connection ~~Failure For example, the sending component is unable to establish connection to the receiving component.~~	For example, the sending component is unable to establish connection to the receiving component.

A.5.7 Card Range Data

Table A.6 Card Range Data

Data Element/Field Name	Description	Length/Format/Values	Inclusion
Start Protocol Version	~~Note: The Start Protocol Version shall be 2.0.1.~~Refer to Table 1.5 for active version numbers.		

A.6 Message Extension Data

Update:

Updated the JSON code sample.

A.6.3 Criticality

When an extension is critical, recipients of the message must recognise and be able to process the extension. If a 3-D Secure application ~~other than the DS~~ receives a message containing a critical extension that it does not recognise, it must treat the message as invalid and return Error Code = 202.

~~Note: DS requirements for responding to an unrecognised critical Extension element are described in Table A.4~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.1 Cardholder Account Information

Table A.8 Cardholder Account Information

Added leading zeroes to field values bringing the field length to two characters to the following fields:

- Cardholder Account Age Indicator
- Cardholder Account Password Change Indicator
- Shipping Address Usage Indicator
- Suspicious Account Activity

A.7.2 Merchant Risk Indicator

The Merchant Risk Indicator contains optional information about the specific purchase by the Cardholder. The detailed data elements, which are optional are outlined in Table A.9.

Table A.9 Merchant Risk Indicator

Added leading zeroes to the values of the following field:

- Delivery Timeframe

A.7.3 3DS Requestor Authentication Information

Table A.10 3DS Requestor Authentication Information

3DS Requestor Authentication Method: Adjusted the Reserved for EMVCo future use from 05–~~80~~ to 05–79

A.7.6 Device Rendering Options Supported

Note: All Device Rendering Options must be supported by all components.

Table A.13 Device Rendering Options Supported

Added leading zeroes to field values bringing the field length to two characters to the following fields:

- Interface
- UI Type

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Annex B Message Format

Table B.1 AReq Message Data Elements

~~ACS Challenge Mandated Indicator~~

Table B.2 ARes Message Data Elements

~~ACS Ephemeral Public key (QT)~~

Table B.3 CReq Message Data Elements

~~SDK Ephemeral Public Key (QC)~~

Table B.4 CRes Message Data Elements

Field Name: ~~uiType~~ to acsUiType

Table B.7 PRes Message Data Elements

~~Action Indicator~~

Annex D Approved Transport Layer Security Versions

D.1.2 Not Supported:

The following cipher suites shall not be presented or accepted:

- Any cipher suite represented as ‘Null’, ‘Anonymous/Anon’
- Any cipher suite incorporating any of the algorithms ‘RC2’, ‘RC4’, ‘DES’, ‘IDEA’, KRB5’, ‘SEED’, ‘ARIA’, ‘CHACHA20’ or ‘MD5’
- Any cipher suite incorporating an export grade algorithm using ‘EXPORT’.

Note: 3DES and SHA-1 are to be phased out and may become unsupported algorithms in future versions of this specification.

~~D.1.2 TLS Cipher Suites 1.2 Should support:~~

- ~~TLS_RSA_WITH_AES_128_GCM_SHA256~~
- ~~TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256~~
- ~~TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256~~

~~D.1.3 TLS Cipher Suites 1.2 May support:~~

- ~~TLS_RSA_WITH_AES_256_GCM_SHA384~~
- ~~TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384~~
- ~~TLS_RSA_WITH_AES_256_CBC_SHA~~
- ~~TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA~~
- ~~TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA~~
- ~~TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

- ~~TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA~~

D.2.1 Implementation Notes

~~RC4 is not an approved algorithm so any Cipher Suite incorporating RC4 shall be disabled.~~

~~Most implementations supporting TLS 1.1 also support TLS 1.2, thus only TLS 1.2 is approved.~~

The use of 3DES and SHA (SHA-1) are to be phased out. They ~~are included here under “may be supported” for legacy reasons only and~~ may be deprecated in future versions of this specification.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

April Publication of EMV 3-D Secure Protocol and Core Functions Specifications, Version 2.0.1

Chapter 1 Introduction

Table 1.3—Definitions

- EMV Payment Token
- Registered Application Provider Identifier (RID)

Chapter 2 EMV 3-D Secure Overview

2.5.2 Challenge Flow

Note: Processing Payment EMV Payment Tokens:

General configuration

In order for this specification to appropriately handle flows that initiate with EMV Payment Tokens, the Payment Token ranges must be shared and configured on the appropriate DS. This is essential for the EMV Payment Token transaction to be routed to the appropriate DS and then to the ACS.

EMV Payment Token handling during transaction flow

During the Authentication Request (AReq) message flow, it might be necessary for the Payment Token to be detokenized and the actual PAN to be placed in the Cardholder Account Number for the remainder of the AReq flow. For instance, this could be required if the transaction initiated with an EMV Payment Token and the ACS was configure based off of the PAN. In this case, the EMV Payment Token Indicator in the AReq message is set to True to indicate that the transaction initiated with an EMV Payment Token.

Chapter 3 EMV 3-D Secure Authentication Flow Requirements

3.1 App-based Requirements

Step 2: The 3DS Requestor App

- The 3DS Requestor App uses the Cardholder Account Number and optionally other cardholder information to identify the ~~DS~~ Payment System. Payment Systems are identified by their ISO RID (as defined in ~~DS~~ Table 1.2).

- The 3DS Requestor App invokes ~~a~~ the createTransaction method within the 3DS SDK to initiate 3-D Secure Cardholder authentication ~~by obtaining the SDK Transaction ID, SDK App ID, and the Device Information~~. The 3DS Requestor App provides the SDK the DirectoryServerID value which is the Payment Systems RID and obtains the SDK Transaction ID, SDK App ID, and the Device Information.

  Note: As described in the EMV 3DS SDK Specification, the 3DS SDK encrypts the device information by using the DS public key. This key is identified based on the DirectoryServerID that is passed to the createTransaction method. DirectoryServerID is defined in the *EMV 3-D Secure SDK Specification*.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Step 5: The 3DS Server

[Req 300]

Ensure, if the 3DS Requestor and 3DS Server are separate components, that data transferred between them is protected at a level that satisfies PCI DSS with mutual authentication of both servers.

If the communication is not established as required, the 3DS Server **ends 3-D Secure processing**.

[Req 8]

Generate the 3DS Server Transaction ID.

~~This ID will, for the 3DS Server, uniquely identify this transaction within all messages in the authentication process (AReq/ARes, CReq/CRes, and RReq/RRes).~~

Step 6: The DS

[Req 16]

Generate the DS Transaction ID.

~~For the DS, the DS Transaction ID uniquely identifies the transaction in all subsequent messages in the authentication process (AReq/Ares, and RReq/RRes).~~

[Req 17]

Decrypt the SDK Encrypted Data data element of the AReq message as defined in Section 6.2.2.2 and move the SDK Encrypted Data data content ~~(decrypted during the validation process defined in Section 5.9.1)~~ to the Device Information data element of the AReq message for the ACS. If decryption fails, the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 302 and **ends processing**

[Req 23]

Establish a secure link with the ACS as defined in Section 6.1.3.1.

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 405 with detailed information about the issue in Error Description and **ends processing**.

[Req 24]

Send the AReq message to the ACS using the secured link established in Seq 3.24.

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 402 with detailed information about the issue in Error Description and **ends processing**.

Step 18: The ACS

[Req 64]

If the Cardholder abandons the challenge during the processing of Step 16 and Step 17, or the ACS has received an abandonment CReq message from the 3DS SDK (as defined in *[Req 61]* Seq 3.62), then the ACS sets the Challenge Completion Indicator = N in the CRes message~~ and sets the Challenge Cancelation Indicator to the appropriate value in the RReq message~~. Refer to Annex A for the specific values.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

3.3 Browser-based Requirements

Step 5: The 3DS Requestor Environment

[Req 301]

Ensure, if the 3DS Requestor and 3DS Server are separate components, that data transferred between them is protected at a level that satisfies PCI DSS with mutual authentication of both servers.

- If the communication is not established as required, the 3DS Server ends 3-D Secure processing.

Step 7: The DS

[Req 100]

Establish a secure link with the ACS as defined in Section 6.1.3.1.

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 405 with detailed information about the issue in Error Description and ends processing.

[Req 101]

Send the AReq to the ACS using the secured link established in Seq 3.101.

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 402 with detailed information about the issue in Error Description and ends processing.

Step 8: The ACS

[Req 109]

If a challenge is deemed necessary, (Transaction Status = C) the ACS determines whether an acceptable challenge method is supported by the 3DS Server based on the Device Channel data element received in the AReq message. The ACS performs the following:

b. Sets the ACS URL field in the Ares message that will be utilised in the Browser to ACS link ~~as defined in *[Req 100]*~~.

c. Stores the 3DS Server Transaction ID, DS URL, and DS Transaction ID (for subsequent RReq processing).

Step 10: The 3DS Server

[Req 117]

For a transaction with a challenge (Transaction Status = C):

- If the 3DS Requestor accepts the challenge:
  - o Validate the signature of the ACS Signed Content data element (as defined in 6.2.3.3) and sSend necessary information from the ARes message to the 3DS Requestor Environment.

Step 16: The ACS

[Req 126]

If the Cardholder abandons the challenge during the processing of Step 12 through Step 14, then the ACS sets the ~~Challenge Completion Indicator =N in the RReq message and sets the~~ Challenge Cancelation Indicator to the appropriate value in the RReq message. Refer to Annex A for the specific values.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

3.4 3RI-based Requirements

Step 2: The 3DS Server

[Req 302]

Generate the 3DS Server ID.

Step 3 The DS:

[Req 284]

~~Store the 3DS Server URL with the DS Transaction ID (for possible RReq processing).~~

[Req 285]

Establish a secure link with the ACS as defined in Section 6.1.3.1.

If the connection cannot be established with the ACS, the DS returns to the 3DS Server EITHER:

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 405 with detailed information about the issue in Error Description

[Req 286]

Send the AReq to the ACS using the secured link established in *[Req 285]*.

If the DS does not receive an ARes message from the ACS (as defined in Section A.5.5), the DS returns to the 3DS Server EITHER:

- an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = ~~403~~ 402 with detailed information about the issue in Error Description and ends processing.

Chapter 4 EMV 3-D Secure User Interface Templates, Requirements, and Guidelines

4.2.1 Processing Screen Requirements

[Req 147]

Create the Processing Screen with just the default Processing Graphic (for example, a progress bar or a spinning wheel) of the Consumer Device OS (refer to Figure 4.~~3~~4 for an example).

[Req 151]

Display the Processing screen during the ~~entire~~ CReq/CRes message cycle.

~~[Req 149]~~

~~If requested, integrate the DS Logo into the Processing screen.~~

~~[Req 150]~~

~~Store the DS logo.~~

~~[Req 152]~~

~~Display the Processing screen for a minimum of two seconds.~~

New:

Figure 4.4 Sample App-based Processing Flow

Figure 4.10 Sample Challenge Information Text Indicator—Payment Authentication

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Notes:

- **There is no difference in Native vs. HTML experience.**
- **The SDK could dim the challenge window during spinning wheel display or the wheel could be translucent.**
- **The spinning wheel would be displayed through the timeout/retry process.**

4.2.3.3 3DS SDK

The 3DS ~~Requestor~~ SDK shall:

4.2.5.3 3DS SDK

[Req 171]

Return control to the 3DS Requestor App when the Cancel action in the 3DS Requestor header is selected.

- On HTML submit:
    - o The SDK passes the received data, unchanged, to the ACS in the ACS HTML data entry data element of the CReq message. The SDK shall not modify or reformat the data in any way.

The 3DS ~~Server~~ SDK transmits the CReq message to the ACS.

4.3.1.2 ACS:

[Req 178]

Create a Processing screen for display during the HTML exchange CReq/CRes message cycle.

Chapter 5 EMV 3-D Secure Message Handling Requirements

5.1.6 Message Content Validation

[Req 209]

If there are additional data elements received that are not specified for the Message Type, ~~and~~ Message Version Number, Device Channel, and Message Category but the message otherwise passes validation, the message shall be considered valid.

For example, the DS receives an AReq message from the 3DS Server with additional data elements that are not specified in Table A.1 for the AReq message type, Device Channel and Message Category and the DS validates the AReq content, and drops the additional elements when sending the AReq to the ACS.

[Req 213]

Upon finding any message data elements that do not pass format validation, the validating component shall generate a response message having the Error ~~Description~~ Detail include the data element name(s) of the incorrect elements populated.

[Req 211]

~~If the DS receives unrecognised non-critical message extensions, they shall be passed to the ACS in the AReq message or to the 3DS Server in the ARes or RReq messages.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

[Req 214]

~~The AReq message validation shall be based on the two data elements: Device Channel and Message Category and the presence of a data element in the AReq message shall be validated based on the content of these two data elements as defined in Table A.1.~~

Table 5.1 AReq Message Criteria:

		~~Device Channel~~ **Message Category**	
		01 = ~~App~~Payment Authentication	**02 = ~~Browser~~Non-Payment Authentication**
~~Message Category~~ **Device Channel**	01 = ~~Payment Authentication~~App	01, 01	01, 02
	02 = ~~Non-Payment Authentication~~Browser	02, 01	02, 02
	03 = 3RI	03, 01	03, 02

5.4 Error Codes

Update:

[Req 218]

For Error Codes 201, 202, 203 and 204, the 3-D Secure component that identifies the error shall include the name(s) of the erroneous data element(s) in the Error ~~Description~~Detail.

5.5.1 Transaction Timeouts

[Req 221]

If the transaction reaches the 30-second timeout expiry, send an RReq message to the DS to be passed to the 3DS Server with Transaction Status = N and Transaction Status Reason = ~~13~~ 14 (Challenge Transaction Timed Out) and clear the ephemeral key generated and stored for use in the CReq/CRes message exchange for the current transaction.

[Req 224]

If the timeout expires before receiving the next CReq message from the 3DS SDK, send an RReq message to the DS to be passed to the 3DS Server with Transaction Status = N, Transaction Status Reason = 14~~3~~ (Challenge Transaction Timed Out), and Challenge Cancelation Indicator = 04 and then clear any ephemeral key generated and stored for use in the CReq/CRes message exchange for this transaction.

[Req 225]

Upon receiving the CReq message for a transaction that has timed out, ~~send a CRes message with IReq Code 13 (Transaction Timed-Out)~~ send an Error Message with Error Component = A and Error Code = 402 to the 3DS SDK.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.5.2.1 AReq/Ares Message Timeouts

[Req 229]

Any failure to complete the initial TCP/IP connection and TLS handshake to the DS shall result in an immediate retry or the 3DS Server shall try an alternate DS (if available). Upon second failure, the 3DS Server shall send an error to the 3DS Requestor to complete the transaction.

[Req 233]

Any failure to complete the initial TCP/IP connection and TLS handshake to the ACS shall result in an immediate retry or the DS shall try an alternate ACS (if available). Upon second failure, the DS shall send an Error Message with Error Component = D and Error Code = ~~402~~ 405 to the 3DS Server to complete the transaction.

Note: The DS may maintain multiple ACS URLs. If the first URL attempted is not available, then the DS will attempt to connect to one of the alternate URLs.

[Req 237]

The 3DS SDK shall set a 10~~5~~ second timeout value from the time the TLS handshake has completed and the full CReq message is sent for processing to the ACS.

[Req 239]

~~After the retry, I~~If the ACS does not respond with the CRes message before the second 3DS SDK 10~~5~~-second read timeout expiry, the 3DS SDK ends 3-D Secure processing, and passes an error message to the calling app, ending the 3-D Secure transaction.

~~**[Req 230]**~~

~~Upon the second failure to complete the TCP/IP connection and TLS handshake to a DS, the 3DS Server shall try an alternate DS (if available), or stop attempting connections for 30 seconds, and then retry the connection~~

~~**[Req 238]**~~

~~If the ACS does not respond with the CRes message before the 3DS SDK 5-second read timeout expiry limit, the 3DS SDK shall retry sending the CReq message with appropriate UI messaging to the user (for example, display the processing screen).~~

5.5.2.3 RReq/RRes Message Timeouts

[Req 240]

Any failure to complete the initial connection and TLS handshake to the DS shall result in an immediate retry. Upon second failure, the ACS will wait 10 seconds and retry to connect to the DS until the message is delivered ~~to the DS~~ with Transaction Status = U.

[Req 242]

If the DS has not responded with the RRes message before the 5-second read timeout expiry, the ACS shall close and re-establish a new connection and resend with Transaction Status = U.

[Req 243]

Any failure to complete the initial connection and TLS handshake to the 3DS Server shall result in an immediate retry. Upon second failure, the DS shall send an Error Message with Error Component = D and Error Code = ~~402~~ 405 to the ACS to complete the transaction.

Note: No further processing shall occur between the DS and 3DS Server as the SDK has timed out.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

[Req 245]

If the 3DS Server has not sent the RRes message before the 3-second read timeout expiry, the DS shall ~~end processing~~ send an Error Message with Error Component = D and Error Code = 402 to the ACS to complete the transaction.

Note: No further processing shall occur between the DS and 3DS Server as the SDK has timed out.

5.6 PReq/PRes Message Handling Requirements

The PReq/PRes messages are utilised by the 3DS Server to cache information about the version supported by available ACSs and also any URL to be used for the 3DS Method call. The data will be organised by card range as configured by a DS. The information provided on the protocol versions supported by ACS's can be utilized in both the App-based and Browser-based flows.

The 3DS Server formats a PReq message, as described in Table A.1, and sends the request to the DS. If this is the first time the cache is being loaded (or if the cache has been flushed and needs to be reloaded, or if the DS does not support partial cache updates), the Serial Number element is not included in the request, which will result in the DS returning the entire list of participating card range information.

Otherwise, the 3DS Server should include the Serial Number from the most recently processed PRes, which will result in the DS returning only the changes since the previous PRes.

The DS manages the Serial Number to ensure that the response to a PReq for a particular Serial Number includes all updates posted since that Serial Number was issued. If the Serial Number provided in the PReq is invalid (for example, if it is too old and can no longer be found), the response should be an Error Message with an Error Code = 307.

If the PReq does not include a Serial Number, the DS PRes response must contain all card range entries.

[Req 998]

Receive and validate the PReq as defined in Table B.6:

- If any data element present fails validation, the DS:
  - Returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 203.
- If any required data elements are missing, the DS:
  - Returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 201.
- If the Serial Number is invalid, the DS:
  - Returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 307.

[Req 999]

Receive and validate the PRes message as defined in Table B.7:

- If any data element present fails validation, the 3DS Server:
  - Returns to the DS an Error Message (as defined in Section A.5.5) with Error Component = S and Error Code = 203.
- If any required data elements are missing, the 3DS Server:
  - Returns to the DS an Error Message (as defined in Section A.5.5) with Error Component = S and Error Code = 201.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

5.9.1 DS AReq Message Error Handling

- ~~For Device Channel = 01-App, decrypts the SDK Encrypted Data data element of the AReq message as defined in Section 6.2.2.2 :~~
  - If decryption fails, the DS returns to the 3DS Server an Error Message (as defined in Section A.5.5) with Error Component = D and Error Code = 302.

5.9.3.1 Message in Error

If a specific transaction can be identified, sends to the 3DS Server using the secure link established in Seq 3.13 for an app-based transaction, ~~or~~ Seq 3.91 for a browser-based transaction, or ***[Req 275]*** for a 3RI transaction EITHER:

5.9.3.2 Error Message Received

If a specific transaction can be identified, sends to the 3DS Server using the secure link established in Seq 3.13 for an app-based transaction, ~~or~~ Seq 3.91 for a browser-based transaction, or ***[Req 275]*** for a 3RI transaction Seq 3.13 EITHER:

5.96 ACS CReq Message Error Handling—02-BRW

- Sends to the DS an RReq message (as defined in Table B.8) with Transaction Status = U and Challenge Cancelation Indicator = ~~6~~ 05 using the secure link.

5.9.8.1 Message in Error

- Sends to the 3DS Server an RReq message (as defined in Table B.8) with Transaction Status = U and Challenge Cancelation Indicator = ~~6~~ 05 using the secure link.

Chapter 6 EMV 3-D Secure Security Requirements

6.1.1 Link a: Consumer Device—3DS Requestor

The 3DS Requestor link established between the 3DS Requestor and the Consumer device uses a ~~standard~~ TLS Internet protocol as part of the interaction between the 3DS Requestor App or the browser on the Consumer device and the 3DS Requestor.

6.1.4.1 For App-based CReq/Cres

- Protocol—~~standard~~ TLS Internet

6.1.4.2 For Browser-based CReq/Cres

- Protocol—~~standard~~ TLS Internet

6.1.8 Link h: Browser—ACS (for 3DS Method)

- Protocol—~~standard~~ TLS Internet

6.2.2.1 3DS SDK Encryption

- "apv":~~DS Reference Number~~DirectorysServerID

6.2.2.2 DS Decryption

- "apv":~~DS Reference Number~~DirectorysServerID

6.2.3.2 ACS Secure Channel Set-Up

Note: Key separation is good practice for opposite directions of the 3DS SDK to ACS link. In this version of the specification CEKA-S and CEKS-A are extracted with the same value.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Thus, for A128CBC-HS256 the same 256-bit key will be used in both directions. For A128GCM the key is split as two 128 bit components, resulting in separate keys for each direction.

- Includes the resulting JWS in the ARes message as ACS Signed ~~Data~~ Content

6.2.4.1 3DS SDK—CReq

- Creates a JSON object of the data elements identified in the CReq message defined in Section B.3~~ appended with SDKCounterStoA~~.

6.2.4.2 3DS SDK—CRes

- Decrypts the message according to JWE (RFC 7516) using the CEKA-S obtained in Section 6.2.3.3 identified by “kid” If the algorithm is A128GCM the rightmost 128 bits of CEKA-S is used with SDKCounterAtoS (padded to the left with ‘FF’ bytes) as the IV. If decryption fails, ceases processing and reports error.

6.2.4.3 ACS—CReq

- Decrypts the message according to JWE (RFC 7516) using the CEKS-A obtained in Section 6.2.3.2 identified by “kid”. If the algorithm is A128GCM the leftmost 128 bits of CEKS-A is used with ACSCounterStoA (padded to the left with ‘00’ bytes) as the IV. If decryption fails, ceases processing and reports error.

6.2.4.4 ACS—CRes

- Creates a JSON object of the data elements identified in the CRes message defined in Section B.4~~ appended with ACSCounterAtoS~~.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Annex A 3-D Secure Data Elements

Format: is updated to JSON Data Type: for each data element in Annex A. Individual instances are not captured in this Bulletin.

Note: The values of 0 and 00 are not supported in this Annex.

Table A.1 EMV 3-D Secure Data Elements

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
3DS Requestor Challenge Indicator			05–79 = Reserved for EMVCo future use 80-99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
3DS Requestor Initiated Indicator			05~~–80~~79 = Reserved for EMVCo future use 80~~10–~~99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
3DS Requestor Name			Values accepted: Any individual DS may impose specific formatting and character requirements on the contents of this field.				
3DS Requestor Non-Payment Authentication Indicator			04~~–80~~79 = Reserved for EMVCo future use Note: The EMVCo and DS reserved values are invalid until specifically defined.				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
3DS Server Reference Number			Value accepted: Set by the EMVCo Secretariat				
3DS Server Operator ID Field Name: threeDSServerOperatorID	DS assigned 3DS Server identifier. Each DS can provide a unique ID to each 3DS Server on an individual basis.	3DS Server	Length: Variable, maximum 32 characters JSON Data Type: String Value accepted: Any individual DS may impose specific formatting and character requirements on the contents of this field.	01-APP 02-BRW 03-3RI	01-PA 02-NPA	AReq = O PReq = O	
3DS Server Transaction ID			Value accepted: ~~UUID~~				
Account Type			04~~3–~~79~~29~~ = ~~DS or Payment System-specific~~ Reserved for EMVCo future use 80~~30–~~99 = DS or Payment System-specific Note: The EMVCo and DS reserved values are invalid until specifically defined.~~Reserved for EMVCo future use~~				
Acquirer BIN			Length: Variable, maximum 11 characters				
ACS Counter ACS to SDK acsCounterAtoS	Counter used as a security measure in the ACS to 3DS SDK secure channel.	ACS	Length: 3 characters JSON Data Type: String	01 = APP	01-PA 02-NPA	01-PA: CRes = R 02-NPA CRes = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS Operator ID Field Name: acsOperatorID	DS assigned ACS identifier. Each DS can provide an unique ID to each ACS on an individual basis.	ACS	Length: Variable, maximum 32 characters JSON Data Type: String Value accepted: Any individual DS may impose specific formatting and character requirements on the contents of this field.	01-APP 02-BRW 03-3RI	01-PA 02-NPA	ARes = O	
ACS Ephemeral Public Key (Q_T) Field Name: acsEphemPubKey ~~(ACS Signed Content)~~	Public key component of the ephemeral key pair ~~(dT, QT)~~ generated by the ACS and used to establish session keys between the 3DS SDK and the ACS. ~~Note:~~ The data element is ~~will be~~ contained within ~~the~~ ACS Signed Content JWS Object ~~and will not be populated in the ARes in its own field~~. See Section 6.2.3.2and ~~Annex C~~ for additional detail.					~~ARes = C~~See ACS Signed Content	See ACS Signed Content~~Required if Transaction Status = C~~ ~~This data element will be contained within the ACS Signed Content field, and will not be a unique field.~~
ACS Reference Number	~~Unique identifier for the ACS that is assigned by a DS when the ACS is registered~~ Unique identifier assigned by the EMVCo Secretariat upon testing and approval.		Value accepted: Set by the EMVCo Secretariat				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS Rendering Type	~~This field will contain a JSON array with two values, ordered.~~		~~Length: 2 character~~ Note: Data will be formatted into a JSON object prior to being placed into the ACS Rendering Type field of the message. ~~Interface (Position 1):~~ ~~01 = Native UI~~ ~~02 = HTML UI~~ ~~UI Type (Position 2):~~ ~~01 = Text~~ ~~02 = Single Select~~ ~~03 = Multi Select~~ ~~04 = OOB~~ ~~05 = HTML Other~~			RReq = ~~C~~R	For ARes, Required if Transaction Status = C. ~~Otherwise omitted.~~
ACS Signed Content Field Name: acsSignedContent ~~(Signed Content)~~							~~Conditionally~~ Required if the Transaction Status = C.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
ACS UI Type			Length: ~~1~~ 2 character Values accepted: 01 = Text 02 = Single Select 03 = Multi Select 04 = OOB 05 = HTML 06–79 = Reserved for EMVCo future use 80–99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
ACS URL ~~(ACS Signed Content)~~	For App-based, this data element is contained within the ACS Signed Content JWS Object For Browser-based, this data element is present as its own object.					01-APP, see ACS Signed Content 02-BRW, ARes = C	01-APP, see ACS Signed Content. 02-BRW, required if Transaction Status = C.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Action Indicator Field Name: actionInd	Indicates the action to take with the card range. The card ranges must be processed in the order returned. Note: If the serial number was not included in the PReq the action is Add for all ranges returned.	DS	Length: 1 character JSON Data Type: String Values accepted: A = Add the card range to the cache (default value) D = Delete the card range from the cache.	01-APP	01-PA 02-NPA	PRes = O	
Address Match Indicator							Required in 01-PA if 3DS Requestor has ~~Billing and~~ Shipping Address information
Authentication Method			11–~~80~~79 = Future EMVCo Use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
Browser Accept Headers							~~All transaction originating 02-BRW shall include this field.~~
Browser IP Address							~~All transaction originating 02-BRW shall include this field.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Browser Java Enabled			~~Length: Variable, 4–5 characters~~				~~All transaction originating 02-BRW shall include this field.~~
Browser Language							~~All transaction originating 02-BRW shall include this field.~~
Browser Screen Color							~~All transaction originating 02-BRW shall include this field.~~
Browser Screen Height							~~All transaction originating 02-BRW shall include this field.~~
Browser Screen Width							~~All transaction originating 02-BRW shall include this field.~~
Browser Time Zone							~~All transaction originating 02-BRW shall include this field.~~
Browser user Agent							~~All transaction originating 02-BRW shall include this field.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Card Range Data	~~Note: The 3DSMethodURL may be omitted if not required.~~ *Examples Deleted		~~startRange: 13–19 characters~~ ~~endRange: 13–19 characters~~ ~~protocolVersion: 5 characters~~ ~~3DSMethodURL: Variable, maximum 256 characters~~ Values accepted: See Table A.6 for additional information. Note: Data will be formatted into a JSON object prior to being placed into the Card Range Data field of the message.				
Cardholder Account Information			Note: Data will be formatted into a JSON object prior to being placed into the Card Range Data field of the message.				
Cardholder Billing Address Line 3 Field Name: billAddrLine3	Third line of the street address or equivalent local portion of the Cardholder billing address associated with the card used for this purchase.	3DS Server	Length: Variable, maximum 50 characters JSON Data Type: String	01-APP 02-BRW 03-3RI	01-PA 02-NPA	AReq = C	Required if available unless there is a market or regional mandate to restrict sending of this information
Cardholder Billing Address State			Length: Variable, maximum 3 characters				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Cardholder Shipping Address Line 3 Field Name: shipAddrLine3	The third line of the street address or equivalent local portion of the shipping address requested by the Cardholder.	3DS Server	Length: Variable, maximum 50 characters JSON Data Type: String	01-APP 02-BRW 03-3RI	01-PA 02-NPA	AReq = C	Required unless there is a market or regional mandate to restrict sending of this information
Cardholder Work Phone Number	~~A JSON object with the Country Code and Subscriber sections of the number, represented by the following named fields:~~		Values accepted: Country Code and Subscriber sections of the number, represented by the following named fields: • cc • subscriber				
Challenge Additional Information Text			If field is populated this information must be displayed to the cardholder			CRes = ~~C~~O	~~Required based upon the ACS UI format selected.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Challenge Cancelation Indicator			Length: ~~1~~ 2 character Values accepted: ~~01 = Cardholder selected “Choose another payment method.” CReq 01-PA only~~ ~~02~~ 01 = Cardholder selected “Cancel” ~~& Continue Shopping.” CReq 01-PA only~~ ~~03~~ 02 = 3DS Requestor cancelled Authentication. ~~CReq 01-PA, 02-NPA-~~ 03 = Transaction Abandoned 04 = Transaction Timed Out at ACS. ~~RReq only, 01-PA, 02-NPA-~~ 05~~6~~ = Transaction Error. ~~CReq/RReq 01-PA, 02-NPA-~~ 06~~7~~ = Unknown 07-79 = Future EMVCo Use 80–99 = DS Future Use Note: The EMVCo and DS reserved values are invalid until specifically defined.	02-BRW			Required ~~if~~ in the CReq if the authentication transaction was cancelled by user interaction with the cancelation button in the user interface or for ~~any~~ other reasons as indicated. Required in the RReq if the ACS identifies that the authentication transaction was cancelled for reasons as indicated.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Challenge Data Entry							Required when ACS UI Type = 01, 02, 03, or 04 and challenge data has been entered into the UI.~~Required if the Cardholder entered information for the ACS UI to validate.~~ ~~Conditional for Native UI.~~
Challenge Information Header			If field is populated this information must be displayed to the cardholder			CRes = ~~C~~O	~~Required if ACS UI Type = 1–4.~~
Challenge Information Label	Label to modify the “Challenge Data Entry” field ~~text~~ provided by the Issuer ~~to describe what is being requested from the Cardholder~~		If field is populated this information must be displayed to the cardholder			CRes = ~~C~~O	~~Required if ACS UI Type = 1–4.~~
Challenge Information Text			If field is populated this information must be displayed to the cardholder			CRes = ~~C~~O	~~Required if ACS UI Type = 1–4.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Challenge Information Text ~~Colour~~ Indicator Field Name: ~~challengeInfoTextColour~~ challengeInfoTextIndicator	Indicates when the Issuer/ACS would like a warning icon or similar visual indicator to draw attendtion to the “Challenge Iinformation Ttext” that is being displayed. ~~should be displayed in a different colour.Defined in w3c HTML data types.~~ ~~Example: #F53506~~		Length: ~~7~~1 characters Y = Display indicator N = Do no display indicator If field is populated this information must be displayed to the cardholder.			CRes = ~~C~~O	~~Required if ACS UI Type = 1–4.~~
ACS Challenge Mandated Indicator Field Name: acschallengeMandated			Y = ~~A challenge will be required by the Issuer as a condition for approving the transaction.~~A challenge is mandated N = A challenge is not ~~required by the Issuer as a condition for approving the transaction.~~mandated			~~AReq = O~~	
Challenge Selection Information			Note: Data will be formatted into a JSON object prior to being placed into the Challenge Selection Information field of the message. If field is populated this information must be displayed to the cardholder			Cres = CO	~~Required if the ACS UI Type = 2 or 3~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Device Channel			04–79 = Reserved for EMVCo future use 80–99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
Device Information Field Name: deviceInfo ~~(SDK Encrypted Data)~~	This will be populated by the DS as unencrypted data to the ACS obtained from ~~Contained in~~ SDK Encrypted Data ~~as the body of the JWE Object~~	DS ~~originating from the 3DS SDK to the ACS~~				AReq = C~~R~~	Required ~~if there is no market or regional mandate to restrict sending of this information~~.between the DS and ACS but will not be present from 3DS Server to DS

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Device Rendering Options Supported	~~This field will contain a JSON array with two objects:~~ ~~interface = selected number value~~ ~~UI = JSON Array with all selected values.~~ ~~Example:~~ ~~“deviceRenderOptions” = [{ “interface” = 3 } , { “UI” = [1,2,3,4] }]~~		Note: Data will be formatted into a JSON object prior to being placed into the Device Rendering Options Supported field of the message.~~interface (Position 1 value):~~ ~~1 = Native~~ ~~2 = HTML~~ ~~3 = Both~~ ~~UI (Position 2, JSON array as value):~~ ~~1 = Text~~ ~~2 = Single Select~~ ~~3 = Multi Select~~ ~~4 = OOB~~ ~~5 = HTML Other~~				
Error Message Type						Erro = C~~R~~	Conditional on Message Type being recognizable
~~Expandable Information Label 2~~ ~~Field Name:~~ ~~expandInfoLabel2~~	~~Label to be displayed to the Cardholder for the content in Expandable Information Text 2.~~	~~ACS~~	~~Length: Variable, maximum 45 characters~~ ~~Format: String~~	~~01-APP~~	~~01-PA~~ ~~02-NPA~~	~~CRes = C~~	~~Required based upon the ACS UI Type selected.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
~~Expandable Information Text 2~~ ~~Field Name: expandInfoText2~~	~~Text provided by the Issuer from the ACS to be displayed to the Cardholder for additional information and the format will be an expandable text field.~~	~~ACS~~	~~Length: Variable, maximum 256 characters~~ ~~Format: String~~	~~01-APP~~	~~01-PA~~ ~~02-NPA~~	~~CRes = C~~	~~Required based upon the ACS UI Type selected.~~
Instalment Payment Data							Required if the Merchant and Cardholder have agreed to instalment payments. Omitted if not an instalment payment authentication.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Issuer Image	~~Three fully qualified URLs with small, medium and large images to be loaded and cached for use in the current challenge. SDK to select size appropriate for the current device screen resolution. Example:~~ ~~“issuerImage” =~~ ~~{“small”: “http://acs.com/small_image.jpg,~~ ~~”medium”: “http://acs.com/medium_image.jpg”,~~ ~~“large”: }~~ ~~Option 2: May also be ‘none” if no image is to be displayed.~~ ~~Example:~~ ~~“issuerImage” = “none”~~		~~Length: Variable, maximum 2048 characters~~ ~~Option 1:~~ ~~JSON Object, string, values as follows:~~ ~~“small”: Fully qualified URL of small image resource~~ ~~“medium” Fully qualified URL of medium image resource~~ ~~“large”– Fully qualified URL of large image resource~~ ~~Option 2:~~ ~~String containing the value “none”~~Refer to Table A.14 for data elements.				
Merchant Category Code	~~The same value must be used in the authorisation request. Supported values are specified by each Payment System or DS.~~						

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Merchant Country Code	This value correlates to the Merchant Country Code as defined by each Payment System or DS.~~The same value must be used in the authorisation request.~~						
Merchant Risk Indicator			Note: Data will be formatted into a JSON object prior to being placed into the Device Merchant Risk Indicator field of the message				
Message Category			03–79 = Reserved for EMVCo future use 80-99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
OOB App URL						CRes = ~~C~~O	~~Required for ACS UI Type = 4 or 5~~ Note: this element has been defined to support future enhancements to the OOB message flow. The 3DS SDK will not perform any processing of the OOP App URL in this version of the specification.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Payment System Image	~~Option 1:~~ ~~Three fully qualified URLs with small, medium and large images to be loaded and cached for use in the current challenge. SDK to select size appropriate for the current device screen resolution.~~ ~~Example:~~ ~~“psImage” =~~ ~~{“small”: “http://ds.com/small_image.jpg,~~ ~~”medium”: “http://ds.com/medium_image.jpg”,~~ ~~“large”: “http://ds.com/large_image.jpg”~~ ~~}~~		~~Length: Variable, maximum 2048 characters~~ ~~Option 1:~~ ~~JSON Object, string values as follows~~ ~~“small”: Fully qualified URL of small image resource~~ ~~“medium”: Fully qualified URL of medium image resource~~ ~~“large”: Fully qualified URL of large image resource~~ ~~Option 2:~~ ~~String containing the value “none”~~				
Resend Challenge Information Code							Required for Native UI if the Cardholder is requesting ~~for~~ the ACS to resend challenge information. ~~Conditional for Native UI.~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Resend Information Label							Required for Native UI ~~when the ACS~~ if the ACS is allowing the Cardholder to request resending authentication information.~~UI Type = 1, 2, or 3~~
Results Message Status			Length: 2~~1~~ character 01 = Results Request Received for further Processing 02 = Challenge Request not sent to ACS by 3DS Requestor (3DS Server or 3DS Requestor opted out of the challenge) 03 = ARes challenge data not ~~passed~~ delivered to the 3DS Requestor due to technical error 04–79 = Reserved for EMVCo future use 80-99 = Reserved for DS use Note: The EMVCo and DS reserved values are invalid until specifically defined.				
SDK Counter SDK to ACS Field Name: s dkCounterStoA	Counter used as a security measure in the 3DS SDK to ACS secure channel.	3DS SDK	Length: 3 character JSON Data Type: String	01-APP	01-PA 02-NPA	01-PA: CReq = R 02-NPA CReq = R	

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	**Description**	**Source**	**Length/Format/ Values**	**Device Channel**	**Message Category**	**Message Inclusion**	**Condition Inclusion**
SDK Ephemeral Public Key (QC) Field Name: sdkEphemPubKey ~~(Signed Content)~~	Public key component of the ephemeral key pair ~~(dc, Qc)~~ generated by the 3DS SDK and used to establish session keys between the 3DS SDK and ACS. ~~In cryptographic algorithms throughout this specification, this value is identified as "Qc".~~ In AReq, this data element is present as its own object. ~~Note:~~ In ~~the~~ ARes, this data element is~~will be~~ contained within the ACS Signed Content JWS Object ~~and will not be populated in the ARes in its own field~~.	~~(sent via 3DS Server)~~				ARes = ~~C~~See ACS Signed Content	For ~~the~~ ARes, see ACS Signed Content ~~required if Transaction = C~~

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Data Element/ Field Name	Description	Source	Length/Format/ Values	Device Channel	Message Category	Message Inclusion	Condition Inclusion
Serial Number Field Name: serialNum	If present in PReq, the DS returns card range data that has been updated since the time of the PRes. If absent, the DS returns all card ranges. If present in the PRes, indicates the current state of the card range data (the specific value is only meaningful to the DS). The 3DS Server should retain this value for submission in a future PReq to request only changes that have been made to the card range data since the PRes message was generated.	DS 3DS Server	Length: Variable, maximum 20 characters JSON Data Type: String	01-APP 02-BRW	01-PA 02-NPA	PReq = O PRes = O	
Transaction Status		DS	U = Authentication Could Not Be Performed; Technical or other problem, as indicated in ARes or ~~CRes~~RReq				For the CRes, ~~this data element~~ only present in the final CRes message. For 02–NPA optional as defined by DS
Transaction Status Reason		DS	Note: The EMVCo and DS reserved values are invalid until specifically defined.				

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.5.5 Error Code, Error Description, and Error Details

~~Note: Additional Error Code values may be defined at any time. All components must accept any value.~~

Table A.4 Error Code, Error Description, and Error Detail

Value	Error Code	Error Description	Error Detail
101	Message ~~element~~ received invalid ~~not a defined message~~	Message not recognized	Invalid Message OR Invalid Message for the receiving component OR Invalid Format Message ~~Invalid Message Type~~
204			~~Note: For a JSON parser that does not catch a duplicate data element error situation, this value is used to communicate the error.~~
302		Data could not be decrypted by the ~~DS~~ receiving system due to technical or other reason	
307	Serial Number not valid	Invalid Serial number in the PReq/PRes message (too old, not found)	None Required.
405	System Connection Failure	For example, the sending component is unable to establish connection to the receiving component.	Description of the failure

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.5.7 Card Range Data

Table A.6 Card Range Data

Data Element/Field Name	Description	Length/Format/Values	Inclusion
Action Indicator Field Name: actionInd	Indicates the action to take with the card range	Length: 1 character JSON Data Type: String	O
Start Protocol Version Field Name: startProtocolVersion	The earliest (i.e. oldest) protocol version that is supported by the ACS. Note: The Start Protocol Version shall be 2.0.1.	Length: 5 characters JSON Data Type: String Value: n.n.n "n" represents numeric digits that relate to the major and minor digits of the Message Version Number.	R
End Protocol Version Field Name: ~~protocolVersion~~endProtocolVersion	The most recent protocol version that is valid for the URL of the ACS ~~that will be used by the 3DS Method~~.		R

A.6 Message Extension Data

Table A.7 Message Extension Attributes

Attribute Name	Description	Length/Format	Inclusion
criticalityIndicator		~~Length: 4–5 characters~~	R
data		Length: Variable, Maximum ~~8192~~8059 characters	R

A.7 3DS Requestor Risk Information

- **3DS Requestor Prior Transaction Authentication**—How the 3DS Requestor previously used 3DS to authenticate the cardholder.

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.1 Cardholder Account Information

Table A.8 Cardholder Account Information

Data Element / Field Name	Description	Length/Format/Values
Cardholder Account Age Indicator		Values accepted: 1~~0~~ = No account (guest check-out) 2~~1~~ = Created during this transaction 3~~2~~ = Less than 30 days 4~~3~~ = 30–60 days 5~~4~~ = More than 60 days
Cardholder Account Password Change Indicator		1~~0~~ = No change 2~~1~~ = Changed during this transaction 3~~2~~ = Less than 30 days 4~~3~~ = 30–60 days 5~~4~~ = More than 60 days
Suspicious Account Activity		1~~0~~ = No suspicious activity has been observed 2~~1~~ = Suspicious activity has been observed
Shipping Name Indicator		1~~0~~ = Account Name identical to shipping Name 2~~1~~ = Account Name different than shipping Name
Payment Account Age Indicator		1~~0~~ = No account (guest check-out) 2~~1~~ = During this transaction 3~~2~~ = Less than 30 days 4~~3~~ = 30–60 days 5~~4~~ = More than 60 days

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.2 Merchant Risk Indicator

Table A.9 Merchant Risk Indicator

Data Element / Field Name	Description	Length/Format/Values
Reorder Items Indicator		1~~0~~ = First time ordered 2~~1~~ = Reordered
Pre-Order Purchase Indicator		1~~0~~ = Merchandise available 2~~1~~ = Future availability

A.7.4 3DS Requestor Prior Transaction Authentication Information

Table A.11 3DS Requestor Prior Transaction Authentication Information

Data Element / Field Name	Description	Length/Format/Values
3DS Requestor Prior Transaction Reference		This data element contains a ACS Transaction ID for a prior authenticated transaction (for example, the first recurring transaction that was authenticated with the cardholder)

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.5 ACS Rendering Type

The ACS Rendering Type contains information about the rendering type that the ACS is sending for the cardholder authentication. The detailed data elements are outlined in Table A.12.

Table A.12: ACS Rendering Type

Data Element / Field Name	Description	Length / Format / Values
Interface Field Name: interface	The is the interface that the challenge will be presented to the cardholder	Length: 2 characters JSON Data Type: String Values accepted: 01 = Native UI 02 = HTML UI
UI Type Field Name: uiType	This is the UI Type that the ACS will be presenting.	Length: 2 characters JSON Data Type: String Values accepted: 01 = Text 02 = Single Select 03= Multi Select 04 = OOB 05 = HTML Other

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.6 Device Rendering Options Supported

The Device Rendering Options Supported contains information about the rendering types and interface that the device is able to handle. The detailed data elements outlined in Table A.13.

Table A.13: Device Rendering Options Supported

Data Element / Field Name	Description	Length / Format / Values
Interface Field Name: interface	This will list all of the Interface types that the device supports for displaying specific challenge user interfaces within the SDK.	Length: 1 character JSON Data Type: String Values accepted: 1 = Native 2 = HTML 3 = Both
UI Type Field Name: uiType	This will list all of the UI types that the device supports for displaying specific challenge user interfaces within the SDK.	Length: 1 character JSON Data Type: String Values accepted: 1 = Text 2 = Single Select 3 = Multi Select 4 = OOB 5 = HTML Other

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

A.7.7 Issuer Image

The Issuer Image is supplied by the ACS to be displayed during the challenge message exchange. The detailed data elements in Table A.14.

Table A.14: Issuer Image

Data Element / Field Name	Description	Length / Format / Values

Medium Density Image
Field Name: medium
High Denisity Image
Field Name: high
Extra High Density Image
Field Name: extraHigh
No Issuer Image
Field name: none
	Include up to three fully qualified URLs defined as either; medium density, high density and extra high density images of the Issuer Image.
	Examples:
	Option 1 – Images to display:
	{
	“issuerImage” =
	{“medium”: “http://acs.com/medium_image.svg”,
	”high”: “http://acs.com/high_image.svg”,
	“extraHigh”: http://acs.com/extraHigh_image.svg” }
	}
	Option 1 – No Image to display:
	{“issuerImage” = “none”}
		Length: Variable, maximum 2048 characters
		JSON Data Type: String
		Values accepted:
		Option 1 – Images to display:
		Fully qualified URL in correct JSON Object format
		Option 1 – No Image to display:
		The value of “none” should be passed

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

Annex B Message Format

Updated to reflect new, modified, and deleted data elements included in Annex A updates.

Annex C Generate ECC Key Pair

A number of available cryptographic libraries include elliptic curve key pair generation as a function.

If not available, it is recommended to use a method ~~The following method~~ of elliptic curve key pair generation followings [ISO/IEC 15946-1].

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.

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

© 2017 EMVCo, LLC. All rights reserved. Reproduction, distribution and other use of this document is permitted only pursuant to the applicable agreement between the user and EMVCo found at www.emvco.com. EMV® is a registered trademark or trademark of EMVCo, LLC in the United States and other countries.