3DS payments - Customizable authentication and authorization calls

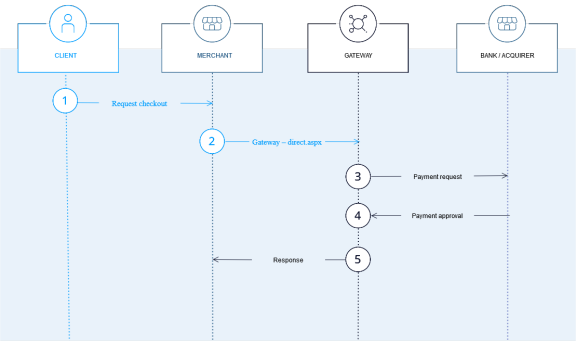
> The Server-2-Server payment is for PCI DSS compliant merchant.

To be able to create a Server-2-Server payment, the merchant have to create and host his own page.

The PCI DSS certification is mandatory for payments with PAN (first payments) not for payments with PCNr (used in one-click for example).

Chart of process flow via Server-to-Server

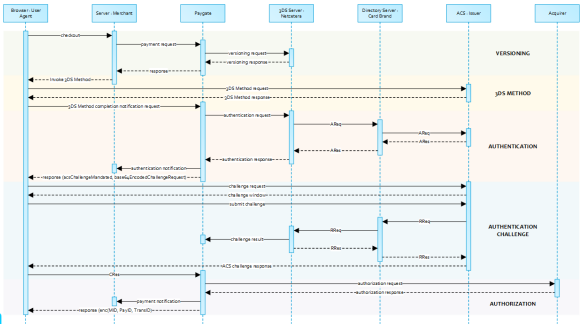
For the server-to-server payment processes please refer to the programming basics manual.



Overview

A 3-D Secure 2.0 payment sequence may comprise the following distinct activities:

- Versioning
  - Request ACS and DS Protocol Version(s) that correspond to card account range as well as an optional 3-D Secure Method URL
- 3-D Secure Method
  - Connect the cardholder browser to the issuer ACS to obtain additional browser data
- Authentication
  - Submit authentication request to the issuer ACS
- Challenge
  - Challenge the carholder if mandated
- Authorization
  - Authorize the authenticated transaction with the acquirer



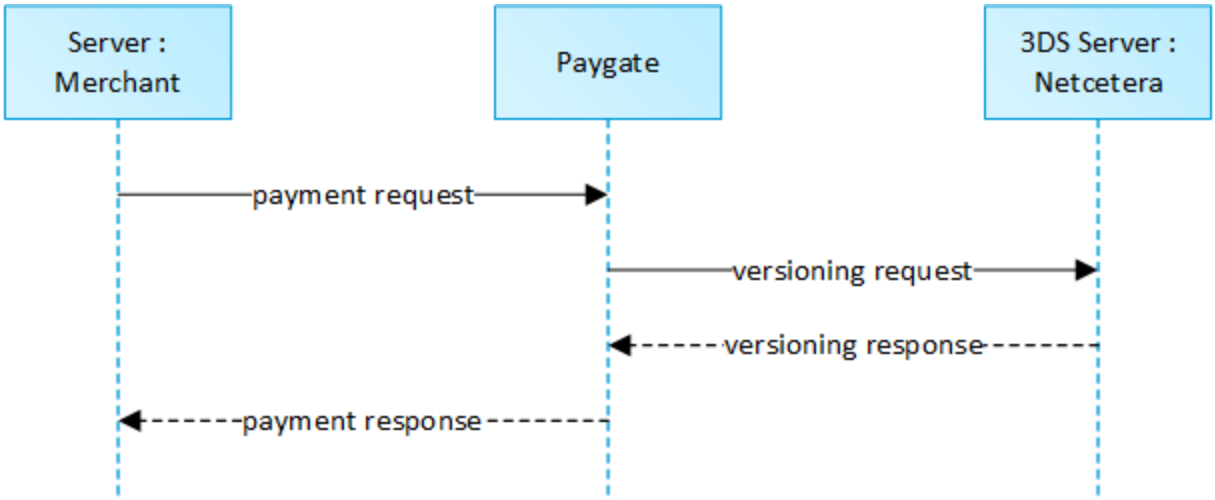
Server-2-Server Sequence Diagram

Please note that the the communication between client and Access Control Server (ACS) is implemented through iframes. Thus, responses arrive in an HTML subdocument and you may establish correspondent event listeners in your root document.

Alternatively you could solely rely on asynchronous notifications delivered to your backend. In those cases you may have to consider methods such as long polling, SSE or websockets to update the client.

Payment initiation

The initial request to will be the same regardless of the underlying 3-D Secure Protocol.



Request Elements

In order to start a server-to-server 3-D Secure card payment sequence please post the following key-value-pairs to

https://paymentpage.axepta.bnpparibas/direct.aspx

Notice: For security reasons, Axepta Platform rejects all payment requests with formatting errors. Therefore, please use the correct data type for each parameter.

Notice: In case of a merchant initiated recurring transaction the JSON objects (besides credentialOnFile and card), the URLNotify and TermURL are not mandatory parameters, because no 3D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response.

Key	Format	CND	Description
MerchantID	ans..30	M	MerchantID, assigned by . Additionally this parameter has to be passed in plain language too.

<table>
<tr><td>MsgVer</td><td>ans..5</td><td>M</td><td>Message version.

Values accepted:
<code>2.0</code>
<table><tr><th>Value</th><th>Description</th></tr><tr><td>2.0</td><td>With 3-D Secure 2.x a lot of additional data were required (e.g. browser-information, billing/shipping-address, account-info, ...) to improve authentication processing. To handle these information the JSON-objects have been put in place to handle such data. To indicate that these data are used the MsgVer has been implemented.</td></tr></table></td></tr>
<tr><td>TransID</td><td>ans..64</td><td>M</td><td>TransactionID provided by you which should be unique for each payment</td></tr>
<tr><td>RefNr</td><td>an..12</td><td>M</td><td>Merchant’s unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file (CTSF) we cannot add the additional payment data.

Notes:
Fixed length of 12 characters (only characters (A..Z, a..z) and digits (0..9) are allowed, no special characters like whitespace, underscore...)
If the number of characters entered is lower than 12, BNP will complete, starting from the left side, with "0" (Example : 000018279568)</td></tr>
<tr><td>scheme ReferenceID</td><td>ans..64</td><td>C</td><td>Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmssions.

Mandatory: CredentialOnFile – initial false – unschedule MIT / recurring</td></tr>
<tr><td>Amount</td><td>n..10</td><td>M</td><td>Amount in the smallest currency unit (e.g. EUR Cent). Please contact the , if you want to capture amounts <100 (smallest currency unit).</td></tr>
<tr><td>Currency</td><td>a3</td><td>M</td><td>Currency, three digits DIN / ISO 4217, e.g. EUR, USD, GBP. Please find an overview here: Currency table</td></tr>
<tr><td>card</td><td>JSON</td><td>M</td><td>Card data</td></tr>
<tr><td>Capture</td><td>an..6</td><td>O</td><td>Determines the type and time of capture.<table><tr><th>Capture Mode</th><th>Description</th></tr><tr><td>AUTO</td><td>Capturing immediately after authorisation (default value).</td></tr><tr><td>MANUAL</td><td>Capturing made by the merchant. Capture is normally initiated at time of delivery.</td></tr><tr><td><Number></td><td>Delay in hours until the capture (whole number; 1 to 696).</td></tr></table></td></tr>
<tr><td>MAC</td><td>an64</td><td>M</td><td>Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here:
HMAC Authentication (Request)
HMAC Authentication (Notify)</td></tr>
<tr><td>channel</td><td>a..20</td><td>C</td><td>Indicates the type of channel interface being used to initiate the transaction.

Values accepted:
<code>Browser</code>
<code>App</code>
<code>3RI</code>
If not present the value <code>Browser</code> is implied.</td></tr>
<tr><td>billingDescriptor</td><td>ans..22</td><td>O</td><td>A descriptor to be printed on a cardholder’s statement. Please also refer to the additional comments made elswhere for more information about rules and regulations.</td></tr>
<tr><td>OrderDesc</td><td>ans..768</td><td>O</td><td>Order description</td></tr>
<tr><td>TermURL</td><td>ans..256</td><td>M</td><td>In case of 3-D Secure 1.0 fallback: the URL the customer will be returned to at the end of the 3-D Secure 1.0 authentication process.</td></tr>
<tr><td>AccVerify</td><td>a3</td><td>O</td><td>Indicator to request an account verification (aka zero value authorization). If an account verification is requested the submitted amount will be optional and ignored for the actual payment transaction (e.g. authorization).

Values accepted:
<code>Yes</code></td></tr>
<tr><td>threeDS Policy</td><td>JSON</td><td>O</td><td>Object specifying authentication policies and exemption handling strategies</td></tr>
<tr><td>threeDS Data</td><td>JSON</td><td>C</td><td>Object detailing authentication data in case authentication was performed through a third party or by the merchant</td></tr>
</table>

priorAuthenticationInfo	JSON	O	Prior Transaction Authentication Information contains optional information about a 3-D Secure cardholder authentication that occurred prior to the current transaction
browserInfo	JSON	M	Accurate browser information are needed to deliver an optimized user experience. Required for 3-D Secure 2.0 transactions.
accountInfo	JSON	O	The account information contains optional information about the customer account with the merchant. Optional for 3-D Secure 2.0 transactions.
billToCustomer	JSON	C	The customer that is getting billed for the goods and / or services. Required unless market or regional mandate restricts sending this information.
shipToCustomer	JSON	C	The customer that the goods and / or services are sent to. Required (if available and different from billToCustomer) unless market or regional mandate restricts sending this information.
billingAddress	JSON	C	Billing address. Required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.
shippingAddress	JSON	C	Shipping address. If different from billingAddress, required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.
credentialOnFile	JSON	C	Object specifying type and series of transactions using payment account credentials (e.g. account number or payment token) that is stored by a merchant to process future purchases for a customer. Required if applicable.
merchantRiskIndicator	JSON	O	The Merchant Risk Indicator contains optional information about the specific purchase by the customer
URLNotify	an..256	M	Complete URL which Platform calls up in order to notify the shop about the payment result. The URL may be called up only via port 443. It may not contain parameters: Use the UserData parameter instead. **Common notes:** • We recommend to use parameter "response=encrypted" to get an encrypted response by Platform • However, fraudster may just copy the encrypted DATA-element which are sent to URLFailure and send the DATA to URLSuccess/URLNotify. Therefore ensure to check the "code"-value which indicates success/failure of the action. Only a result of "code=00000000" should be considered successful.
UserData	ans..1024	O	If specified at request, forwards the parameter with the payment result to the shop.

Key	Format	CND	Description	Beschreibung
MsgVer	ans..5	M	Message version. Values accepted: • **2.0** **Value** / **Description** 2.0 — With 3-D Secure 2.x a lot of additional data were required (e.g. browser-information, billing /shipping-address, account-info, ...) to improve authentication processing. To handle these information the JSON-objects have been put in place to handle such data. To indicate that these data are used the MsgVer has been implemented.	Message-Version. Zulässige Werte: • **2.0**

Key	Format	CND	Description
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Key	Format	CND	Description	Beschreibung
RefNr	an..12	M	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file (CTSF) we cannot add the additional payment data. Notes: • **Fixed length** of 12 characters (only characters (A..Z, a..z) and digits (0..9) are allowed, **no special characters** like whitespace, underscore...) • If the number of characters entered is lower than 12, BNP will complete, starting from the left side, with "0" (Example : 000018279568)	Eindeutige Referenznummer des Händlers, welche als Auszahlungsreferenz in der entsprechenden Acquirer EPA-Datei angegeben wird. Bitte beachten Sie, ohne die Übergabe einer eigenen Auszahlungsreferenz können Sie die EPA-Transaktionen nicht zuordnen, zusätzlich kann das **The page DE:Wording was not found** -- Please check /update the page name used in the MultiExcerpt-Include macro Settlement File (CTSF) auch nicht zusätzlich angereichert werden.

schemeReferenceID	ans..64	C	Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmssions. Mandatory: CredentialOnFile – initial false – unschedule MIT / recurring	Kartensystemspezifische Transaktions-ID, die für nachfolgende Zahlungen mit hinterlegten Daten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist. Pflicht: CredentialOnFile – initial false – unschedule MIT / recurring

Key	Format	CND	Description
Amount	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the , if you want to capture amounts <100 (smallest currency unit).

Key	Format	CND	Description
Currency	a3	M	Currency, three digits DIN / ISO 4217, e.g. EUR, USD, GBP. Please find an overview here: Currency table

Key	Format	CND	Description	Beschreibung
card	JSON	M	Card data	Kartendaten

Key	Format	CND	Description
Capture	an..6	O	Determines the type and time of capture. **Capture Mode** / **Description** **AUTO** – Capturing immediately after authorisation (default value). **MANUAL** – Capturing made by the merchant. Capture is normally initiated at time of delivery. **<Number>** – Delay in hours until the capture (whole number; 1 to 696).

Key	Format	CND	Description	Beschreibung
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)	
channel	a..20	C	Indicates the type of channel interface being used to initiate the transaction. Values accepted: • **`Browser`** • **`App`** • **`3RI`** If not present the value **`Browser`** is implied.	Gibt die Art der verwendeten Schnittstelle zur Initiierung der Transaktion an. Zulässige Werte: • **`Browser`** • **`App`** • **`3RI`** Wenn nicht angegeben, wird der Wert **`Browser`** verwendet.
billingDescriptor	ans..22	O	A descriptor to be printed on a cardholder's statement. Please also refer to the additional comments made elswhere for more information about rules and regulations.	Ein auf dem Kontoauszug des Karteninhabers zu druckender Beschreiber. Beachten Sie bitte auch die andernorts gemachten zusätzlichen Hinweise für weitere Informationen über Regeln und Vorschriften.
OrderDesc	ans..768	O	Order description	Beschreibung der Bestellung
TermURL	ans..256	M	In case of 3-D Secure 1.0 fallback: the URL the customer will be returned to at the end of the 3-D Secure 1.0 authentication process.	
AccVerify	a3	O	Indicator to request an account verification (aka zero value authorization). If an account verification is requested the submitted amount will be optional and ignored for the actual payment transaction (e.g. authorization). Values accepted: • **`Yes`**	
threeDSPolicy	JSON	O	Object specifying authentication policies and exemption handling strategies	
threeDSData	JSON	C	Object detailing authentication data in case authentication was performed through a third party or by the merchant	

priorAuthenticationInfo	JSON	O	Prior Transaction Authentication Information contains optional information about a 3-D Secure cardholder authentication that occurred prior to the current transaction	
browserInfo	JSON	M	Accurate browser information are needed to deliver an optimized user experience. Required for 3-D Secure 2.0 transactions.	Exakte Browserinformationen sind nötig, um eine optimierte Nutzererfahrung zu liefern. Erforderlich für 3-D Secure 2.0 Transaktionen.
accountInfo	JSON	O	The account information contains optional information about the customer account with the merchant. Optional for 3-D Secure 2.0 transactions.	
billToCustomer	JSON	C	The customer that is getting billed for the goods and / or services. Required unless market or regional mandate restricts sending this information.	
shipToCustomer	JSON	C	The customer that the goods and / or services are sent to. Required (if available and different from billToCustomer) unless market or regional mandate restricts sending this information.	
billingAddress	JSON	C	Billing address. Required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.	
shippingAddress	JSON	C	Shipping address. If different from billingAddress, required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.	
credentialOnFile	JSON	C	Object specifying type and series of transactions using payment account credentials (e.g. account number or payment token) that is stored by a merchant to process future purchases for a customer. Required if applicable.	
merchantRiskIndicator	JSON	O	The Merchant Risk Indicator contains optional information about the specific purchase by the customer	
URLNotify	an..256	M	Complete URL which Platform calls up in order to notify the shop about the payment result. The URL may be called up only via port 443. It may not contain parameters: Use the UserData parameter instead. **Common notes:** • We recommend to use parameter "response=encrypted" to get an encrypted response by Platform • However, fraudster may just copy the encrypted DATA-element which are sent to URLFailure and send the DATA to URLSuccess/URLNotify. Therefore ensure to check the "code"-value which indicates success /failure of the action. Only a result of "code=00000000" should be considered successful.	Die Händler-URL, die asynchrone Anfragen während des Authentisierungsprozesses empfängt

Key	Format	CND	Description
UserData	ans..1024	O	If specified at request, forwards the parameter with the payment result to the shop.

General parameters for credit card payments via socket connection

Please note the additional parameter for a specific credit card integration in the section "Specific parameters"

Response Elements

The following table describes the result parameters with which the Axepta Platform responds to your system

pls. be prepared to receive additional parameters at any time and do not check the order of parameters

the key (e.g. MerchantId, RefNr) should not be checked case-sentive

Key	Format	CND	Description
MID	ans..30	M	MerchantID, assigned by
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by

TransID	ans..64	M	TransactionID provided by you which should be unique for each payment
Status	a..20	M	Status of the transaction. Values accepted: • **AUTHENTICATION_REQUEST** • **PENDING** • **FAILED**
RefNr	an12	M	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file we cannot add the additional payment data. Notes: • **Fixed length** of 12 characters (only characters (A..Z, a..z) and digits (0..9) are allowed, **no special characters** like whitespace, underscore...) • **For AMEX** : RefNr is **mandatory** • If the number of characters entered is lower than 12, BNP will complete, starting from the left side, with "0" (Example : 000018279568)
Description	ans..1024	M	Further details in the event that payment is rejected. Please **do not** use the **Description but** the Code parameter for the transaction status analysis!
Code	n8	M	Error code according to Response Codes (A4 Response codes)
UserData	ans..1024	O	If specified at request, forwards the parameter with the payment result to the shop.
versioningData	JSON	M	The Card Range Data data element contains information that indicates the most recent EMV 3-D Secure version supported by the ACS that hosts that card range. It also may optionally contain the ACS URL for the 3-D Secure Method if supported by the ACS and the DS Start and End Protocol Versions which support the card range.
threeDSLegacy	JSON	M	Object containing the data elements required to construct the Payer Authentication request in case of a fallback to 3-D Secure 1.0.
schemeReferenceID	ans..64	C	Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmssions.
card	JSON	M	Card data
ipInfo	JSON	O	Object containing IP information
threeDSData	JSON	M	Authentication data
resultsResponse	JSON	C	In case the authentication process included a cardholder challenge additional information about the challenge result will be provided.

Key	Format	CND	Description
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.

Key	Format	CND	Description
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by

Key	Format	CND	Description
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Key	Format	CND	Description	Beschreibung
Status	a..20	M	Status of the transaction. Values accepted: • **AUTHENTICATION_REQUEST** • **PENDING** • **FAILED**	Status der Transaktion. Zulässige Werte: • **AUTHENTICATION_REQUEST** • **PENDING** • **FAILED**

RefNr	an12	M	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file we cannot add the additional payment data. Notes: • **Fixed length** of 12 characters (only characters (A..Z, a..z) and digits (0..9) are allowed, **no special characters** like whitespace, underscore...) • **For AMEX** : RefNr is **mandatory** • If the number of characters entered is lower than 12, BNP will complete, starting from the left side, with "0" (Example : 000018279568)	

Key	Format	CND	Description
Description	ans..1024	M	Further details in the event that payment is rejected. Please **do not** use the **Description but** the Code parameter for the transaction status analysis!

Key	Format	CND	Description
Code	n8	M	Error code according to Response Codes (A4 Response codes)

Key	Format	CND	Description
UserData	ans..1024	O	If specified at request, forwards the parameter with the payment result to the shop.

Key	Format	CND	Description	Beschreibung
versioningData	JSON	M	The Card Range Data data element contains information that indicates the most recent EMV 3-D Secure version supported by the ACS that hosts that card range. It also may optionally contain the ACS URL for the 3-D Secure Method if supported by the ACS and the DS Start and End Protocol Versions which support the card range.	Das Datenelement Card Range Data enthält Informationen, welche die jüngste vom ACS, der den Kartenbereich hostet, unterstützte EMV 3-D Secure-Version angeben. Es kann optional auch die ACS URL für die 3-D Secure Methode enthalten, falls vom ACS unterstützt, sowie die DS Start- und End-Protokoll-Versionen, die den Kartenbereich unterstützen.
threeDSLegacy	JSON	M	Object containing the data elements required to construct the Payer Authentication request in case of a fallback to 3-D Secure 1.0.	Objekt, dass die erforderlichen Datenelemente für die Konstruktion der Anfrage zur Zahler-Authentisierung im Falle eines Fallbacks auf 3-D Secure 1.0 enthält.
schemeReferenceID	ans..64	C	Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmssions.	
card	JSON	M	Card data	
ipInfo	JSON	O	Object containing IP information	
threeDSData	JSON	M	Authentication data	
resultsResponse	JSON	C	In case the authentication process included a cardholder challenge additional information about the challenge result will be provided.	

The **`versioningData`** object will indicate the EMV 3-D Secure protocol versions (i.e. 2.1.0 or higher) that are supported by Access Control Server of the issuer.

If the corresponding protocol version fields are NULL it means that the BIN range of card issuer is not registered for 3-D Secure 2.0 and a fallback to 3-D Secure 1.0 is required for transactions that are within the scope of PSD2 SCA.

When parsing **`versioningData`** please also refer to the subelement **`errorDetails`** which will specify the reason if some fields are not pupoluated (e.g. Invalid cardholder account number passed, not available card range data, failure in encoding/serialization of the 3-D Secure Method data etc).

versioningData

BASEURL=

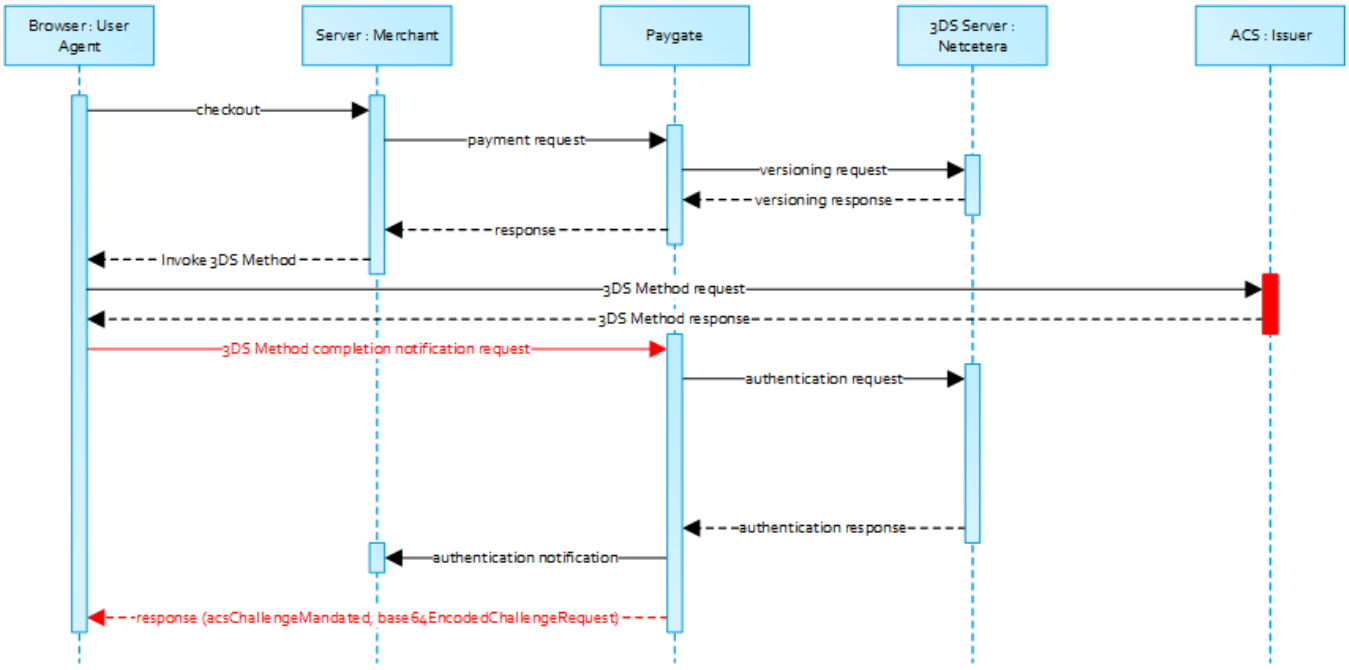
```
{
        "threeDSServerTransID": "14dd844c-b0fc-4dfe-8635-366fbf43468c",
        "acsStartProtocolVersion": "2.1.0",
        "acsEndProtocolVersion": "2.1.0",
        "dsStartProtocolVersion": "2.1.0",
        "dsEndProtocolVersion": "2.1.0",
        "threeDSMethodURL": "http://www.acs.com/script",
        "threeDSMethodDataForm":
"eyJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMIjoiaHR0cHM6Ly93d3cuY29tcHV0b3AtcGF5Z2F0ZS5jb20vY2JUaHJlZURTLmFzcHg_
YWN0aW9uPW10aGROdGZuIiwidGhyZWVEU1NlcnZlclRyYW5zSUQiOiIxNGRkODQ0Yy1iMGZjLTRkZmUtODYzNS0zNjZmYmY0MzQ2OGMifQ=="
,
        "threeDSMethodData": {
                "threeDSMethodNotificationURL": "BASEURLcbThreeDS.aspx?action=mthdNtfn",
                "threeDSServerTransID": "14dd844c-b0fc-4dfe-8635-366fbf43468c"
        }
}
```


3-D Secure Method

The 3-D Secure Method allows for additional browser information to be gathered by an ACS prior to receipt of the authentication request message (AReq) to help facilitate the transaction risk assessment. Support of 3-D Secure Method is optional and at the discretion of the issuer.

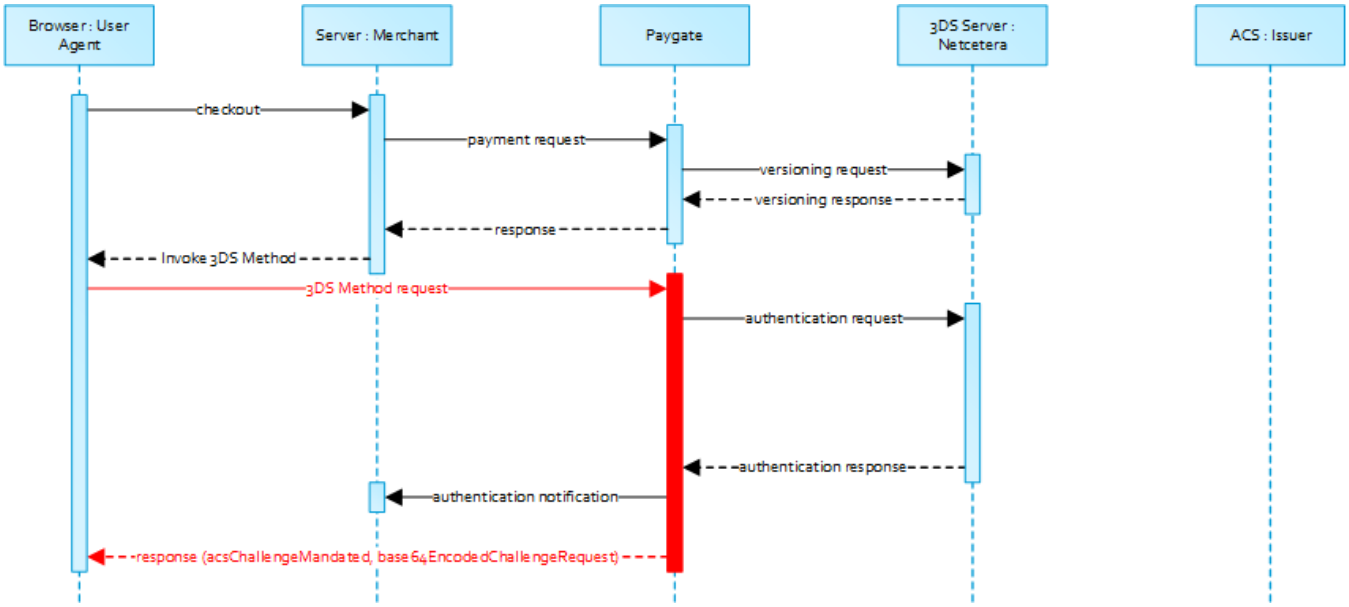
The **`versioningData`** object contains a value for **`threeDSMethodURL`** . The merchant is supposed to invoke the 3-D Secure Method via a hidden HTML iframe in the cardholder browser and send a form with a field named **`threeDSMethodData`** via HTTP POST to the ACS 3-D Secure Method URL.

3-D Secure Method: **`threeDSMethodURL`**



Please not that the **`threeDSMethodURL`** will be populated by if the issuer does not support the 3-D Secure Method. The 3-D Secure Method Form Post as outlined below must be performed independently from whether it is supported by the issuer. This is necessary to facilitate direct communication between the browser and in case of a mandated challenge or a frictionless flow.

3-D Secure Method: No issuer **`threeDSMethodURL`**



3-D Secure Method Form Post

```
<form name="frm" method="POST" action="Rendering URL">
    <input type="hidden" name="threeDSMethodData" value="
eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6IjNhYzdjYWE3LWFhNDItMjY2My03OTFiLTJhYzA1YTU0MmM0YSIsInRocmVlRFNNZXRob2ROb3RpZ
mljYXRpb25VUkwiOiJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMIn0">
</form>
```

The ACS will intercat with the Cardholder browser via the HTML iframe and then store the applicable values with the 3-D Secure Server Transaction ID for use when the subsequent authentication message is received containing the same 3-D Secure Server Transaction ID.

Netcetera 3DS Web SDK

You may use the operations **`init3DSMethod`** or **`createIframeAndInit3DSMethod`** at your discreation from the nca3DSWebSDK in order to iniatiate the 3-D Secure Method. Please refer to the Integration Manual at https://mpi.netcetera.com/3dsserver/doc/current/integration.html#Web_Service_API.

Once the 3-D Secure Method is concluded the ACS will instruct the the cardholder browser through the iFrame response document to submit **`threeDSMethodData`** as a hidden form field to the 3-D Secure Method Notification URL.

ACS Response Document

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8"/>
    <title>Identifying...</title>
</head>
<body>
<script>
    var tdsMethodNotificationValue =
'eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImUxYzFlYmViLTc0ZTgtNDNiMi1iMzg1LTJlNjdkMWFhY2ZhMiJ9';

    var form = document.createElement("form");
    form.setAttribute("method", "post");
    form.setAttribute("action", "notification URL");

    addParameter(form, "threeDSMethodData", tdsMethodNotificationValue);

    document.body.appendChild(form);
    form.submit();

    function addParameter(form, key, value) {
        var hiddenField = document.createElement("input");
        hiddenField.setAttribute("type", "hidden");
        hiddenField.setAttribute("name", key);
        hiddenField.setAttribute("value", value);
        form.appendChild(hiddenField);
    }
</script>
</body>
</html>
```

3-D Secure Method Notification Form

```
<form name="frm" method="POST" action="3DS Method Notification URL">
    <input type="hidden" name="threeDSMethodData" value="
eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImUxYzFlYmViLTc0ZTgtNDNiMi1iMzg1LTJlNjdkMWFhY2ZhMiJ9">
</form>
```

⚠ Please note that the **`threeDSMethodNotificationURL`** as embedded in the Base64 encoded **`threeDSMethodData`** value points to and must not be modified. The merchant notification is delivered to the URLNotify as provided in the original request or as configured for the MerchantID in .

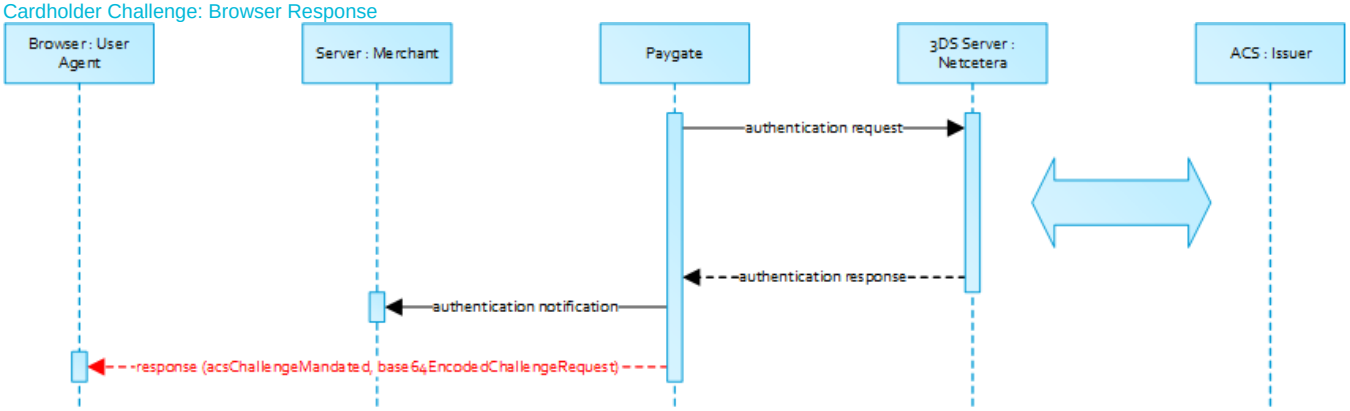
Authentication

If 3-D Secure Method is supported by the issuer ACS and was invoked by the merchant will automatically continue with the authentication request once the 3-D Secure Method has completed (i.e. 3-D Secure Method Notification).

The authentication result will be transferred via HTTP POST to the **`URLNotify`** . It may indicate that the Cardholder has been authenticated, or that further cardholder interaction (i.e. challenge) is required to complete the authentication.

In case a cardholder challenge is deemed necessary will transfer a JSON object within the body of HTTP browser response with the elements **`acsChallengeMandated`** , **`challengeRequest`** , **`base64EncodedChallengeRequest`** and **`acsURL`** . Otherwise, in a frictionless flow, will automatically continue and respond to the cardholder browser once the authorization completed.

Cardholder Challenge: Browser Response



Browser Challenge Response

Data Elements

Key	Format	CND	Description
acsChallengeMandated	boolean	M	Indication of whether a challenge is required for the transaction to be authorised due to local/regional mandates or other variable
challengeRequest	object	M	Challenge request object
base64EncodedChallengeRequest	string	M	Base64-encoded Challenge Request object
acsURL	string	M	Fully qualified URL of the ACS to be used to post the Challenge Request

Schema: Browser Challenge Response

```
{
        "$schema": "http://json-schema.org/draft-07/schema#",
        "type": "object",
        "properties": {
                "acsChallengeMandated": {"type": "boolean"},
                "challengeRequest": {"type": "object"},
                "base64EncodedChallengeRequest": {"type": "string"},
                "acsURL": {"type": "string"}
        },
        "required": ["acsChallengeMandated", "challengeRequest", "base64EncodedChallengeRequest", "acsURL"],
        "additionalProperties": false
}
```

Sample: Browser Challenge Response

```
{
        "acsChallengeMandated": true,
        "challengeRequest": {
                "threeDSServerTransID": "8a880dc0-d2d2-4067-bcb1-b08d1690b26e",
                "acsTransID": "d7c1ee99-9478-44a6-b1f2-391e29c6b340",
                "messageType": "CReq",
                "messageVersion": "2.1.0",
                "challengeWindowSize": "01",
                "messageExtension": [
                        {
                                "name": "emvcomsgextInChallenge",
                                "id": "tc8Qtm465Ln1FX0nZprA",
                                "criticalityIndicator": false,
                                "data": "messageExtensionDataInChallenge"
                        }
                ]
        },
        "base64EncodedChallengeRequest": "base64-encoded-challenge-request",
        "acsURL": "acsURL-to-post-challenge-request"
}
```


Authentication Notification

The data elements of the authentication notification are listed in the table below.

Key	Format	CND	Description
MID	ans..30	M	MerchantID, assigned by
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment
Code	n8	M	Error code according to Response Codes (A4 Response codes)
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)
authenticationResponse	JSON	M	Response object in return of the authentication request with the ACS

Key	Format	CND	Description
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.

Key	Format	CND	Description
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Key	Format	CND	Description
Code	n8	M	Error code according to Response Codes (A4 Response codes)

Key	Format	CND	Description
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)

Key	Format	CND	Description	Beschreibung

authenticationResponse	JSON	M	Response object in return of the authentication request with the ACS	Antwort-Objekt als Rückgabe zur Authentisierungs-Anfrage beim ACS

Browser Challenge

If a challenge is deemed necessary (see challengeRequest) the browser challenge will occur within the cardholder browser. To create a challenge it is required to post the value **`base64EncodedChallengeRequest`** via an HTML iframe to the ACS URL.

Challenge Request

```
<form name="challengeRequestForm" method="post" action="acsChallengeURL">
        <input type="hidden" name="creq" value="
ewogICAgInRocmVlRFNTZXJ2ZXJUcmFuc0lEIjogIjhhODgwZGMwLWQyZDItNDA2Ny1iY2IxLWIwOGQxNjkwYjI2ZSIsCiAgICAiYWNzVHJhb
nNJRCI6ICJkN2MxZWU5OS05NDc4LTQ0YTYtYjFmMi0zOTFlMjljNmIzNDAiLAogICAgIm1lc3NhZ2VUeXBlIjogIkNSZXEiLAogICAgIm1lc3
NhZ2VWZXJzaW9uIjogIjIuMS4wIiwKICAgICJjaGFsbGVuZ2VXaW5kb3dTaXplIjogIjAxIiwKICAgICJtZXNzYWdlRXh0ZW5zaW9uIjogWwo
JCXsKCQkJIm5hbWUiOiAiZW12Y29tc2dleHRJbkNoYWxsZW5nZSIsCgkJCSJpZCI6ICJ0YzhRdG00NjVMbjFGWDBuWnByQSIsCgkJCSJjcml0
aWNhbGl0eUluZGljYXRvciI6IGZhbHNlLAoJCQkiZGF0YSI6ICJtZXNzYWdlRXh0ZW5zaW9uRGF0YUluQ2hhbGxlbmdlIgoJCX0KICAgIF0Kf
Q==">
</form>
```

You may use the operations **`init3DSChallengeRequest`** or **`createIFrameAndInit3DSChallengeRequest`** from the nca3DSWebSDK in order submit the challenge message through the cardholder browser.

Init 3-D Secure Challenge Request - Example

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <script src="nca-3ds-web-sdk.js" type="text/javascript"></script>
    <title>Init 3-D Secure Challenge Request - Example</title>
</head>
<body>
<!-- This example will show how to initiate Challenge Reqeuests for different window sizes. -->
<div id="frameContainer01"></div>
<div id="frameContainer02"></div>
<div id="frameContainer03"></div>
<div id="frameContainer04"></div>
<div id="frameContainer05"></div>
<iframe id="iframeContainerFull" name="iframeContainerFull" width="100%" height="100%"></iframe>

<script type="text/javascript">
    // Load all containers
    iFrameContainerFull = document.getElementById('iframeContainerFull');
    container01 = document.getElementById('frameContainer01');
    container02 = document.getElementById('frameContainer02');
    container03 = document.getElementById('frameContainer03');
    container04 = document.getElementById('frameContainer04');
    container05 = document.getElementById('frameContainer05');


    // nca3DSWebSDK.init3DSChallengeRequest(acsUrl, creqData, container);
    nca3DSWebSDK.init3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-request',
iFrameContainerFull);

    // nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest(acsUrl, creqData, challengeWindowSize, frameName,
rootContainer, callbackWhenLoaded);
    nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-
request', '01', 'threeDSCReq01', container01);
    nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-
request', '02', 'threeDSCReq02', container02);
    nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-
request', '03', 'threeDSCReq03', container03);
    nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-
request', '04', 'threeDSCReq04', container04);
    nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest('http://example.com', 'base64-encoded-challenge-
request', '05', 'threeDSCReq05', container05, () => {
        console.log('Iframe loaded, form created and submitted');
    });
</script>

</body>
</html>
```

Once the cardholder challenge is completed, was cancelled or timed out the ACS will instruct the browser to post the results to the notfication URL as specified in the challenge request and to send a Result Request (RReq) via the Directory Server to the 3-D Secure Server.

⚠ Please note that the notification URL submited in the challenge request points to and must not be changed.

Authorization

After succefull cardholder authentication or proof of attempted authentication/verification is provided will automatically continue with the payment authorization.

In case the cardholder authentication was not succesfull or proof proof of attempted authentication/verification can not be provided will not continue with an authorization request.

In both cases will deliver a final notification to the merchant specified **`URLNotify`** with the data elements as listed in the table below.

Payment Notification

Key	Format	CND	Description
MID	ans..30	M	MerchantID, assigned by
MsgVer	ans..5	M	Message version. Accepted values: • **`2.0`** **Value** / **Description** 2.0 — With 3-D Secure 2.x a lot of additional data were required (e.g. browser-information, billing/shipping-address, account-info, ...) to improve authentication processing. To handle these information the JSON-objects have been put in place to handle such data. To indicate that these data are used the MsgVer has been implemented.
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment
schemeReferenceID	ans..64	C	Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmissions. Mandatory: CredentialOnFile – initial false – unscheduled MIT / recurring
TrxTime	an21	M	Transaction time stamp in format DD.MM.YYYY HH:mm:ssff
Status	a..20	M	Status of the transaction. Values accepted: • **`Authorized`** • **`OK`** (Sale) • **`PENDING`** • **`FAILED`** In case of **Authentication-only** the **Status** will be either **`OK`** or **`FAILED`** .
Description	ans..1024	M	Further details in the event that payment is rejected. Please **do not** use the **Description but** the Code parameter for the transaction status analysis!
Code	n8	M	Error code according to Response Codes (A4 Response codes)
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)
card	JSON	M	Card data
ipInfo	JSON	O	Object containing IP information
threeDSData	JSON	M	Authentication data
resultsResponse	JSON	C	In case the authentication process included a cardholder challenge additional information about the challenge result will be provided.

Key	Format	CND	Description	Beschreibung

MsgVer	ans..5	M	Message version. Accepted values: • **`2.0`** **Value** / **Description** 2.0 – With 3-D Secure 2.x a lot of additional data were required (e.g. browser-information, billing /shipping-address, account-info, ...) to improve authentication processing. To handle these information the JSON-objects have been put in place to handle such data. To indicate that these data are used the MsgVer has been implemented.	Message-Version. Zulässige Werte: • **`2.0`**

Key	Format	CND	Description
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.

Key	Format	CND	Description
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by

Key	Format	CND	Description
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Key	Format	CND	Description	Beschreibung
schemeReferenceID	ans..64	C	Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmissions. Mandatory: CredentialOnFile – initial false – unscheduled MIT / recurring	Kartensystemspezifische Transaktions-ID, die für nachfolgende Zahlungen mit hinterlegten Daten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist
TrxTime	an21	M	Transaction time stamp in format DD.MM.YYYY HH:mm:ssff	Zeitstempel der Transaktion im Format DD.MM.YYYY HH:mm:ssff
Status	a..20	M	Status of the transaction. Values accepted: • **`Authorized`** • **`OK`** (Sale) • **`PENDING`** • **`FAILED`** In case of **Authentication-only** the **Status** will be either **`OK`** or **`FAILED`** .	Status der Transaktion. Zulässige Werte: • **`Authorized`** • **`OK`** (Sale) • **`PENDING`** • **`FAILED`** Im Falle von **nur Authentisierung** ist der **Status** entweder **`OK`** oder **`FAILED`**.

Key	Format	CND	Description
Description	ans..1024	M	Further details in the event that payment is rejected. Please **do not** use the **Description but** the Code parameter for the transaction status analysis!

Key	Format	CND	Description
Code	n8	M	Error code according to Response Codes (A4 Response codes)

Key	Format	CND	Description
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)

Key	Format	CND	Description	Beschreibung
card	JSON	M	Card data	Kartendaten

ipInfo	JSON	O	Object containing IP information	Objekt mit IP-Informationen
threeDSData	JSON	M	Authentication data	Authentisierungsdaten
resultsResponse	JSON	C	In case the authentication process included a cardholder challenge additional information about the challenge result will be provided.	Falls der Authentisierungsprozess eine Challenge des Karteninhabers enthalten hat, werden zusätzliche Informationen über das Ergebnis der Challenge bereitgestellt

Browser Payment Response

Additionally the JSON formated data elements as listed below are trasferred in the HTTP response body to the cardholder browser. Please note that the data elements (i.e. **`MID`** , **`Len`** , **`Data`**) are base64 encoded.

Data Elements

Key	Format	CND	Description
MID	ans..30	M	MerchantID, assigned by
Len	integer	M	Length of the unencrypted **`Data`** string
Data	string	M	Blowfish encrypted string containg a JSON object with **`MID`** , **`PayID`** and **`TransID`**

Key	Format	CND	Description	Beschreibung
Len	integer	M	Length of the unencrypted **`Data`** string	Länge des unverschlüsselten Strings **`Data`**
Data	string	M	Blowfish encrypted string containg a JSON object with **`MID`** , **`PayID`** and **`TransID`**	Blowfish-verschlüsselter String, der ein JSON-Objekt mit **`MID`**, **`PayID`** und **`TransID`** enthält

Schema

```
{
        "$schema": "http://json-schema.org/draft-07/schema#",
        "type": "object",
        "properties": {
                "MID": {
                        "type": "string"
                },
                "Len": {
                        "type": "integer"
                },
                "Data": {
                        "type": "string"
                }
        },
        "required": ["MID", "Len", "Data"],
        "additionalProperties": false
}
```

Merchants are supposed to forward these data elements to their server for decryption and mapping agianst the payment notification. Based on the payment results the merchant server may deliver an appropriate response to the cardholder browser (e.g. success page).

Decrypted `Data`

Key	Format	CND	Description
MID	ans..30	M	MerchantID, assigned by
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Key	Format	CND	Description
PayID	an32	M	ID assigned by for the payment, e.g. for referencing in batch files as well as for capture or credit request.

Key	Format	CND	Description
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment

Sample decrypted `Data`

```
MID=YourMID&PayID=PayIDassignedbyPlatform&TransID=YourTransID
```