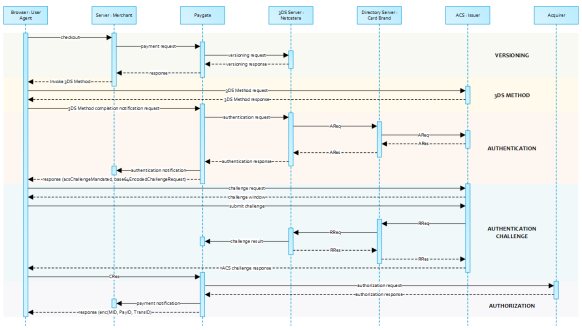
Server-2-Server Integration

Card processing - Server-2-Server integration

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 card holder 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.

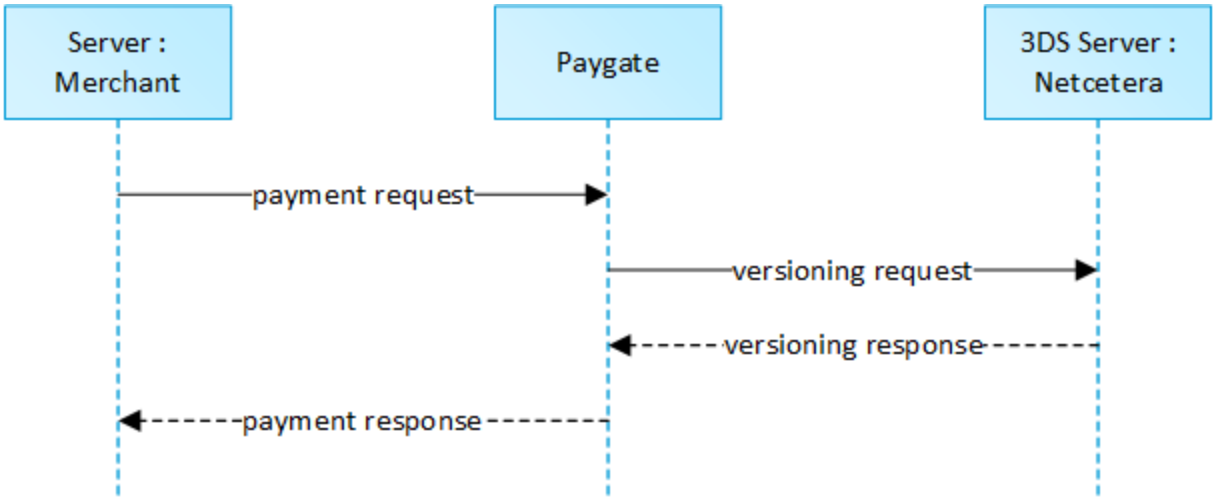
Alternatively you could solely rely on asynchronous notifications delivered to your backend. In those cases you may have to consider methods such as long polling, SSE or websockets to update the client.

EMV 3-D Secure

API Playground

Payment Initiation

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



In order to start a server-to-server 3-D Secure card payment sequence please post the following key-value-pairs to https://www.computop-paygate.com/ direct.aspx.

Call of interface: general parameters

Notice: For credit card payments with 3-D Secure, please note the different cases as explained separately in the chapter at the start of the handbook. If the credit card is registered for Verified or SecureCode or SafeKey, the next phase is divided into two steps of authentication and payment. However it always begins in the same way via the **direct.aspx** interface. The first response however is the receipt of Javascript code or other parameters in order to carry out a second call up of the **direct3d.aspx** interface. Only after that, do you receive the listed parameter as a response.

To carry out a credit card payment via a Server-to-Server connection, please use the following URL:

https://www.computop-paygate.com/direct.aspx

Request Elements

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

The following table describes the encrypted payment request parameters:

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 3-D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response.

Key	REST	Format	CND	Description
MerchantID	BasicAuth.Username	ans..30	M	MerchantID, assigned by Computop. Additionally this parameter has to be passed in plain language too.
msgver	---	ans..5	M	Computop Paygate Message version. Valid values: **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.
TransID	"transactionId": "..."	ans..64	M	TransactionID provided by you which should be unique for each payment
ReqId	"requestId": "..."	ans..32	O	To avoid double payments or actions (e.g. by ETM), enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction or action is submitted again with the same ReqID, Computop Paygate will not carry out the payment or new action, but will just return the status of the original transaction or action. Please note that the Computop Paygate must have a finalized transaction status for the first initial action (authentication/authorisation). This does not apply to 3-D Secure authentications that are terminated by a timeout. The 3-D Secure Timeout status does not count as a completed status in which the ReqID functionality on Paygate does not take effect. Submissions with identical ReqID for an open status will be processed regularly. **Notice:** Please note that a ReqID is only valid for 12 month, then it gets deleted at the Paygate.
RefNr	"referenceNumber": "..."		O	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 Computop settlement file (CTSF) we cannot add the additional payment data.

<table>
<tr><td></td><td></td><td></td><td></td><td>Details on supported format can be found below in payment specific section.

Only ASCII characters allowed, special characters ("Umlaute", diacritics) are not allowed and must be replaced by their ASCII-representation (e.g. ü ue, é e, ...).</td></tr>
<tr><td>schemeReferenceID</td><td>"payment": {"card": { "schemeReferenceId": "..." }}</td><td>ans..64</td><td>C</td><td>Card scheme specific transaction ID required for subsequent credential-on-file payments, delayed authorizations and resubmissions.

Mandatory: CredentialOnFile – initial false – unscheduled MIT / recurring

schemeReferenceID is returned for 3DS2-payments. In case of fallback to 3DS1 you will also need to check for TransactionId.

The schemeReferenceID is a unique identifier generated by the card brands and as a rule Computop merchants can continue to use the SchemeReferenceIDs for subscription plans that were created while using another PSP environment / Paygate MerchantID / Acquirer ContractID / Acquirer.</td></tr>
<tr><td>industrySpecificTxType</td><td>"payment": {"card": { "industrySpecificTransactionType": "..." }}</td><td>ans..20</td><td>C</td><td>This parameter is required whenever an industry specific transaction is processed according to the card brands MIT (Merchant Initiated Transactions) Framework, i.e.: specific use cases like described below.

Only supported with Omnipay and GICC.

Supported with CB2A for Reauthorization, only.

Values accepted:

<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>Resubmission</td><td>A merchant performs a re-submission in cases where it requested an authorization, but received a decline due to insufficient funds; however, the goods or services were already delivered to the cardholder.

Merchants in such scenarios can resubmit the request to recover outstanding debt from cardholders.</td></tr>
<tr><td>Reauthorization</td><td>A merchant initiates a re-authorization when the completion or fulfillment of the original order or service extends beyond the authorization validity limit set by Visa.

There are two common re-authorization scenarios:

• Split or delayed shipments at eCommerce retailers. A split shipment occurs when not all the goods ordered are available for shipment at the time of purchase. If the fulfillment of the goods takes place after the authorization

validity limit set by Visa, eCommerce merchants perform a separate authorization to ensure that consumer funds are available.

• Extended stay hotels, car rentals, and cruise lines. A re-authorization is used for stays, voyages, and/or rentals that extend beyond the authorization validity period set by Visa.</td></tr>
<tr><td>DelayedCharges</td><td>Delayed charges are performed to process a supplemental account charge after original services have been rendered and respective payment has been processed.</td></tr>
<tr><td>NoShow</td><td>Cardholders can use their Visa cards to make a guaranteed reservation with certain merchant segments. A guaranteed reservation ensures that the reservation will be honored and allows a merchant to perform a No Show transaction to charge the cardholder a penalty according to the merchant's cancellation policy.
Note: For merchants that accept token-based payment credentials to guarantee a reservation, it is necessary to perform a CIT (Account Verification Service) at the time of reservation to be able perform a No Show transaction later.</td></tr>
</table>

Note: It is always submitted in conjunction with the "schemeReferenceID" parameter. Please contact Computop Helpdesk for the supported Acquirer and card brands.</td></tr>
<tr><td>Amount</td><td>"amount": { "value": ...}</td><td>n..10</td><td>M</td><td>Amount in the smallest currency unit (e.g. EUR Cent). Please contact the Computop Helpdesk, if you want to capture amounts <100 (smallest currency unit).</td></tr>
<tr><td>Currency</td><td>"amount": { "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: A1 Currency table</td></tr>
<tr><td>card</td><td>"payment": {"card": JSON}</td><td>JSON</td><td>M</td><td>Card data</td></tr>
<tr><td>Capture</td><td>"capture": {"auto": "Yes"}

"capture": {"manual": "Yes"}

"capture": ...</td><td>an..6</td><td>OM</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>billingDescriptor</td><td>"billing": {"addressInfo": { "descriptor": "..." }}</td><td>ans..22</td><td>O</td><td>A descriptor to be printed on a card holder's statement. Please also refer to the additional comments made elsewhere for more information about rules and regulations.</td></tr>
<tr><td>OrderDesc</td><td>"order": {"description": "..."}</td><td>ans..768</td><td>O</td><td>Order description</td></tr>
<tr><td>AccVerify</td><td>"payment": {"card": { "accountVerification": "..." }}</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:

• Yes</td></tr>
<tr><td>threeDSPolicy</td><td>"payment": {"card": { "threeDsPolicy": JSON }}</td><td>JSON</td><td>O</td><td>Object specifying authentication policies and exemption handling strategies</td></tr>
<tr><td>threeDSData</td><td>"payment": {"card": { "threeDSData": JSON }}</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>
<tr><td>priorAuthenticationInfo</td><td>"payment": {"card": { "priorAuthenticationInfo": JSON }}</td><td>JSON</td><td>O</td><td>Prior Transaction Authentication Information contains optional information about a 3-D Secure cardholder authentication that occurred prior to the current transaction</td></tr>
<tr><td>browserInfo</td><td>"browserInfo": JSON</td><td>JSON</td><td>C</td><td>Accurate browser information are needed to deliver an optimized user experience. Required for 3-D Secure 2.0 transactions.</td></tr>
<tr><td>accountInfo</td><td>"accountInfo": JSON</td><td>JSON</td><td>O</td><td>The account information contains optional information about the customer account with the merchant. Optional for 3-D Secure 2.0 transactions.</td></tr>
</table>

billToCustomer	"billing": JSON	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	"shipping": JSON	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	"billing": {"addressInfo": JSON}	JSON	C	Billing address. Required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.
shippingAddress	"shipping": {"addressInfo": JSON}	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	"credentialOnFile": JSON	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	"riskIndicator": JSON	JSON	O	The Merchant Risk Indicator contains optional information about the specific purchase by the customer
subMerchantPF	"subMerchantPaymentFacilitator": JSON	JSON	O	Object specifying SubMerchant (Payment Facilitator) details Only supported by SafeCharge
TermURL	"payment": {"threeDSLegacy": { "termUrl": "..." }}	ans..256	C	Only for 3-D Secure: URL of the shop which has been selected by the Access Control Server (ACS) of the bank to transmit the result of the authentication. The bank transmits the parameters **PayID**, **TransID** and **MerchantID** via GET and the **PAResponse** parameter via POST to the TermURL. 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 3-D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response.
URLNotify	"urls": {"notify": "..."}	ans..256	C	Complete URL which Paygate 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. 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 3-D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response. **Common notes:** • We recommend to use parameter "response=encrypt" to get an encrypted response by Paygate • 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	"metadata[userData]": "..."	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.
MAC	---	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm. Details can be found here: • HMAC Authentication (Request) • HMAC Authentication (Notify)

Key	REST	Format	CND	Description	Beschreibung
msgver	---	ans..5	M	Computop Paygate Message version. Valid values: **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.	Computop Paygate Message-Version. Zulässige Werte: **Wert** / **Beschreibung** 2.0 – Mit 3-D Secure 2.x wurde eine Vielzahl zusätzlicher Daten (Browser-Information, Rechnungs-/Versand-Adresse, ...) erforderlich, um den Authentifizierungs-Prozess zu optimieren. Um diese Informationen zu handhaben, wurden die JSON-Objekte eingeführt. Der Parameter MsgVer zeigt an, dass diese Daten verwendet werden.

Key	REST	Format	CND	Description	Beschreibung
TransID	"transactionId": "..."	ans..64	M	TransactionID provided by you which should be unique for each payment	Ihre eigene TransaktionsID, die für jede Zahlung eindeutig sein muss

Key	REST	Format	CND	Description	Beschreibung
ReqId	"requestId": "..."	ans..32	O	To avoid double payments or actions (e.g. by ETM), enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction or action is submitted again with the same ReqID, Computop Paygate will not carry out the payment or new action, but will just return the status of the original transaction or action. Please note that the Computop Paygate must have a finalized transaction status for the first initial action (authentication /authorisation). This does not apply to 3-D Secure authentications that are terminated by a timeout. The 3-D Secure Timeout status does not count as a completed status in which the ReqID functionality on Paygate does not take effect. Submissions with identical ReqID for an open status will be processed regularly. **Notice:** Please note that a ReqID is only valid for 12 month, then it gets deleted at the Paygate.	Um Doppelzahlungen (z.B. durch ETM) zu vermeiden, übergeben Sie einen alphanumerischen Wert, der Ihre Transaktion oder Aktion identifiziert und nur einmal vergeben werden darf. Falls die Transaktion oder Aktion mit derselben ReqID erneut eingereicht wird, führt das Computop Paygate keine Zahlung oder weitere Aktion aus, sondern gibt nur den Status der ursprünglichen Transaktion oder Aktion zurück. Bitte beachten Sie, dass das Computop Paygate für die erste initiale Aktion (Authentifizierung/Autorisierung) einen abgeschlossenen Transaktionsstatus haben muss. Dies gilt nicht für 3-D Secure Authentifizierungen, die durch einem Timeout beendet werden. Der Status 3-D Secure Timeout gilt nicht als abgeschlossener Status, bei dem ReqID-Funktionalität am Paygate nicht greift. Einreichungen mit identischer ReqID auf einen offenen Status werden regulär verarbeitet. **Hinweis:** Bitte beachten Sie, dass eine ReqID nur 12 Monate gültig ist, danach wird sie vom Paygate gelöscht.

Key	REST	Format	CND	Description	Beschreibung
RefNr	"referenceNumber": "..."		O	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 Computop settlement file (CTSF) we cannot add the additional payment data. Details on supported format can be found below in payment specific section. Only ASCII characters allowed, special characters ("Umlaute", diacritics) are not allowed and must be replaced by their ASCII-representation (e.g. ü ue, é e, ...).	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 Computop Settlement File (CTSF) auch nicht zusätzlich angereichert werden. Informationen zum unterstützten Format finden Sie weiter unten in der zahlartspezifischen Beschreibung. Es sind ausschließlich ASCII-Zeichen erlaubt. Sonderzeichen wie ("Umlaute", ...) sind nicht erlaubt und müssen ggf. durch ASCII-Zeichen ersetzt werden (z.B. ü ue, é e, ...).

Key	REST	Format	CND	Description	Beschreibung
schemeReferenceID	"payment": {"card": {"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 schemeReferenceID is returned for 3DS2-payments. In case of fallback to 3DS1 you will also need to check for TransactionId. The schemeReferenceID is a unique identifier generated by the card brands and as a rule Computop merchants can continue to use the SchemeReferenceIDs for subscription plans that were created while using another PSP environment / Paygate MerchantID / Acquirer ContractID / Acquirer.	Spezifische Transaktions-ID des Kartenschemas, die für nachfolgende Zahlungen mit gespeicherten Zugangsdaten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist. Pflicht: CredentialOnFile – initial false – unschedule MIT / recurring schemeReferenceID wird bei 3DS2-Zahlungsvorgängen zurückgegeben. Bei einem Fallback auf 3DS1 prüfen Sie bitte zusätzlich auf TransactionId. Die SchemeReferenceID ist eine eindeutige Kennung, die von den Kartenmarken generiert wird. In der Regel können Computop-Händler die SchemeReferenceIDs für Abonnements übergreifend verwenden, welche unter Verwendung eines anderen PSP / separater Paygate-MerchantID / separater Acquirer ContractID / Acquirer erstellt wurden.

Key	REST	Format	CND	Description	Beschreibung
industrySpecificTxType	"payment": {"card": {"industrySpecificTransactionType": "..." }}	ans..20	C	This parameter is required whenever an industry specific transaction is processed according to the card brands MIT (**M**erchant **I**nitiated **T**ransactions) Framework, i.e.: specific use cases like described below. Only supported with Omnipay and GICC. Supported with CB2A for Reauthorization, only. Values accepted: **Value** / **Description** Resubmission: A merchant performs a re-submission in cases where it requested an authorization, but received a decline due to insufficient funds; however, the goods or services were already delivered to the cardholder. Merchants in such scenarios can resubmit the request to recover outstanding debt from cardholders. Reauthorization: A merchant initiates a re-authorization when the completion or fulfillment of the original order or service extends beyond the authorization validity limit set by Visa. There are two common re-authorization scenarios: • ***Split or delayed shipments*** at eCommerce retailers. A split shipment occurs when not all the goods ordered are available for shipment at the time of purchase. If the fulfillment of the goods takes place after the authorization validity limit set by Visa, eCommerce merchants perform a separate authorization to ensure that consumer funds are available. • ***Extended stay*** *hotels, car rentals, and cruise lines*. A re-authorization is used for stays, voyages, and/or rentals that extend beyond the authorization validity period set by Visa. DelayedCharges: Delayed charges are performed to process a supplemental account charge after original services have been rendered and respective payment has been processed. NoShow: Cardholders can use their Visa cards to make a guaranteed reservation with certain merchant segments. A guaranteed reservation ensures that the reservation will be honored and allows a merchant to perform a No Show transaction to charge the cardholder a penalty according to the merchant's cancellation policy. ***Note***: For merchants that accept token-based payment credentials to guarantee a reservation, it is necessary to perform a CIT (Account Verification Service) at the time of reservation to be able perform a No Show transaction later.	Dieser Parameter ist erforderlich, wenn eine branchenspezifische Transaktion entsprechend dem Kartenmarken MIT-Framework (**M**erchant **I**nitiated **T**ransactions) verarbeitet wird. Der Parameter wird nur für bestimmte Use Cases verwendet, die unten beschrieben sind. Wird nur von Omnipay und GICC unterstützt. CB2A unterstützt nur den Wert Reauthorization Zulässige Werte: **Wert** / **Beschreibung** Resubmission: Ein Händler führt eine erneute Einreichung durch, wenn er eine Autorisierung angefordert hat, diese aber aufgrund unzureichender Mittel abgelehnt wurde; die Waren oder Dienstleistungen wurden jedoch bereits an den Karteninhaber geliefert. In solchen Szenarien können Händler den Antrag auf Beitreibung ausstehender Forderungen von Karteninhabern erneut einreichen. Reauthorization: Ein Händler leitet eine erneute Autorisierung ein, wenn Abschluss oder Erfüllung der ursprünglichen Bestellung oder Dienstleistung die von Visa festgelegte Gültigkeitsdauer der Autorisierung überschreitet. Es gibt zwei gängige Szenarien für die erneute Autorisierung: • ***Geteilte oder verzögerte Lieferung*** be E-Commerce-Händlern. Eine Teillieferung liegt vor, wenn zum Zeitpunkt des Kaufs nicht alle bestellten Waren versandbereit sind. Erfolgt die Lieferung der Ware nach der von Visa festgelegten Gültigkeitsdauer der Autorisierung, führen E-Commerce-Händler eine separate Autorisierung durch, um sicherzustellen, dass Kundengelder verfügbar sind. • ***Verlängerte*** *Hotelaufenthaltens, Autovermietungen und Keuzfahrten*. Eine erneute Autorisierung wird für Aufenthalte, Reisen und/oder Anmietungen verwendet, die über die von Visa festgelegte Gültigkeitsdauer der Autorisierung hinausgehen. DelayedCharges: Verzögerte Gebühren dienen dazu, um eine zusätzliche Kontogebühr zu verarbeiten, nachdem die ursprünglichen Dienstleistungen erbracht und die entsprechende Zahlung verarbeitet wurde. NoShow: Karteninhaber können mit ihren Visa-Karten eine garantierte Reservierung bei bestimmten Händlersegmenten vornehmen. Eine garantierte Reservierung stellt sicher, dass die Reservierung berücksichtigt wird und ermöglicht es einem Händler, eine No-Show-Transaktion durchzuführen, um dem Karteninhaber eine Strafe gemäß den Stornierungsbedingungen des Händlers zu berechnen. ***Hinweis***: Für Händler, die tokenbasierte Zahlungsinformationen akzeptieren, um eine Reservierung zu garantieren, ist es zum Zeitpunkt der Reservierung erforderlich, einen CIT (Kontoverifizierungsservice) durchzuführen, um später eine No-Show-Transaktion durchführen zu können. **Hinweis**: Das wird immer zusammen mit dem Parameter "schemeReferenceID" übermittelt. Bezüglich unterstützer Acquirer und Kartenmarken wenden Sie sich bitte an den Computop Helpdesk.

				Note: It is always submitted in conjunction with the "schemeReferenceID" parameter. Please contact Computop Helpdesk for the supported Acquirer and card brands.	

Key	REST	Format	CND	Description	Beschreibung
Amount	"amount": { "value": ...}	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the Computop Helpdesk, if you want to capture amounts <100 (smallest currency unit).	Betrag in der kleinsten Währungseinheit (z.B. EUR Cent). Bitte wenden Sie sich an den Computop Helpdesk, wenn Sie Beträge < 100 (kleinste Währungseinheit) buchen möchten.

Key	REST	Format	CND	Description	Beschreibung
Currency	"amount": { "currency": "..."}	a3	M	Currency, three digits DIN / ISO 4217, e.g. EUR, USD, GBP. Please find an overview here: A1 Currency table	Währung, drei Zeichen DIN / ISO 4217, z.B. EUR, USD, GBP. Hier eine Übersicht: A1 Währungstabelle

Key	REST	Format	CND	Description	Beschreibung
card	"payment": {"card": JSON}	JSON	M	Card data	Kartendaten

Key	REST	Format	CND	Description	Beschreibung
Capture	"capture": {"auto": "Yes"} "capture": {"manual": "Yes"} "capture": ...	an..6	OM	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).	Bestimmt Art und Zeitpunkt der Buchung (engl. Capture). **Buchungsart** / **Beschreibung** **AUTO** – Buchung sofort nach Autorisierung (Standardwert). **MANUAL** – Buchung erfolgt durch den Händler - in der Regel die Buchung zum Zeitpunkt der Warenauslieferung bzw. Leistungserbringung. **<Zahl>** – Verzögerung in Stunden bis zur Buchung (ganze Zahl; 1 bis 696).

Key	REST	Format	CND	Description	Beschreibung
billingDescriptor	"billing": {"addressInfo": { "descriptor": "..." }}	ans..22	O	A descriptor to be printed on a card holder's statement. Please also refer to the additional comments made elsewhere 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	"order": {"description": "..."}	ans..768	O	Order description	Beschreibung der Bestellung
AccVerify	"payment": {"card": { "accountVerification": "..." }}	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**	Indikator zur Anforderung einer Konto-Verifizierung (alias Nullwert-Autorisierung). Wenn eine Konto-Verifizierung angefordert wird, ist der übermittelte Betrag optional und wird für die tatsächliche Zahlungstransaktion (d.h. Autorisierung) ignoriert. Zulässige Werte: • **Yes**
threeDSPolicy	"payment": {"card": { "threeDsPolicy": JSON }}	JSON	O	Object specifying authentication policies and exemption handling strategies	Objekt, dass die Authentisierungs-Richtlinien und Strategien zur Behandlung von Ausnahmen angibt
threeDSData	"payment": {"card": { "threeDSData": JSON }}	JSON	C	Object detailing authentication data in case authentication was performed through a third party or by the merchant	Objekt mit Details der Authentisierungsdaten, falls die Authentisierung durch Dritte oder durch den Händler durchgeführt wurde
priorAuthenticationInfo	"payment": {"card": { "priorAuthenticationInfo": JSON }}	JSON	O	Prior Transaction Authentication Information contains optional information about a 3-D Secure cardholder authentication that occurred prior to the current transaction	Das Objekt Prior Transaction Authentication Information enthält optionale Informationen über eine 3-D Secure-Authentisierung eines Karteninhabers, die vor der aktuellen Transaktion erfolgt ist
browserInfo	"browserInfo": JSON	JSON	C		Exakte Browserinformationen sind nötig, um eine optimierte Nutzererfahrung zu liefern. Erforderlich für 3-D Secure 2.0

				Accurate browser information are needed to deliver an optimized user experience. Required for 3-D Secure 2.0 transactions.	Transaktionen.
accountInfo	"accountInfo": JSON	JSON	O	The account information contains optional information about the customer account with the merchant. Optional for 3-D Secure 2.0 transactions.	Die Kontoinformationen enthalten optionale Informationen über das Kundenkonto beim Händler
billToCustomer	"billing": JSON	JSON	C	The customer that is getting billed for the goods and / or services. Required unless market or regional mandate restricts sending this information.	Der Kunde, dem die Waren und / oder Dienstleistungen in Rechnung gestellt werden. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
shipToCustomer	"shipping": JSON	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.	Der Kunde, an den die Waren und / oder Dienstleistungen gesendet werden. Erforderlich (falls verfügbar und von billToCustomer abweichend), sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
billingAddress	"billing": {"addressInfo": JSON}	JSON	C	Billing address. Required for 3-D Secure 2.0 (if available) unless market or regional mandate restricts sending this information.	Rechnungsadresse. Erforderlich für 3-D Secure 2.0 (falls verfügbar), sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
shippingAddress	"shipping": {"addressInfo": JSON}	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.	Lieferadresse. Falls abweichend von billingAddress, erforderlich für 3-D Secure 2.0 (falls verfügbar), sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
credentialOnFile	"credentialOnFile": JSON	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.	Objekt, dass Art und Reihe der Transaktionen angibt, die unter Verwendung von beim Händler hinterlegten Zahlungsdaten (z.B. Kontonummer oder Zahlungs-Token) zur Verarbeitung künftiger Käufe eines Kunden erfolgen. Erforderlich, falls zutreffend.
merchantRiskIndicator	"riskIndicator": JSON	JSON	O	The Merchant Risk Indicator contains optional information about the specific purchase by the customer	Der Händler-Risikoindikator enthält optionale Informationen über den bestimmten Einkauf des Kunden
subMerchantPF	"subMerchantPaymentFacilitator": JSON	JSON	O	Object specifying SubMerchant (Payment Facilitator) details Only supported by SafeCharge	Objekt, das die Details des SubMerchant (Payment Facilitator) angibt Wird ausschließlich von SafeCharge unterstützt.

Key	REST	Format	CND	Description	Beschreibung
TermURL	"payment": {"threeDSLegacy": { "termUrl": "..." }}	ans..256	C	Only for 3-D Secure: URL of the shop which has been selected by the Access Control Server (ACS) of the bank to transmit the result of the authentication. The bank transmits the parameters **PayID**, **TransID** and **MerchantID** via GET and the **PAResponse** parameter via POST to the TermURL. 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 3-D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response.	Nur bei 3-D Secure: URL des Shops, die vom Access Control Server (ACS) der Bank aufgerufen wird, um das Ergebnis der Authentisierung zu übermitteln. Dabei übergibt die Bank per GET die Parameter **PayID**, **TransID**, **MerchantID** und per POST den Parameter **PAResponse** an die TermURL. Im Falle einer vom Händler initiierten wiederkehrenden Transaktion sind die JSON-Objekte (außer credentialOnFile und card), URLNotify und TermURL keine obligatorischen Parameter, da kein 3-D Secure und keine Risikobewertung durch die kartenausgebende Bank erfolgt und das Zahlungsergebnis direkt erfolgt innerhalb der Antwort zurückgegeben.

Key	REST	Format	CND	Description	Beschreibung
URLNotify	"urls": {"notify": "..."}	ans..256	C	Complete URL which Paygate 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. 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 3-D Secure and no risk evaluation is done by the card issuing bank and the payment result is directly returned within the response. **Common notes:** • We recommend to use parameter "response=encrypt" to get an encrypted response by Paygate • 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.	Vollständige URL, die das Paygate aufruft, um den Shop zu benachrichtigen. Die URL darf nur über Port 443 aufgerufen werden. Sie darf keine Parameter enthalten: Nutzen Sie stattdessen den Parameter UserData. Im Falle einer vom Händler initiierten wiederkehrenden Transaktion sind die JSON-Objekte (außer credentialOnFile und card), URLNotify und TermURL keine obligatorischen Parameter, da kein 3-D Secure und keine Risikobewertung durch die kartenausgebende Bank erfolgt und das Zahlungsergebnis direkt erfolgt innerhalb der Antwort zurückgegeben. **Allgemeine Hinweise:** • Wir empfehlen, den Parameter "response=encrypt" zu verwenden, um eine verschlüsselte Antwort von Paygate zu erhalten • Betrüger könnten das verschlüsselte DATA-Element kopieren, welches an URLFailure gesendet wurde, und betrügerisch dasselbe DATA an URLSuccess/URLNotify senden. Überprüfen Sie daher unbedingt den "code"-Wert des DATA-Elements. Nur eine Antwort mit "code=00000000" sollte als erfolgreich angesehen werden.

Key	REST	Format	CND	Description	Beschreibung

UserData	"metadata [userData]": "..."	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.	Wenn beim Aufruf angegeben, übergibt das Paygate die Parameter mit dem Zahlungsergebnis an den Shop.

Key	REST	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)	Hash Message Authentication Code (HMAC) mit SHA-256-Algorithmus. Details finden Sie hier: • HMAC-Authentisierung (Anfrage) • HMAC-Authentisierung (Notify)

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 (authentication)

In case of using REST API

In case of using REST API you will always receive a link where the merchant has to redirect the consumer to complete the payment.

REST	Format	CND	Description
"paymentId": "..."	an32	M	May be "00000000000000000000000000000000" if not yet set by Computop Paygate
"_Links.self.type": "..."	an..20	M	"application/json"
"_Links.redirect.href": "..."	an..1024	M	Merchant needs to redirect consumer to this URL to complete payment
"_Links.redirect.type": "..."	an..20	M	"text/html"

Merchant can use inquire.aspx

The following table describes the result parameters with which the Computop Paygate 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 Computop
PayID	an32	M	ID assigned by Paygate 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 Paygate
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment
refnr		O	Reference number as given in request
Status	a..20	M	Status of the transaction. Values accepted: • **`AUTHENTICATION_REQUEST`** • **`PENDING`** • **`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 Paygate Response Codes (A4 Error codes)
UserData	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.
card	JSON	M	Card data
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	C	Object containing the data elements required to construct the Payer Authentication request in case of a **fallback** to 3-D Secure 1.0.

Key	Format	CND	Description	Beschreibung
PayID	an32	M	ID assigned by Paygate for the payment, e.g. for referencing in batch files as well as for capture or credit request.	Vom Paygate vergebene ID für die Zahlung; z.B. zur Referenzierung in Batch-Dateien sowie im Capture- oder Credit-Request.

Key	Format	CND	Description	Beschreibung
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by Paygate	Vom Paygate vergebene ID für alle einzelnen Transaktionen (Autorisierung, Buchung, Gutschrift), die für eine Zahlung durchgeführt werden

Key	Format	CND	Description	Beschreibung
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment	Ihre eigene TransaktionsID, die für jede Zahlung eindeutig sein muss

Key	Format	CND	Description	Beschreibung
refnr		O	Reference number as given in request	Referenznummer wie im Request angegeben
Status	a..20	M	Status of the transaction. Values accepted: • **`AUTHENTICATION_REQUEST`** • **`PENDING`** • **`FAILED`**	Status der Transaktion. Zulässige Werte: • **`AUTHENTICATION_REQUEST`** • **`PENDING`** • **`FAILED`**

Key	Format	CND	Description	Beschreibung
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!	Nähere Beschreibung bei Ablehnung der Zahlung. Bitte nutzen Sie **nicht** den Parameter **Description, sondern** Code für die Auswertung des Transaktionsstatus!

Key	Format	CND	Description	Beschreibung
Code	n8	M	Error code according to Paygate Response Codes (A4 Error codes)	Fehlercode gemäß Paygate Antwort-Codes (A4 Fehlercodes)

Key	Format	CND	Description	Beschreibung
UserData	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.	Wenn beim Aufruf angegeben, übergibt das Paygate die Parameter mit dem Zahlungsergebnis an den Shop.

Key	Format	CND	Description	Beschreibung
card	JSON	M	Card data	Kartendaten
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	C	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.

versioningData

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).

BASEURL= https://www.computop-paygate.com/

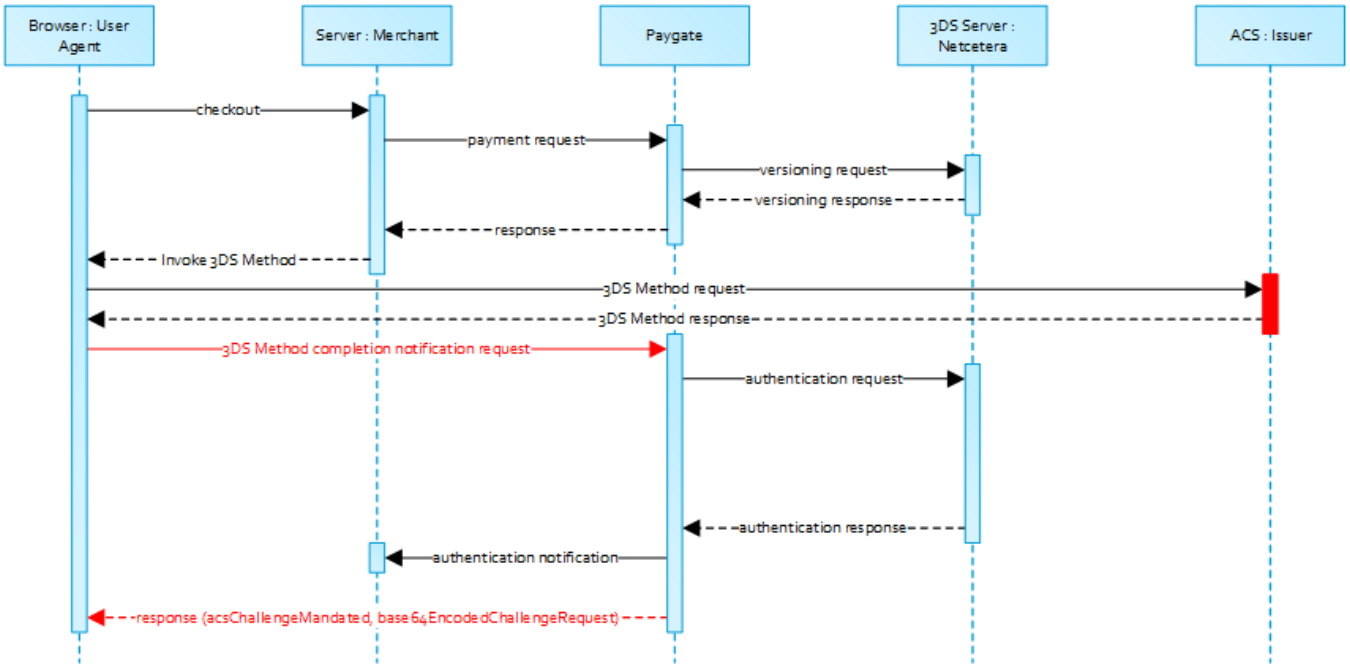

```
{
        "threeDSServerTransID": "14dd844c-b0fc-4dfe-8635-366fbf43468c",
        "acsStartProtocolVersion": "2.1.0",
        "acsEndProtocolVersion": "2.1.0",
        "dsStartProtocolVersion": "2.1.0",
        "dsEndProtocolVersion": "2.1.0",
        "threeDSMethodURL": "http://www.acs.com/script",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

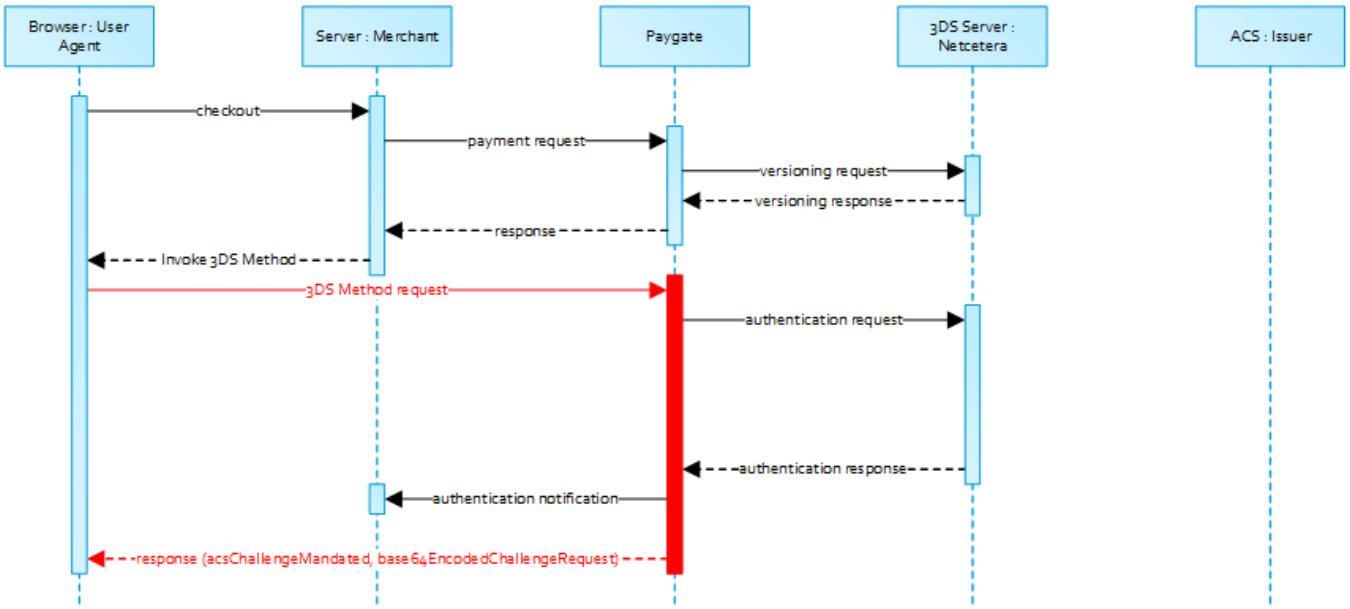
The **versioningData** object contains a value for **threeDSMethodURL** . The merchant is supposed to invoke the 3-D Secure Method via a hidden HTML iframe in the cardholder browser and send a form with a field named **threeDSMethodData** via HTTP POST to the ACS 3-D Secure Method URL.

3-D Secure Method: threeDSMethodURL



Please note that the **threeDSMethodURL** will be populated by Computop Paygate 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 Computop Paygate 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 interact 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 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 Computop Paygate 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 Computop Paygate .

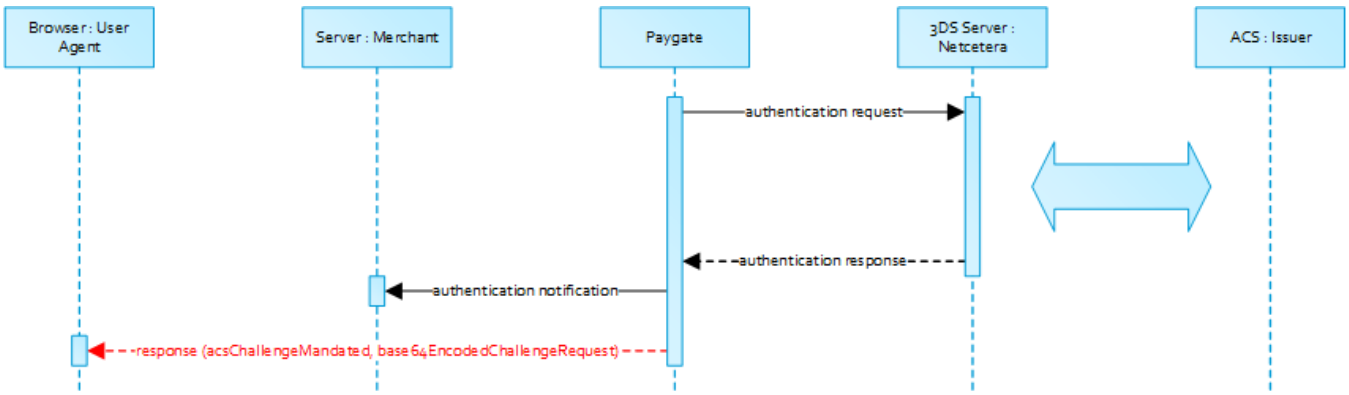
Authentication

If 3-D Secure Method is supported by the issuer ACS and was invoked by the merchant Computop Paygate 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 Computop Paygate 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, Computop Paygate 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: • true Challenge is mandated by local/regional regulations • false Challenge is not mandated by local/regional regulations, but is **deemed necessary** by the ACS
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": false,
        "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 Computop
PayID	an32	M	ID assigned by Paygate 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 Paygate Response Codes (A4 Error 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	Beschreibung
PayID	an32	M	ID assigned by Paygate for the payment, e.g. for referencing in batch files as well as for capture or credit request.	Vom Paygate vergebene ID für die Zahlung; z.B. zur Referenzierung in Batch-Dateien sowie im Capture- oder Credit-Request.

Key	Format	CND	Description	Beschreibung
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment	Ihre eigene TransaktionsID, die für jede Zahlung eindeutig sein muss

Key	Format	CND	Description	Beschreibung
Code	n8	M	Error code according to Paygate Response Codes (A4 Error codes)	Fehlercode gemäß Paygate Antwort-Codes (A4 Fehlercodes)

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)	Hash Message Authentication Code (HMAC) mit SHA-256-Algorithmus. Details finden Sie hier: • HMAC-Authentisierung (Anfrage) • HMAC-Authentisierung (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 Computop Paygate and must not be changed.

Authorization

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

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

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

Payment Notification

In case of using REST API

In case of using REST API you will always receive a link where the merchant has to redirect the consumer to complete the payment.

REST	Format	CND	Description
"paymentId": "..."	an32	M	May be "00000000000000000000000000000000" if not yet set by Computop Paygate
"_Links.self.type": "..."	an..20	M	"application/json"
"_Links.redirect.href": "..."	an..1024	M	Merchant needs to redirect consumer to this URL to complete payment

"_Links.redirect.type": "..."	an..20	M	"text/html"

Merchant can use inquire.aspx

In case of using Key-Value-Pair API

The following table gives the result parameters which Computop Paygate transmits to **URLSuccess** or **URLFailure** and **URLNotify**. If you have specified the **Response=encrypt** parameter, the following parameters are sent Blowfish encrypted 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 Computop
msgver	ans..5	M	Computop Paygate Message version. Valid values: **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 Paygate 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 Paygate
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 schemeReferenceID is returned for 3DS2-payments. In case of fallback to 3DS1 you will also need to check for TransactionId. The schemeReferenceID is a unique identifier generated by the card brands and as a rule Computop merchants can continue to use the SchemeReferenceIDs for subscription plans that were created while using another PSP environment / Paygate MerchantID / Acquirer ContractID / Acquirer.
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 Paygate Response Codes (A4 Error 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.
externalPaymentData	JSON	O	Optional additional data from acquirer/issuer/3rd party for authorization.
PCNr	n16	O	Pseudo Card Number: Random number generated by Computop Paygate which represents a genuine credit card number. The pseudo card number (PCN) starts with 0 and the last 3 digits correspond to those of the real card number. The PCN can be used like a genuine card number for authorisation, capture and credits.

			PCNr is a **response value** from Computop Paygate and is sent as CCNr in Request or part of card-JSON

Key	Format	CND	Description	Beschreibung
msgver	ans..5	M	Computop Paygate Message version. Valid values: **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.	Computop Paygate Message-Version. Zulässige Werte: **Wert** / **Beschreibung** 2.0 – Mit 3-D Secure 2.x wurde eine Vielzahl zusätzlicher Daten (Browser-Information, Rechnungs-/Versand-Adresse, ...) erforderlich, um den Authentifizierungs-Prozess zu optimieren. Um diese Informationen zu handhaben, wurden die JSON-Objekte eingeführt. Der Parameter MsgVer zeigt an, dass diese Daten verwendet werden.

Key	Format	CND	Description	Beschreibung
PayID	an32	M	ID assigned by Paygate for the payment, e.g. for referencing in batch files as well as for capture or credit request.	Vom Paygate vergebene ID für die Zahlung; z.B. zur Referenzierung in Batch-Dateien sowie im Capture- oder Credit-Request.

Key	Format	CND	Description	Beschreibung
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by Paygate	Vom Paygate vergebene ID für alle einzelnen Transaktionen (Autorisierung, Buchung, Gutschrift), die für eine Zahlung durchgeführt werden

Key	Format	CND	Description	Beschreibung
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment	Ihre eigene TransaktionsID, die für jede Zahlung eindeutig sein muss

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 schemeReferenceID is returned for 3DS2-payments. In case of fallback to 3DS1 you will also need to check for TransactionId. The schemeReferenceID is a unique identifier generated by the card brands and as a rule Computop merchants can continue to use the SchemeReferenceIDs for subscription plans that were created while using another PSP environment / Paygate MerchantID / Acquirer ContractID / Acquirer.	Spezifische Transaktions-ID des Kartenschemas, die für nachfolgende Zahlungen mit gespeicherten Zugangsdaten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist. Pflicht: CredentialOnFile – initial false – unschedule MIT / recurring schemeReferenceID wird bei 3DS2-Zahlungsvorgängen zurückgegeben. Bei einem Fallback auf 3DS1 prüfen Sie bitte zusätzlich auf TransactionId. Die SchemeReferenceID ist eine eindeutige Kennung, die von den Kartenmarken generiert wird. In der Regel können Computop-Händler die SchemeReferenceIDs für Abonnements übergreifend verwenden, welche unter Verwendung eines anderen PSP / separater Paygate-MerchantID / separater Acquirer ContractID / Acquirer erstellt wurden.

Key	Format	CND	Description	Beschreibung
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	Beschreibung
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!	Nähere Beschreibung bei Ablehnung der Zahlung. Bitte nutzen Sie **nicht** den Parameter **Description, sondern** Code für die Auswertung des Transaktionsstatus!

Key	Format	CND	Description	Beschreibung
Code	n8	M	Error code according to Paygate Response Codes (A4 Error codes)	Fehlercode gemäß Paygate Antwort-Codes (A4 Fehlercodes)

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)	Hash Message Authentication Code (HMAC) mit SHA-256-Algorithmus. Details finden Sie hier: • HMAC-Authentisierung (Anfrage) • HMAC-Authentisierung (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
external Payment Data	JSON	O	Optional additional data from acquirer/issuer/3rd party for authorization.	Optionale Daten des Acquirers/Issuers/externen Dienstleisters für eine Autorisierung

Key	Format	CND	Description	Beschreibung
PCNr	n16	O	Pseudo Card Number: Random number generated by Computop Paygate which represents a genuine credit card number. The pseudo card number (PCN) starts with 0 and the last 3 digits correspond to those of the real card number. The PCN can be used like a genuine card number for authorisation, capture and credits. PCNr is a **response value** from Computop Paygate and is sent as CCNr in Request or part of card-JSON	Pseudo Card Number: Vom Computop Paygate generierte Zufallszahl, die eine reale Kreditkartennummer repräsentiert. Die Pseudokartennummer (PKN) beginnt mit 0, und die letzten 3 Stellen entsprechen denen der realen Kartennummer. Die PKN kann wie eine Kreditkartennummer für Autorisierung, Buchung und Gutschriften verwendet werden. PCNr ist ein **Antwortwert** von Computop Paygate und kann ebenfalls als CCNr im Request oder als Teil von card-JSON verwendet werden.

Browser Payment Response

Additionally the JSON formatted data elements as listed below are transferred 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 Computop
Len	integer	M	Length of the unencrypted **`Data`** string
Data	string	M	Blowfish encrypted string containing 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 containing 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 Computop
PayID	an32	M	ID assigned by Paygate 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	Beschreibung
PayID	an32	M	ID assigned by Paygate for the payment, e.g. for referencing in batch files as well as for capture or credit request.	Vom Paygate vergebene ID für die Zahlung; z.B. zur Referenzierung in Batch-Dateien sowie im Capture- oder Credit-Request.

Key	Format	CND	Description	Beschreibung
TransID	ans..64	M	TransactionID provided by you which should be unique for each payment	Ihre eigene TransaktionsID, die für jede Zahlung eindeutig sein muss

Sample decrypted Data

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