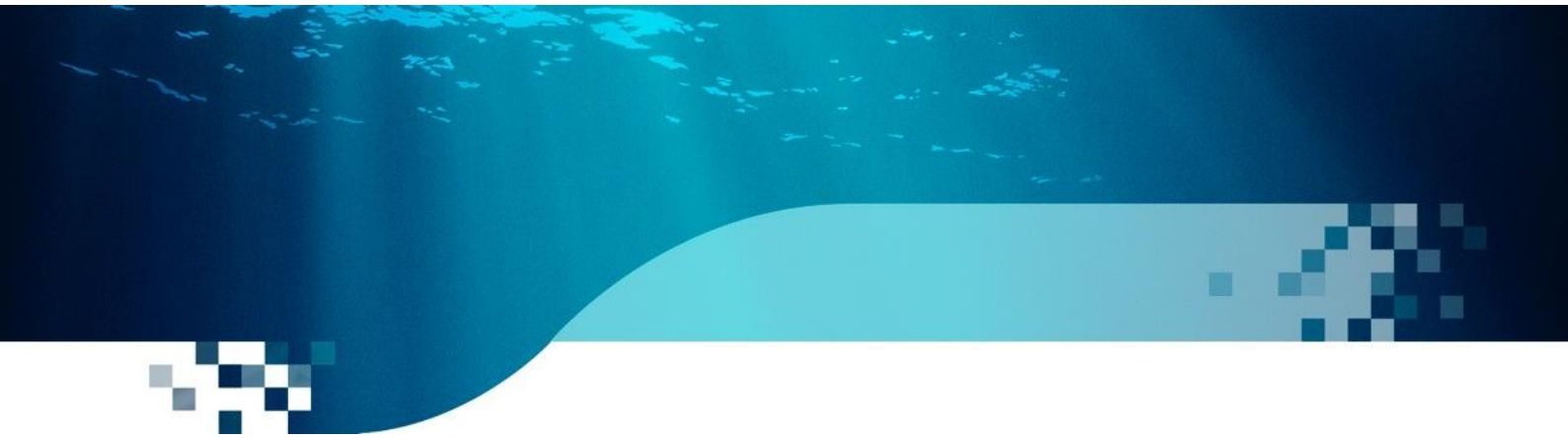


Technische Referenz EVO E-PAY

Integration EMV 3-D Secure

27. April 2022

EVOGMBH_TRE_0095_2022-04-27_DE

Änderungen gegenüber der vorherigen Dokumentversion

Kapitel Nr.	Stichwort	Was wurde geändert?	Verfasser (Kürzel)	Datum
3.1.2	**URLSuccess**, **URLFailure**, **URLNotify**	Allgemeine Hinweise zu **URLSuccess**, **URLFailure** und **URLNotify** eingefügt	HD	27.04.2022

Zu diesem Dokument

Erläuterung von Rollenbegriffen

Im komplexen Szenario der Zahlungsabwicklung gibt es drei Hauptakteure. Je nach Blickwinkel und Situation können zwei von ihnen als Kunden agieren und zwei von ihnen als Anbieter. Um eine klare Unterscheidung zu ermöglichen, werden konsequent die folgenden Begriffe verwendet:

Händler oder Mandant

Nutzt als Händler oder Dienstleister das Angebot der EVO Payments für die Abwicklung der Bezahlung für von ihm angebotene Waren oder Dienstleistungen. Er ist Vertragspartner und somit direkter „Kunde" der EVO Payments.

Kunde

Bezieht Waren oder Dienstleistungen des Händlers bzw. Dienstleisters. Er ist Vertragspartner des Händlers bzw. Dienstleisters und kein Kunde der EVO Payments.

EVO Payments

Erbringt Leistungen im Zusammenhang mit der Zahlungsabwicklung und fungiert als Bindeglied zwischen den Parteien, insbesondere dem Händler bzw. Dienstleister und weiteren Stellen wie Kartenorganisationen und anderen an der Abwicklung von Zahlungen beteiligten Institutionen.

Hinweis zu Begriffen in der EVO Payments XML-Sprache

Alle Begriffe, die zur XML-Sprache gehören, werden in diesem Dokument in der Schriftart „Consolas" dargestellt (z. B. **`PaymentTransactionType`**). Wenn solche XML-Begriffe durch Zeilenumbrüche geteilt werden, enthalten sie keinen Trennungsstrich, da dies zu Irrtümern in der Programmier-Schreibweise führen kann. Wenn also ein XML-Begriff in diesem Dokument durch einen Zeilenumbruch geteilt ist, muss der Umbruch bei der Programmierung ignoriert werden.

Beispiel:
Text in diesem Dokument Text in diesem Dokument Text in diesem Dokument Text in diesem **`Payment TransactionType`** Dokument Text in diesem Dokument Text in …

Text beim Programmieren: **`PaymentTransactionType`**

Inhaltsverzeichnis

1. Regulatorische Anforderungen

1.1 EBA Mandat

Die Europäische Bankenaufsichtsbehörde (EBA) hat angeordnet, dass alle Zahler, die online auf ihr Zahlungskonto zugreifen und elektronische Zahlungstransaktionen über einen Remote-Kanal auslösen, beginnend ab 14. September 2019 stark authentifiziert werden müssen (alias Starke Kundenauthentifizierung (SCA)). Die Kartenorganisationen haben diese Möglichkeit ergriffen, um das etablierte Protokoll 3-D Secure für die Karteninhaber-Authentifizierung zu überarbeiten und mehrere Probleme anzugehen, welche die Annahme im Markt gebremst haben.

1.2 3DS 2.0

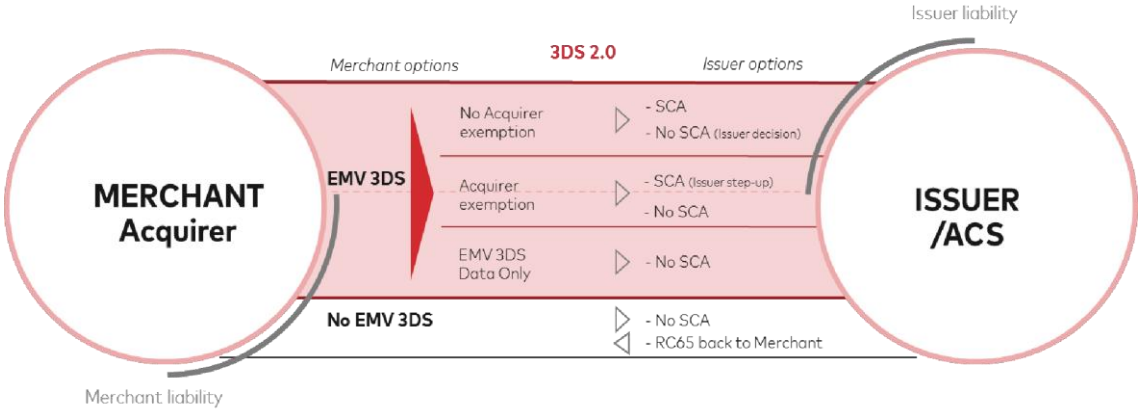
Bisher hatten Internethändler die Wahl, dem Karteninhaber eine Challenge (z.B. TAN / Passwort) zu präsentieren oder 3DS gänzlich zu übergehen. Einige haben einen dynamischen Ansatz basierend auf dem PSP oder der eigenen Risikobewertung gewählt, aber viele Händler schätzten einen reibungslosen Kassenvorgang und hohe Konversionsraten mehr als die möglichen Vorteile einer Haftungsverschiebung. Die Gesamtstrategie der Kartenorganisationen für 3DS 2.0 ist es, Reibereien durch eine verbesserte Erfahrung der Karteninhaber (Geräte-Bewusstsein) zu verringern und Ausnahmen von der SCA basierend auf einer robusten Transaktionsrisikoanalyse (TRA) auszunutzen mit dem obersten Ziel, optimale Autorisierungsleistung und Konversionsraten zu erreichen. Daher ist die TRA entscheidend für reibungslose Zahlungsabläufe für Remote-Transaktionen mit geringem Risiko. Deshalb hat das Protokoll 3DS 2.0 eine Unmenge zusätzlicher Datenpunkte eingeführt, die dem Kartenherausgeber zur Unterstützung der Transaktionsrisikoanalyse und für die Anwendung von Ausnahmen der SCA übermittelt werden können.

SCA wird erforderlich, wenn:

- die Transaktion nicht außerhalb vom Geltungsbereich der PSD2 RTS ist
- keine Ausnahme der PSD2 SCA für eine Zahlungstransaktion zutrifft
- eine Karte zu einer Händler-Datenbank hinzugefügt wird (hinterlegte Karte)
- eine Vereinbarung für wiederkehrende Zahlungen über feste oder variable Beträge beginnt, einschließlich der Festlegung des anfänglichen Mandats für vom Händler ausgelöste Transaktionen (MIT)
- eine Vereinbarung für wiederkehrende Zahlungen zu einem höheren Betrag geändert wird (beispielsweise ein Premium-Angebot)
- ein White-Listing eingerichtet wird (oder zum Ansehen/Ändern von White-Lists)
- ein Gerät mit einem Karteninhaber verknüpft wird

1.3 Haftungsverschiebung

Als Daumenregel gilt, wenn die Authentifizierung des Karteninhabers über 3-D Secure erfolgt ist, sind Händler normalerweise vor Streitigkeiten bezüglich Betrug im E-Commerce geschützt und die Haftung verschiebt sich vom Händler / Acquirer zum Kartenherausgeber. Es gibt jedoch Ausnahmen vom Schutz des Händlers vor Streitigkeiten. Im Kontext von 3DS 2.0 sind Händler regelmäßig nicht geschützt, falls gewährte Ausnahmen gemäß PSD2 RTS aktiv vom Händler / Acquirer angefragt worden sind.
Das folgende Diagramm zeigt Optionen und Haftungen unter den Anforderungen von PSD2 RTS gemäß Mastercard.

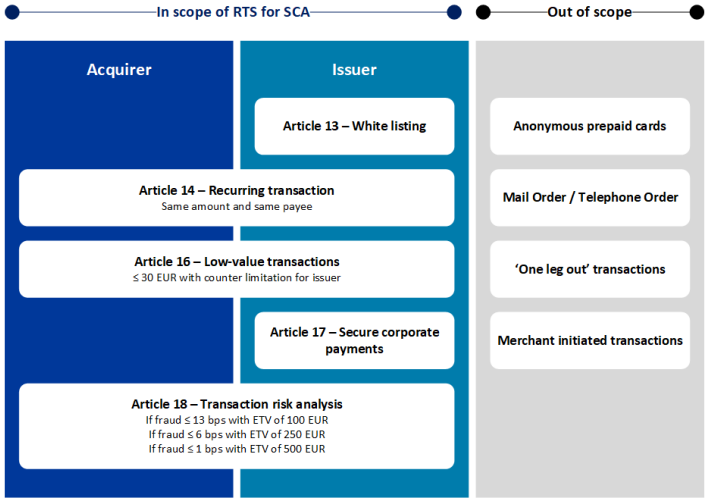


1.4 3DS 2.0 und Compliance zur DSGVO

Karteninhabern müssen ausführliche Informationen darüber gegeben werden, wie ihre Daten erfasst, verarbeitet und verwendet werden. Das kann über eine Datenschutzerklärung erreicht werden, die mindestens die Arten der verarbeiteten Daten, den Zeck ihrer Verarbeitung, die verwendeten Daten usw. enthält. Kartenorganisationen und Kartenherausgeber verwenden die EMV 3DS Daten für keine anderen Zwecke als Betrugsprävention und Authentifizierung. Das schließt die Verwendung persönlicher Daten für andere Zwecke wie Verkauf, Marketing und Data-Mining (außer zur Betrugsprävention) aus.

1.5 Ausnahmen und Ausklammerungen der PSD2 SCA

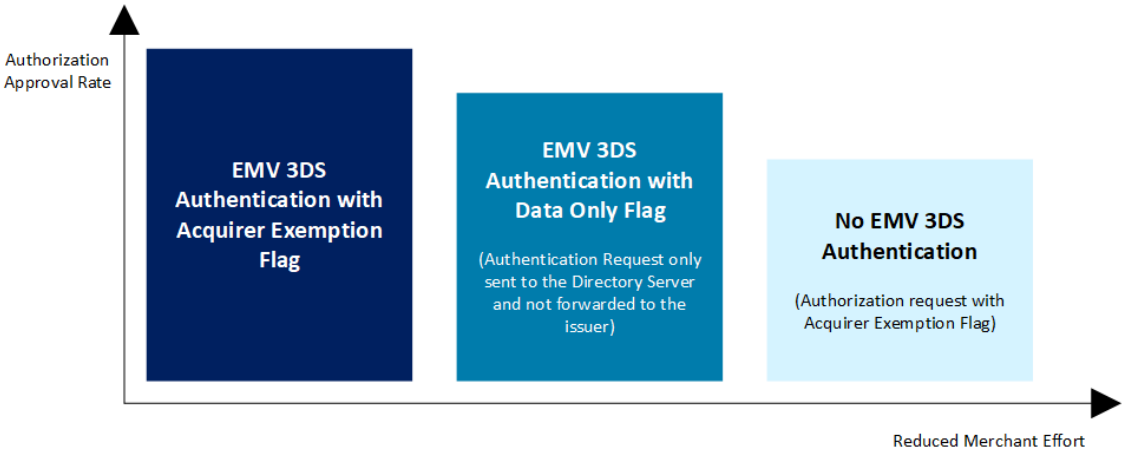
Gemäß den technischen Regulierungsstandards (RTS) gibt es einige wichtige Ausnahmen der SCA, die unter verschiedenen Bedingungen gelten können, welche im folgenden Diagramm dargestellt sind.



2. EVO E-PAY

2.1 Authentifizierungs-Optionen

Einem Acquirer kann erlaubt sein, infolge geringer Betrugsraten und TRA die SCA nicht anzuwenden. Für diese Ausnahmen gibt es verschiedene Optionen zur Verarbeitung, die im folgenden Diagramm dargestellt sind.



Standardmäßig schlägt EVO Payments anwendbare Ausnahmen (sofern unterstützt) im EMV 3DS Authentifizierungsablauf dem Kartenherausgeber vor, um die bestmöglichen Zustimmungsraten der Autorisierung zu erreichen.

EBA-Op-2018-04, Paragraph 47 - Klartstellung zu PSP (Acquirer-Betrugsraten)
Die Betrugsrate ist im Anhang A der RTS definiert und wird für alle Überweisungs-Transaktionen und alle Kartenzahlungen berechnet und kann nicht pro einzelnem Zahlungsempfänger (z.B. Händler) oder pro Kanal (entweder App oder Web-Schnittstelle) definiert werden. Die Betrugsrate, die bestimmt, ob sich ein PSP für die SCA-Ausnahme qualifiziert oder nicht, kann nicht nur für bestimmte Händler berechnet werden, d.h. wenn der Zahler eine Zahlung an einen bestimmten Händler leisten möchte und dieser bestimmte Händler eine Betrugsrate unter dem Grenzwert hat. Während der PSP (Acquirer) des Zahlungsempfängers vertraglich vereinbaren kann, die Überwachung seiner Transaktionsrisikoanalyse an einen gegebenen Händler 'outzusourcen' oder nur bestimmten vordefinierten Händlern erlauben kann, von den Vorteilen von dieser PSP-Ausnahme zu profitieren (basierend auf einer vertraglich vereinbarten geringen Betrugsrate), muss die Betrugsrate, welche einen bestimmten PSP für eine Ausnahme gemäß Artikel 18 geeignet macht, dennoch auf Basis der ausgeführten oder akquirierten Transaktionen vom PSP des Zahlungsempfängers berechnet werden und nicht ausgehend von den Transaktionen des Händlers.

2.2 Message Version 2

Um die Menge der zusätzlichen zahlungsfremden Daten zu handhaben und die Abwärtskompatibilität soweit wie möglich zu erhalten, hat sich EVO Payments dafür entschieden, seine EVO E-PAY Kartenschnittstelle über den zusätzlichen Parameter MsgVer zu versionieren. Die aktualisierte API basiert weiterhin auf Schlüssel-Wert-Paaren, aber setzt stark auf Base64-codierte JSON-Objekte zur Unterstützung der Lesbarkeit und Skript-Nutzung auf der Client-Seite.

2.2.1 Whitelisting von vertrauenswürdigen Begünstigten

Ein Karteninhaber dafür optieren, einen Händler zu einer Liste vertrauenswürdiger Begünstigter hinzuzufügen, die beim Kartenherausgeber geführt wird, um diesen speziellen Händler bei zukünftigen Zahlungen von der SCA auszunehmen. Das passiert normalerweise während einer Challenge des Karteninhabers, aber Karteninhaber können beispielsweise auch über ihre Banking-App eine Liste vertrauenswürdiger Begünstigter verwalten.

Händler können von einer Whitelist-Ausnahme profitieren, wenn diese angefragt ist und wenn nicht anderweitig eine Challenge des Karteninhabers gefordert ist.

> Beachten Sie bitte, dass die Whitelist-Funktion ab 3DS Version 2.2 und höher verfügbar ist. Derzeit unterstützten die Kartenherausgeber meistens 3DS 2.1.

2.2.2 Wiederkehrende Transaktionen

Wiederkehrende Transaktionen sind eine Reihe von Transaktionen, die nach einer Vereinbarung zwischen einem Karteninhaber und einem Händler verarbeitet werden, wobei der Karteninhaber Waren oder Dienstleistungen über einen Zeitraum und mit einer Anzahl getrennter Transaktionen mit dem gleichen Betrag erwirbt Die anfängliche Transaktion muss authentifiziert werden (d.h. vom Karteninhaber ausgelöste Transaktion (CIT)). Nachfolgende wiederkehrende Zahlungen liegen außerhalb vom Geltungsbereich der RTS SCA, da sie regelmäßig vom Händler ausgelöst werden (d.h. ohne dass der Kunde in einer Sitzung ist).

2.2.3 Transaktionen mit geringem Wert

Kartenherausgeber können Transaktionen von der SCA ausnehmen, sofern die folgenden Bedingungen erfüllt sind:

- der Zahlungsbetrag übersteigt nicht 30 Euro,
- der kumulierte Betrag vorheriger Zahlungstransaktionen ohne SCA übersteigt nicht 100 Euro,
- die Anzahl der vorherigen Zahlungstransaktionen ohne SCA übersteigt nicht fünf aufeinanderfolgende Zahlungstransaktionen.

Beachten Sie bitte, dass die Ausnahmen für geringen Wert angefragt werden müssen, um für einen reibungslosen Authentifizierungsablauf berücksichtigt zu werden.

2.2.4 Transaktionsrisikoanalyse (TRA)

Acquirer und Kartenherausgeber dürfen auf die SCA verzichten, sofern die gesamte Betrugsrate nicht höher als die Referenz-Betrugsrate für den Ausnahmengrenzwert (ETV) ist, der in folgender Tabelle angegeben ist und wobei die risikobasierte Beurteilung jeder einzelnen Transaktion als geringes Risiko angesehen werden kann.

ETV	Kartenbasierte Zahlungen
EUR 500	1 bps
EUR 250	6 bps
EUR 100	13 bps

2.2.5 One-Leg-Out-Transaktionen

One-Leg-Out-Transaktionen sind solche Transaktionen, wo sich entweder der Zahlungsdienstleister des Zahlers oder der Zahlungsdienstleister des Empfängers außerhalb der Europäischen Union befindet.
Zahlungsdienstleister im Kontext kartenbasierter Transaktionen und im Geiste der PSD2 sind regelmäßig Acquirer und Issuer.
Daher sind weder die Nationalität des Karteninhabers noch der Geschäftsort des Händlers für die Beurteilung relevant, ob eine Transaktionen infolge der Regel 'one-leg out' außerhalb des Geltungsbereiches liegt.

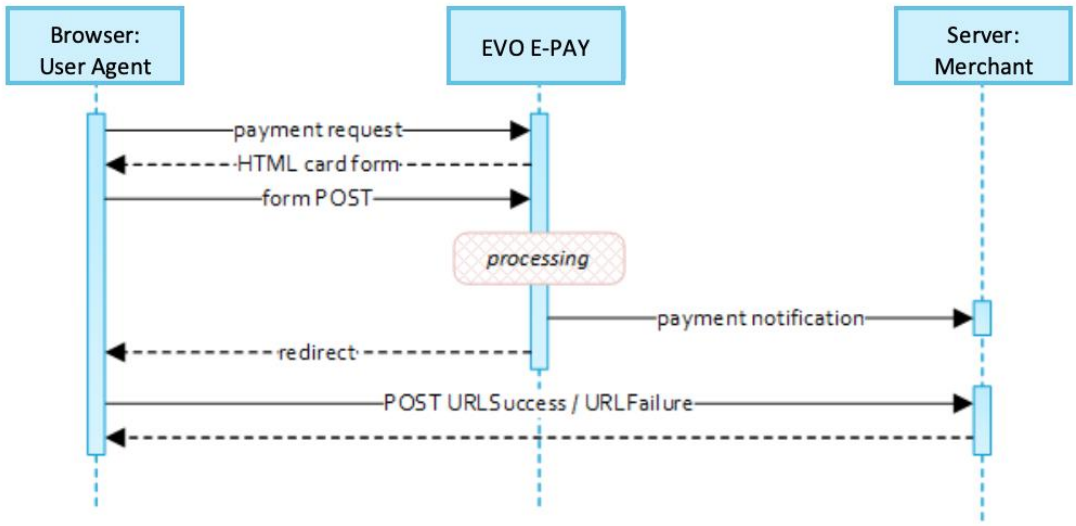
3. Integrations-Methoden

- EVO E-PAY Schnittstelle: per Formular (paySSL)

3.1 EVO E-PAY Schnittstelle: per Formular (paySSL)

Bei Kartenzahlungen über bei EVO Payments gehostete Formulare entfällt die Komplexität bei der Implemntierung von 3-D Secure vollständig – wir empfehlen jedoch dringend Übergabe aufgeführten JSON-Parameter um einen reibungungslosen Zahlungsablauf zu ermöglichen. Abgesehen von diese zusätzlichen Parametern gibt es aus Händlersicht keinen Unterschied zwischen Zahlungen mit und ohne 3DS.

3.1.1 Vereinfachtes Sequenz-Diagramm



3.1.2 Zahlungsanfrage

Zum Abruf eines EVO Payments Formulars für Kartenzahlungen übermitteln Sie bitte folgende Parameter über einen HTTP POST Aufruf an https://spg.evopayments.eu/pay/payssl.aspx.

Als allgemeine Regel ist es dringend empfohlen, bedingt erforderlich Datenelemente (C) stets zu übermitteln, um unnötige Reibereien und Ablehnungen zu vermeiden.

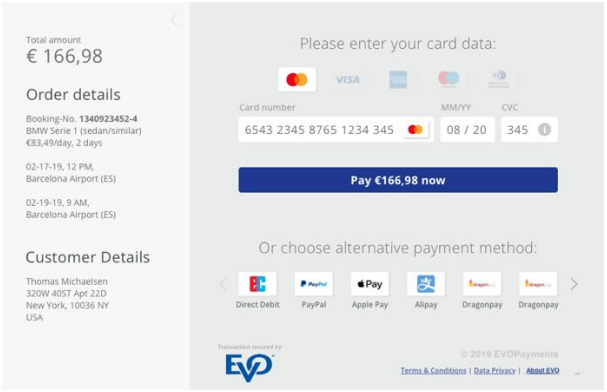
	Parameter	Format	Bedingung	Beschreibung
1	MerchantID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	MsgVer	ans..5	M	Message-Version. Zulässiger Wert: > 2.0
3	TransID	ans..64	M	Transaktions-ID des Händlers
4	RefNr	ans..20	M	Fremdbelegnummer/Transaktionsreferenz (zur Identifizierung des Vorgangs) Die folgenden Zeichen sind erlaubt: > Ziffern (0..9) > Großbuchstaben (A..Z) > Trennzeichen: Punkt (.), Bindestrich (-) und Schrägstrich (/)
5	Amount	n..10	M	Betrag in der kleinsten Währungseinheit (z.B. EUR Cent)
6	Currency	a3	M	Währungskürzel, drei Zeichen DIN / ISO 4217
7	Capture	ans..6	O	Bestimmt Art und Zeit der Zahlungsbuchung (d.h. Dual-Message-Systeme).

	Parameter	Format	Bedingung	Beschreibung
				Zulässige Werte: > `AUTO` = Buchung sofort nach der Autorisierung (Standardwert) > `MANUAL` = Buchung durch den Händler > `NUMBER` = Verzögerung bis zur Buchung in Stunden (ganze Zahl; 1 bis 696).
8	billingDescriptor	ans..22	O	Eine Bezeichnung, die auf dem Kontoauszug des Karteninhbaers gedruckt wird. Beachten Sie bitte auch die zusätzliche Hinweise an anderer Stelle für weitere Informationen über Regeln und Vorschriften.
9	OrderDesc	ans..768	O	Beschreibung der gekauften Waren, Einzelpreise etc.
10	AccVerify	a3	O	Indikator für Anforderung einer Kontoverifizierung (alias Nullwert-Authorisierung). Bei einer angeforderten Kontoverifizierung ist der übermittelte Betrag optional und wird für die tatsächliche Zahlungstransaktion ignoriert (z.B. Autorisierung). Zulässiger Wert: • `Yes`
11	threeDSConfig	JSON	O	Objekt, das Händler-, Acquirer- und Anmeldedaten für die 3DS-Authentifizierung festlegt. Übermittelte Werte überschreiben die für die `MerchantID` gespeicherte Konfiguration.
12	threeDSPolicy	JSON	O	Objekt, das Authentifizierungs-Richtlinien und Vorgaben für die Ausnahmenbehandlung festlegt.
13	priorAuthenticationInfo	JSON	O	Das Objekt Prior Transaction Authentication Information enthält optionale Informationen über eine Authentifizierung eines 3DS-Karteninhabers, die vor der aktuellen Transaktion erfolgt ist.
14	accountInfo	JSON	O	Das Objekt Kontoinformationen enthält optionale Informationen über das Kundenkonto beim Händler.
15	billToCustomer	JSON	C	Der Kunde, dem die Waren und / oder Dienstleistungen in Rechnung gestellt werden. Für EMV 3DS erforderlich, sofern nicht Markt- oder Regionalmandate die Übermittlung dieser Infomrationen beschränken.
16	shipToCustomer	JSON	C	Der Kunde, an den die Waren und / oder Dienstleistungen gesendet werden. Erforderlich falls von billToCustomer abweichend.
17	billingAddress	JSON	C	Rechnungsadresse. Für EMV 3DS erforderlich (falls verfügbar), sofern nicht Markt- oder Regionalmandate die Übermittlung dieser Infomrationen beschränken.
18	shippingAddress	JSON	C	Lieferadresse. Falls von billingAddress abweichend; für EMV 3DS erforderlich (falls verfügbar), sofern nicht Markt- oder Regionalmandate die Übermittlung dieser Infomrationen beschränken.
19	credentialOnFile	JSON	C	Objekt, das Art und Reihe von Transaktionen mittels Zahlungskonto-Zugangsdaten festlegt (z.B. Kontonummer oder Zahlungs-Token), die bei einem Händler für die Verarbeitung zukünftiger Einkäufe für einen Kunden gespeichert sind. Erforderlich falls zutreffend.
20	merchantRiskIndicator	JSON	O	Der Händler-Risikoindikator enthält optionale Informationen über den bestimmten Einkauf des Kunden. Falls `shippingAddress` nicht vorhanden ist, ist es dringend empfohlen, das Merkmal `shippingAddressIndicator` mit einem entsprechenden Wert wie `shipToBillingAddress`, `digitalGoods` oder `noShipment` auszufüllen.
21	URLNotify	an..256	M	Eine FQDN URL zur Übermittlung des finalen Zahlungsergebnisses (HTTP POST)
22	URLSuccess	an..256	M	Eine FQDN URL zur Weiterleitung des Kunden für den Fall, dass die Zahlung erfolgreich war (HTTP POST)

	Parameter	Format	Be-dingung	Beschreibung
23	URLFailure	an..256	M	Eine FQDN URL zur Weiterleitung des Kunden für den Fall, dass die Zahlung nicht erfolgreich war (HTTP POST)
24	userData	ans..1024	O	Base64-codierter benutzerdefinierter Wert, der in Antworten und Benachrichtigungen zurückgegeben wird
25	MAC	an64	M	Hash Message Authentication Code (HMAC) mit SHA-256-Algorithmus

➡ Allgemeine Hinweise zu **URLSuccess**, **URLFailure** und **URLNotify**:
> Wir empfehlen, den Parameter **response=encrypted** zu verwenden, um eine verschlüsselte Antwort von EVO E-PAY zu erhalten.
> Betrüger können das verschlüsselte **DATA**-Element kopieren, das an **URLFailure** gesendet wurde, und betrügerisch **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.

EVO E-PAY gibt in der Antwort ein HTML-Dokument zurück, welches das angeforderte Kartenzahlungsformular darstellt. Das Formular kann in die Checkout-Seite des Händlers integriert oder als selbständige Seite verwendet werden, auf die der Karteninhaber weitergeleitet wird.



Die Authentifizierung des Karteninhabers sowie die Zahlungsautorisierung erfolgen, nachdem der Karteninhaber aller erforderlichen Kartendetails eingegeben und das Formular an das EVO E-PAY übermittelt hat.

Hinweis: Falls Sie ein eigenes Zahlungsformular verwenden (Corporate Payment Page), achten Sie darauf, dass der Name des Karteninhabers auf dem Formular enthalten ist. Der Name des Karteninhabers wird auf den Paygate API-Parameter "CreditCardHolder" abgebildet. Das Feld Cardholder name darf keine Sonderzeichen enthalten und muss eine Mindestlänge von 2 Zeichen und eine Maximallänge von 45 Zeichen haben.

Wenn die Zahlung abgeschlossen ist, sendet das EVO E-PAY eine Benachrichtigung an den Server des Händlers (d.h. **URLNotify**) und leitet den Browser an **URLSuccess** beziehungsweise **URLFailure** weiter.

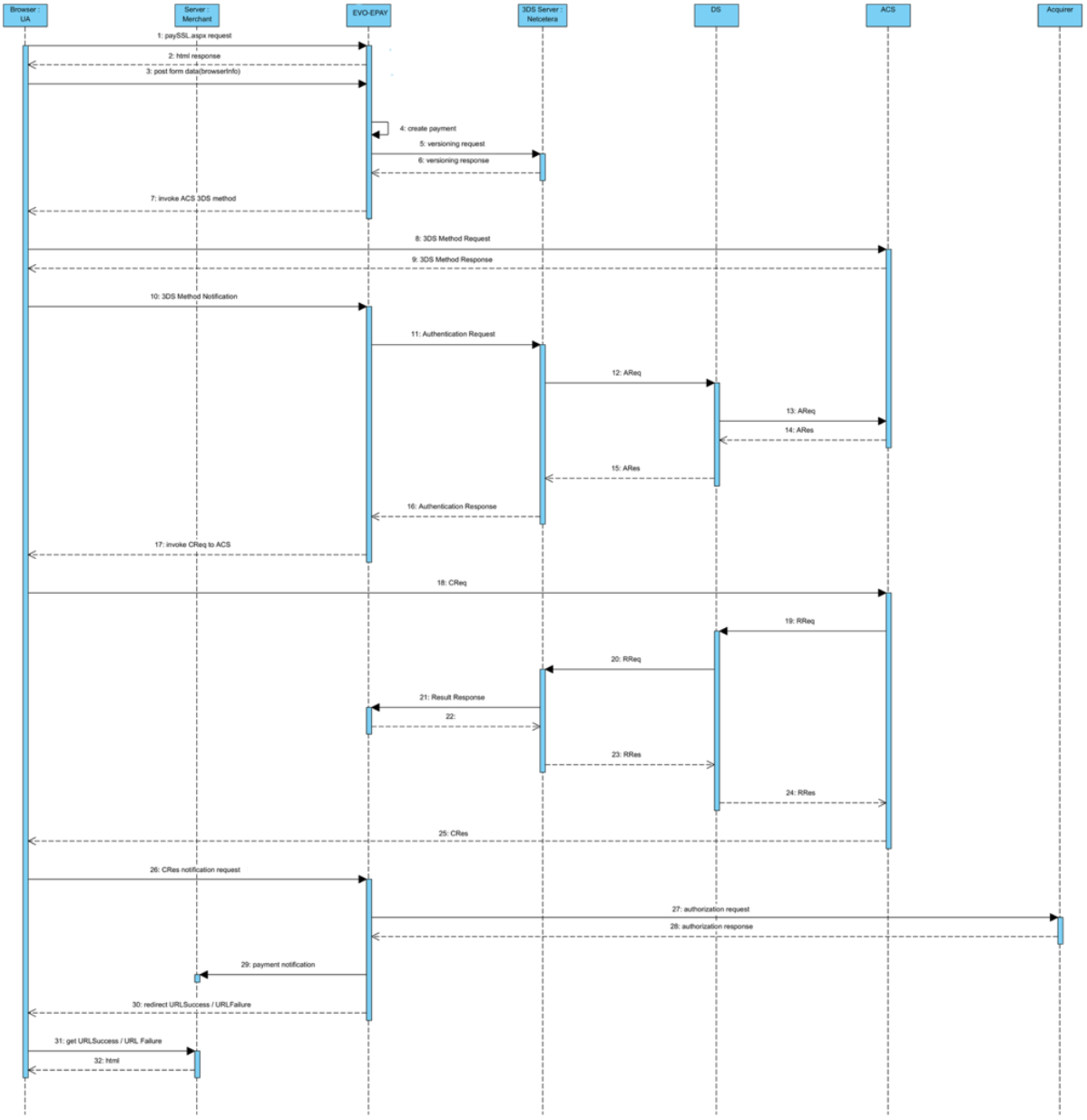
Die per Blowfish verschlüsselten Parameter laut folgender Tabelle werden per HTTP POST an URLNotify und URLSuccess/URLFailure übertragen.

➡ Bitte beachten Sie, dass der Aufruf der **URLSuccess** oder **URLFailure** bei einem Fallback zu 3-D Secure 1.0 mit **GET** stattfindet. Ihre Systeme sollten daher Parameter sowohl per **GET** als auch per **POST** entgegennehmen können.

3.1.3 HTTP POST an URLSuccess / URLFailure / URLNotify

Parameter	Format	Bedingung	Beschreibung
MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
MsgVer	ans..5	M	Message-Version. Zulässiger Wert: • `2.0`
PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
XID	ans64	M	Von EVO E-PAY vergebene ID für die einzelnen Operationen, die zu einer Zahlung durchgeführt werden
TransID	ans..64	M	Transaktions-ID des Händlers, die für jede Zahlung eindeutig sein muss
schemeReferenceID	ans..64	C	Spezifische Transaktions-ID des Kartenschemas, die für nachfolgende Zahlungen mit gespeicherten Zugangsdaten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist.
Status	a..20	M	Status der Transaction. Zulässige Werte: • `Authorized` • `OK` (Sale) • `FAILED` Im Fall von *nur-Authentifizierung* ist der *Status* entweder `OK` oder `FAILED`.
Description	ans..1024	M	Nähere Beschreibung des Codes
Code	n8	M	Antwortcode des EVO E Pay
card	JSON	M	Objekt der Kartendaten
ipInfo	JSON	C	Objekt mit IP-Informationen. Das Vorhandensein hängt von der Konfiguration des Händlers ab.
threeDSData	JSON	M	Objekt der Authentifizierungsdaten
resultsResponse	JSON	C	Falls der Authentifizierungsprozess eine Aufforderung für den Karteninhaber enthalten hat, werden zusätzliche Informationen über das Ergebnis der Aufforderung bereitgestellt
userData	ans..1024	C	Base64-codierter benutzerdefinierter Wert, der beim Aufruf angegeben wurde
MAC	an64	M	Hash Message Authentication Code (HMAC) mit SHA-256-Algorithmus

3.1.4 Erweitertes Sequenz-Diagramm



3.2 EVO E-PAY Schnittstelle: Per Server-zu-Server

Beachten Sie bitte, dass die Server-2-Server Integration **ausschließlich** für **Folgetransaktionen** mit einer **Pseudokartennummer** und entsprechenden **credentialOnFile** sowie **schemeReferenceID Parametern** relevant ist.
Intiale Transaktionen übergben Sie über die Formular Intergation (paySSL).

3.2.1 Überblick

Eine 3DS 2.0 Zahlungssequenz kann aus den folgenden verschiedenen Aktivitäten bestehen:

- Versionierung
    - Anfrage von ACS- und DS-Protokol-Version(en), die mit dem Kartenkontenbereich korrespondieren sowie einer optionalen 3DS Method URL

- 3DS Methode
  - Verbindet den Browser des Karteninhabers mit dem ACS des Issuers, um zusätzliche Browser-daten zu erhalten
- Authentifizierung
  - Übermittlung der Authentifizierungsanfrage an den ACS des Issuers
- Challenge
  - Challenge des Karteninhabers, falls angeordnet
- Autorisierung
  - Autorisierung der authentifizierten Transaktion beim Acquirer

Server-2-Server Sequenzdiagramm

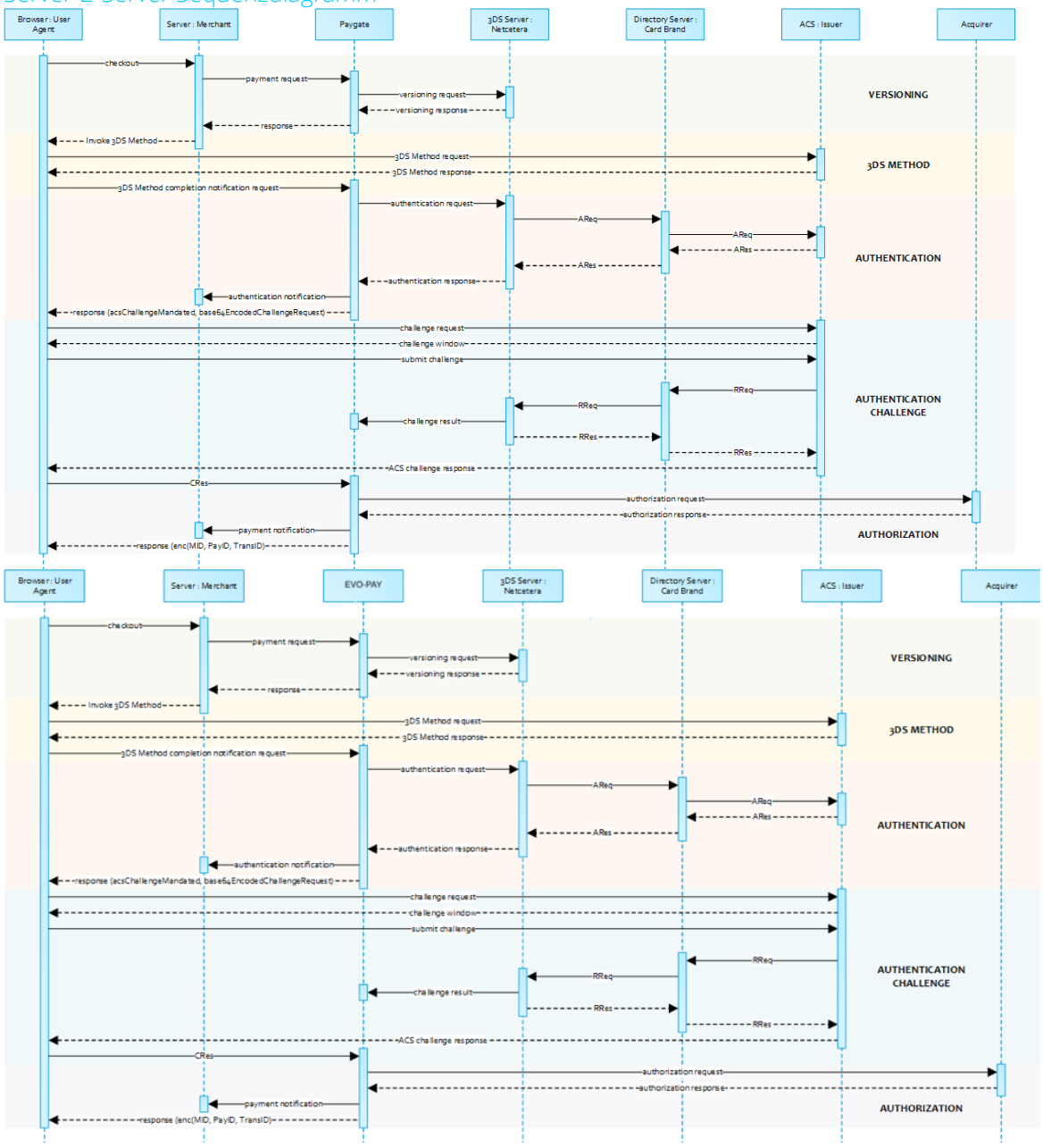
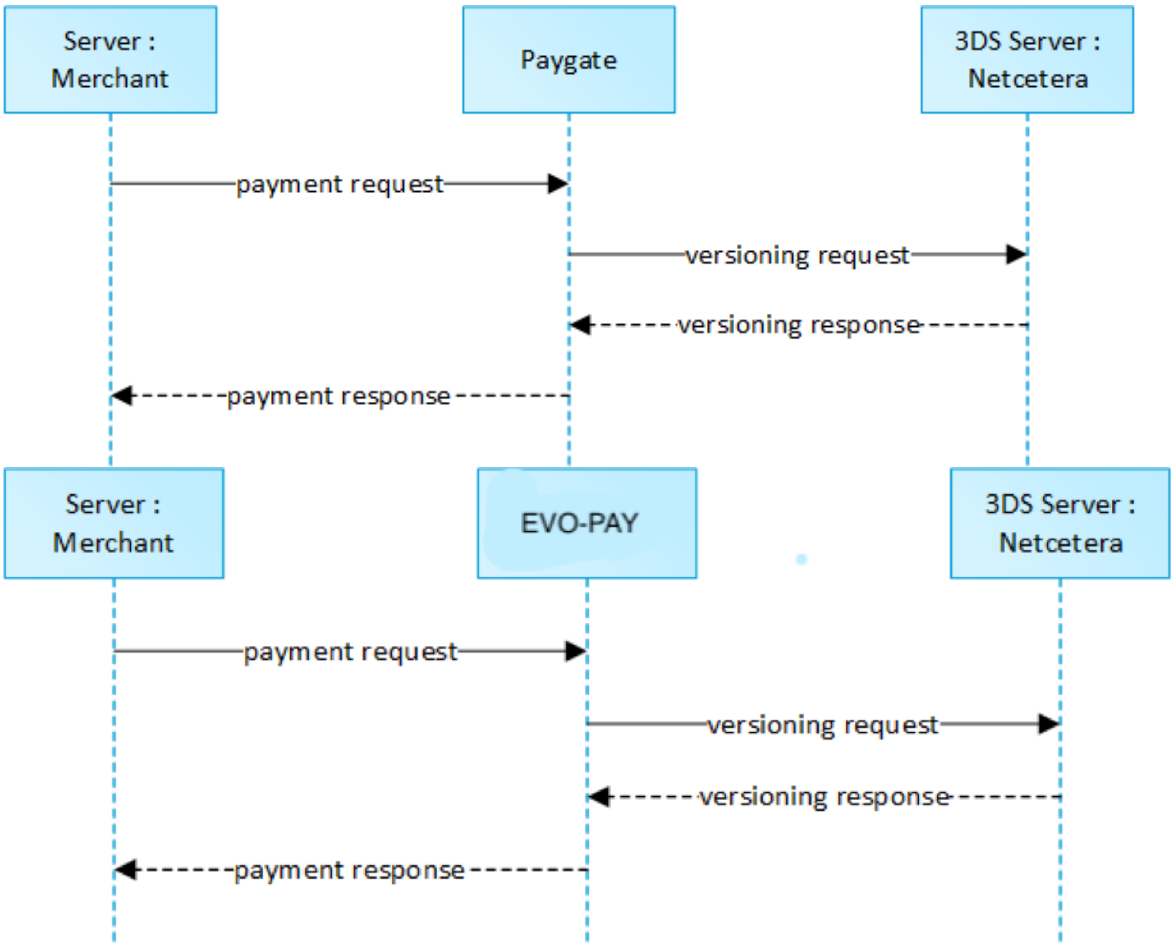


Figure 1 Server-2-Server Sequence Diagram

Beachten Sie bitte, dass die Kommunikation zwischen Client und Access Control Server (ACS) über iFrames implementiert ist. Daher kommen die Antworten in einem HTML-Subdokument an und Sie können entsprechende Event-Listener in Ihrem Root-Dokument einrichten.
Alternativ könnten Sie alleinig auf die asynchronen Benachrichtigungen an ihr Backend vertrauen. In jenen Fällen müssen Sie eventuell Methoden wie Long Polling, SSE oder Websockets zum Update des Clients in Betracht ziehen.

3.2.2 Initiierung der Zahlung

Die anfängliche Anfrage an EVO E-PAY ist unabhängig vom zugrundeliegenden 3DS-Protokoll gleich.



Um eine Server-zu-Server 3-D Secure Kartenzahlungssequenz zu starten, senden Sie bitte folgende Schlüssel-Wert-Paare an https://spg.evopayments.eu/pay/direct.aspx.

3.2.2.1 Aufruf-Elemente

Hinweis: Bei einer vom Händler initiierten, wiederkehrenden Zahlung sind die JSON-Objekte (außer credentialOnFile), die URLNotify und die TermURL keine Pflichtparameter, da kein 3-D Secure und auch keine Risikobewertung durch die kartenausgebende Bank stattfindet und das Ergebnis der Zahlungsanfrage direkt in der Response mitgeteilt wird.

	Parameter	Format	Bedingung	Beschreibung
1	MerchantID	ans..30	M	HändlerID, die von EVO Payments vergeben wird

	Parameter	Format	Bedingung	Beschreibung
2	MsgVer	ans..5	M	Message-Version. Zulässige Werte: • `2.0`
3	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein muss
4	RefNr	ans..20	M	Fremdbelegnummer/Transaktionsreferenz (zur Identifizierung des Vorgangs) Die folgenden Zeichen sind erlaubt: > Ziffern (0..9) > Großbuchstaben (A..Z) Trennzeichen: Punkt (.), Bindestrich (-) und Schrägstrich (/)
5	schemeReferenceID	ans..64	C	Kartensystemspezifische Transaktions-ID, die für nachfolgende Zahlungen mit hinterlegten Daten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist.
6	Amount	n..10	M	Betrag in der kleinsten Währungseinheit (z.B. EUR Cent)
7	Currency	a3	M	Währung, drei Zeichen DIN / ISO 4217
8	card	JSON	M	Kartendaten
9	Capture	ans..6		Bestimmt Art und Zeitpunkt des Zahlungsabschlusses (d.h. Dual-Nachrichtensysteme). Zulässige Werte: • `AUTO` = Abschluss sofort nach der Autorisierung (Standardwert) • `MANUAL` = Abschluss erfolgt durch den Händler • `<Number>` = Verzögerung in Stunden bis zum Abschluss (ganze Zahl; 1 bis 696)
10	channel	a..20	C	Gibt die Art der verwendeten Schnittstelle zur Initiierung der Transaktion an. Zulässige Werte: • `Browser` • `App` • `3RI` Wenn nicht angegeben, wird der Wert `Browser` verwendet.
11	billingDescriptor	ans..22	O	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.
12	OrderDesc	ans..768	O	Beschreibung der Bestellung
13	TermURL	ans..256	M	Im Falle des 3DS 1.0 Fallback: die URL, zu der der Kunde am Ende des 3DS 1.0 Authentifizierungsprozesses zurückgeleitet wird
14	AccVerify	a3	O	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`
15	threeDSConfig	JSON	O	Objekt, dass Händler, Acquirer und Anmeldedaten für die 3DS-Authentifizierung angibt. Wenn es übergeben wird, überschreiben die Werte die gespeicherten Daten für die `MerchantID`.
16	threeDSPolicy	JSON	O	Objekt, dass die Authentifizierungs-Richtlinien und Strategien zur Behandlung von Ausnahmen angibt
17	threeDSData	JSON	C	Objekt mit Details der Authentifizierungsdaten, falls die Authentifizierung durch Dritte oder durch den Händler durchgeführt wurde

	Parameter	Format	Bedingung	Beschreibung
18	priorAuthenticationInfo	JSON	O	Das Objekt Prior Transaction Authentication Information enthält optionale Informationen über eine 3DS-Authentifizierung eines Karteninhabers, die vor der aktuellen Transaktion erfolgt ist
19	browserInfo	JSON	C	Exakte Browserinformationen sind nötig, um eine optimierte Nutzererfahrung zu liefern. Erforderlich für 3DS 2.0 Transaktionen.
20	accountInfo	JSON	O	Die Kontoinformationen enthalten optionale Informationen über das Kundenkonto beim Händler
21	billToCustomer	JSON	C	Der Kunde, dem die Waren und / oder Dienstleistungen in Rechnung gestellt werden. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
22	shipToCustomer	JSON	C	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.
23	billingAddress	JSON	C	Rechnungsadresse. Erforderlich für 3DS 2.0 (falls verfügbar), sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
24	shippingAddress	JSON	C	Lieferadresse. Falls abweichend von billingAddress, erforderlich für 3DS 2.0 (falls verfügbar), sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.
25	credentialOnFile	JSON	C	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.
26	merchantRiskIndicator	JSON	O	Der Händler-Risikoindikator enthält optionale Informationen über den bestimmten Einkauf des Kunden
27	URLNotify	an..256	M	Die Händler-URL, die asynchrone Anfragen während des Authentifizierungsprozesses empfängt
28	userData	ans..1024	O	Base64-codierter individueller Wert, der in Antworten und Benachrichtigungen zurückgegeben wird
29	MAC	an64	M	Hash Message Authentication Code (HMAC) mit SHA-256 Algorithmus

3.2.2.2 Antwort-Elemente

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
3	XID	ans64	M	Von EVO E-PAY vergebene ID für die zu einer Zahlung ausgeführte Operation
4	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte
5	Code	n8	M	PaygateEVO E Pay-Antwortcode
6	Status	a..20	M	Status der Transaktion. Zulässige Werte: • `AUTHENTICATION_REQUEST` • `PENDING` • `FAILED`
7	Description	ans..1024	M	Textliche Beschreibung des Codes

	Parameter	Format	Be-dingung	Beschreibung
8	versioningData	JSON	M	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 3DS Methode enthalten, falls vom ACS unterstützt, sowie die DS Start- und End-Protokoll-Versionen, die den Kartenbereich unterstützen.
9	threeDSLegacy	JSON	M	Objekt, dass die erforderlichen Datenelemente für die Konstruktion der Anfrage zur Zahler-Authentifizierung im Falle eines Fallbacks auf 3DS 1.0 enthält.
10	userData	ans..1024	C	Base64-codierter individueller Wert wie in der Anfrage übergeben
11	MAC	an64	M	Message Authentication Code (HMAC) mit SHA-256 Algorithmus

Das Objekt `versioningData` gibt die EMV 3DS Protokoll-Versionen (d.h. 2.1.0 oder höher) an, die vom Access Control Server des Issuers unterstützt werden.

Wenn die entsprechenden Felder der Protokoll-Version NULL sind, bedeutet dies, dass der BIN-Bereich des Karten-Issuers nicht für 3DS 2.0 registriert ist und ein Fallback auf 3DS 1.0 für Transaktionen erforderlich ist, die unter den Geltungsbereich der PSD2 SCA fallen.

Achten Sie beim Zerlegen von `versioningData` bitte auch auf das Subelement `errorDetails`, das den Grund angibt, falls einige Felder nicht ausgefüllt sind (z.B. Ungültige Kontonumber des Karteninhabers übergeben, nicht verfügbare Kartenbereichsdaten, Fehler beo Codieren/Serialisieren der 3DS Methoden-Daten usw.)

```
{
  "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":                "eyJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMI-
joiaHR0cHM6Ly93d3cuY29tcHV0b3AtcGF5Z2F0ZS5jb20vY2JUaHJlZURTLmFzcHg_YWN0aW9uPW10aGROdG
ZuIiwidGhyZWVEU1NlcnZlclRyYW5zSUQiOiIxNGRkODQ0Yy1iMGZjLTRkZmUtODYzNS0zNjZmYmY0MzQ2OG-
MifQ==",
  "threeDSMethodData": {
    "threeDSMethodNotificationURL":                                  "https://www.computop-
paygate.com/https://spg.evopayments.eu/pay/cbThreeDS.aspx?action=mthdNtfn",
    "threeDSServerTransID": "14dd844c-b0fc-4dfe-8635-366fbf43468c"
  }
}
```

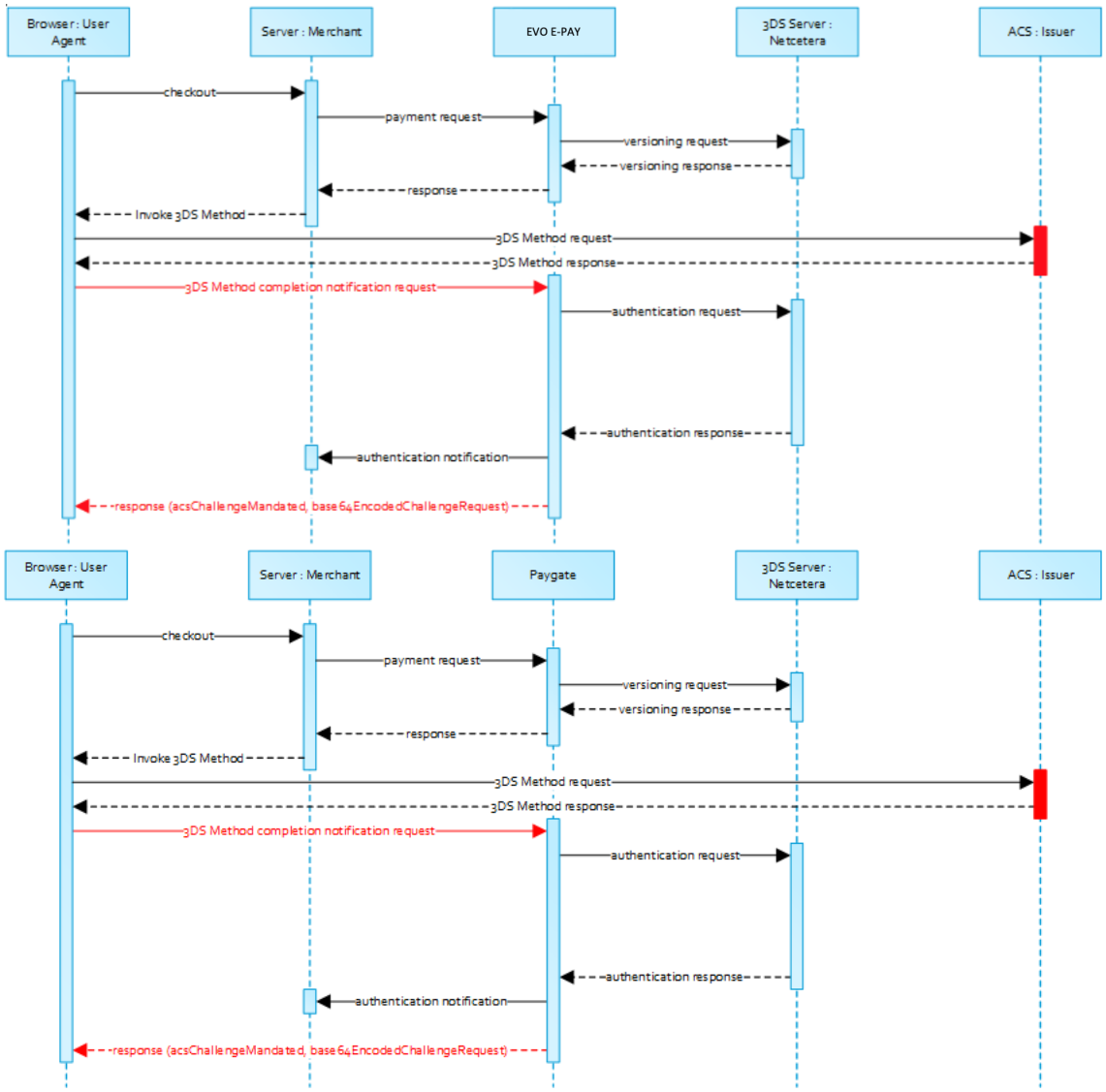
Code Block 1 versioningData

3.2.3 3DS Methode

Die 3DS Methode ermöglicht das Erfassen zusätzlicher Browserinformationen durch einen ACS vor Erhalt der Authensisierungsanfrage (AReq), um die Risikobeurteilung der Transaktion zu erleichtern. Die Unterstützung der 3DS Methode ist optional und liegt im Ermessen des Issuers.

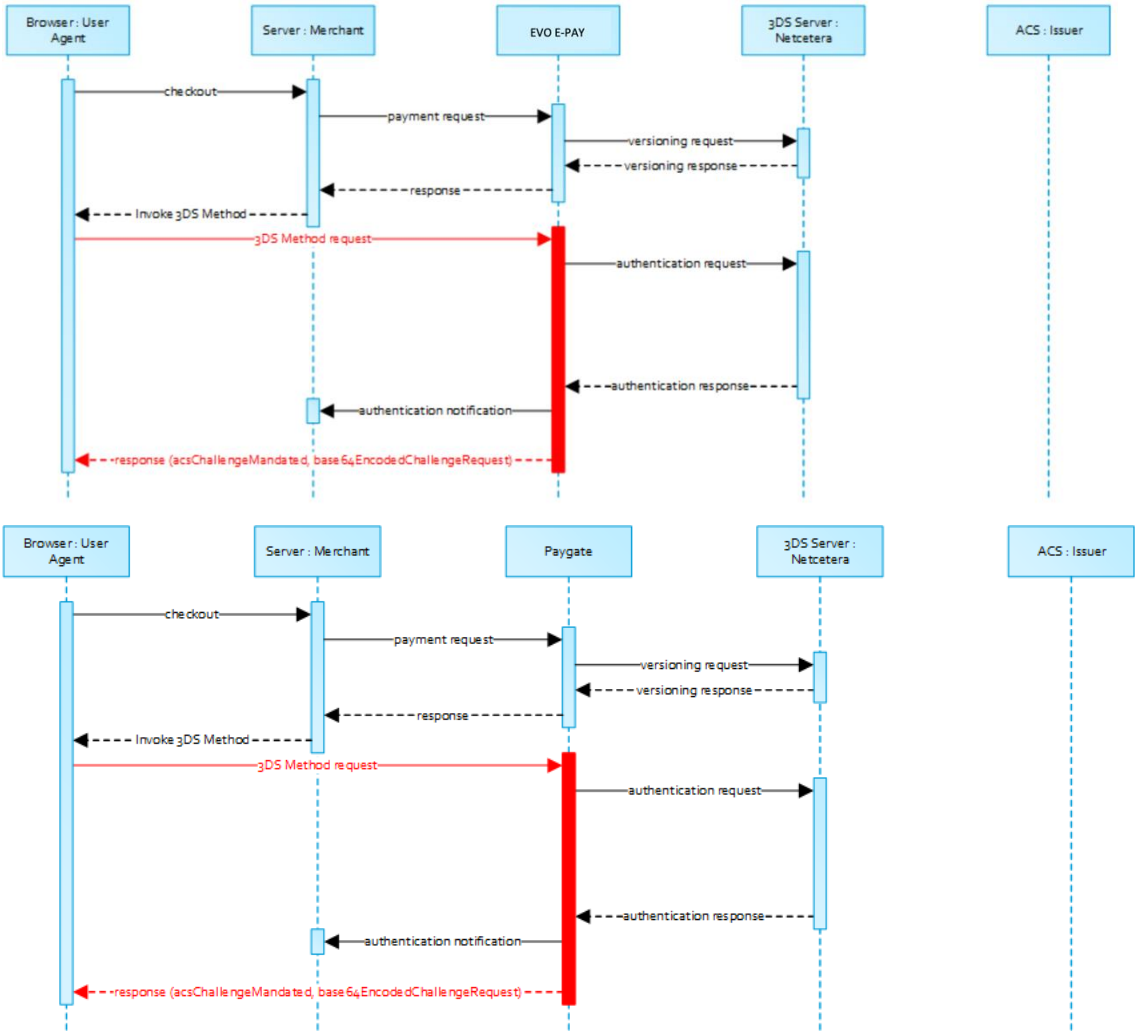
Das Objekt `versioningData` enthält einen Wert für `threeDSMethodURL`. Der Händler sollte die 3DS Methode über einen versteckten HTML-iFrame im Browser des Karteninhabers aufrufen und ein Formular mit einem Feld namens `threeDSMethodData` über HTTP POST an die ACS 3DS Methoden-URL senden.

3DS Methode: `threeDSMethodURL`



Beachten SIe bitte, dass die `threeDSMethodURL` von EVO E-PAY ausgefüllt wird, falls der Issuer die 3DS Methode nicht unterstützt. Der 3DS Methoden-Formular-Post wie unten dargestellt muss unabhängig davon ausgeführt werden, ob diese vom Issuer unterstützt wird. Das ist notwending, um die direkte Kommunikation zwischen dem Browser und EVO E-PAY im Falle einer angeordneten Challenge oder eines reibungslosen Ablaufs zu erleichtern.

3DS Method: Keine Issuer `threeDSMethodURL`




```
<form name="frm" method="POST" action="Rendering URL">
  <input type="hidden" name="threeDSMethodData"
value="eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6IjNhYzdjYWE3LWFhNDItMjY2My03OT-
FiLTJhYzA1YTU0MmM0YSIsInRocmVlRFNNZXRob2ROb3RpZmljYXRpb25VUkwi-
OiJ0aHJlZURTTWV0aG9kTm90aWZpY2F0aW9uVVJMIn0">
</form>
```

Code Block 2 3DS Method Form Post

Der ACS interagiert mit dem Browser des Karteninhabers über den HTML-iFrame und speichert dann die zutreffenden Werte mit der 3DS Server Transaction ID für die Verwendung, wenn eine nachfolgende Authentifizierungsnachricht empfangen wird, welche die gleiche 3DS Server Transaction ID enthält.

Netcetera 3DS Web SDK
Sie können nach eigenem Ermessen die Operationen `init3DSMethod` oder `createIframeAndInit3DSMethod` vom nca3DSWebSDK verwenden, um die 3DS Methode zu initialisieren. Bitte beachten Sie dazu das Integrations-Handbuch unter https://mpi.netcetera.com/3dsserver/doc/current/integration.html#Web_Service_API.

Nachdem die 3DS Methode abgeschlossen ist, weist der ACS den Browser des Karteninhabers über das iFrame-Antwortdokument an, `threeDSMethodData` als ein verstecktes Formularfeld an die 3DS Method Notification URL zu übermitteln.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8"/>
  <title>Identifying...</title>
</head>
<body>
<script>
  var                              tdsMethodNotificationValue                                =
'eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImUxYzFlYmViLTc0ZTgtND-
NiMi1iMzg1LTJlNjdkMWFhY2ZhMiJ9';

  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>
```

Code Block 3 ACS Response Document

```
<form name="frm" method="POST" action="3DS Method Notification URL">
  <input                    type="hidden"                    name="threeDSMethodData"
value="eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImUxYzFlYmViLTc0ZTgtND-
NiMi1iMzg1LTJlNjdkMWFhY2ZhMiJ9">
</form>
```

Code Block 4 3DS Method Notification Form

Beachten SIe bitte, dass die `threeDSMethodNotificationURL` wie sie in den Base64-codierten `threeDSMethodData` eingebettet ist, auf EVO E-PAY weist und nicht verändert werden darf. Die Händler-Benachrichtigung wird an die URLNotify geliefert, wie sie in der Originalanfrage übermittelt oder für die MerchantID in EVO E-PAY konfiguriert ist.

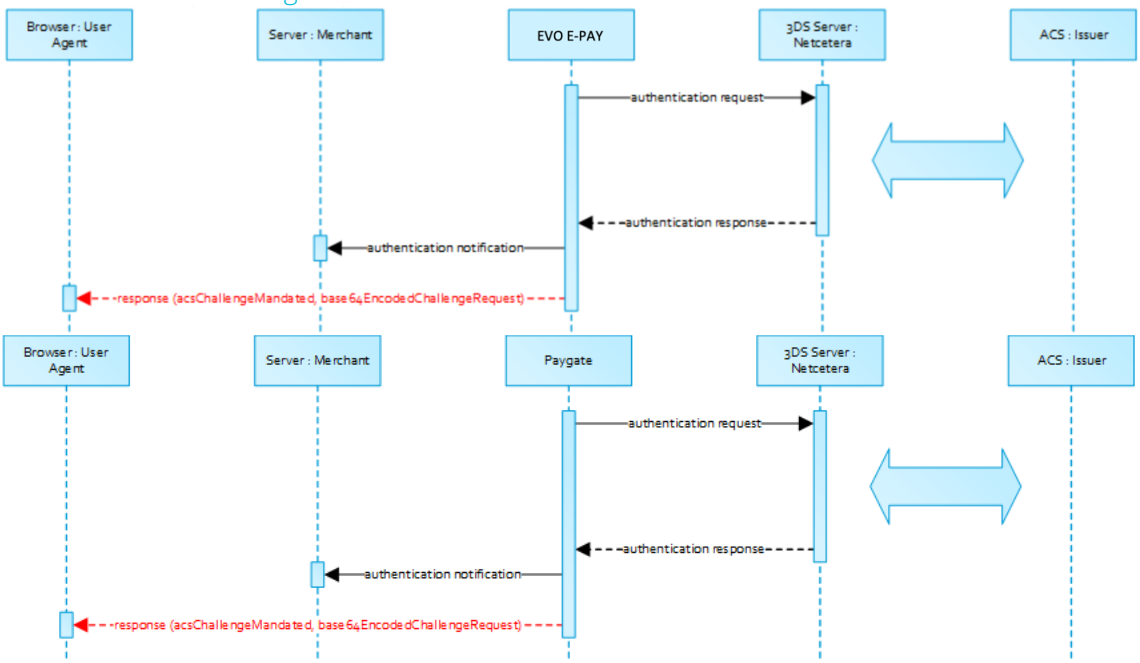
3.2.4 Authentifizierung

Wenn 3DS Methode vom ACS des Issuers unterstützt wird und vom Händler aufgerufen wurde, setzt EVO E-PAY automatisch mit der Authentifizierungsanfrage fort, nachdem die 3DS Methode abgeschlossen ist (d.h. 3DS Methoden-Benachrichtigung).

Das Ergebnis der Authentifizierung wird per HTTP POST an die `URLNotify` übertragen. Es kann anzeigen, dass der Karteninhaber authentifiziert worden ist oder dass eine weitere Interaktion des Karteninhabers (d.h. Challenge) für den Abschluss der Authentifizierung erforderlich ist.

Falls für den Karteninhaber eine Challenge angeordnet ist, überträgt EVO E-PAY ein JSON-Objekt im Body der HTTP Browser-Antwort mit den Elementen `acsChallengeMandated`, `challengeRequest`, `base64EncodedChallengeRequest` und `acsURL`. Anderenfalls setzt EVO E-PAY in einem reibungslosen Ablauf automatisch fort und antwortet dem Browser des Karteninhabers, sobald die Autorisierung abgeschlossen ist.

Kartzeninhaber-Challenge: Browser-Antwort



3.2.4.1 Browser Challenge-Antwort

Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	acsChallengeMandated	boolean	M	Zeigt an, ob für die Autorisierung der Transaktion eine Challenge erforderlich ist
2	challengeRequest	object	M	Objekt Challenge-Anfrage
3	base64EncodedChallengeRequest	string	M	Base64-codiertes Objekt Challenge-Anfrage
4	acsURL	string	M	Vollständige URL des ACS, die für das Posten der Challenge-Anfrage verwendet werden soll

Schema

```
{
  "$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",   "base64EncodedChal-
lengeRequest", "acsURL"],
  "additionalProperties": false
}
```

Code Block 5 Schema: Browser Challenge Response

Beispiel

```
{
  "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"
}
```

Code Block 6 Sample: Browser Challenge Response

3.2.4.2 Authentifizierungs-Benachrichtigung

Die Datenelemente der Authentifizierungs-Benachrichtigung stehen in folgender Tabelle.

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
3	TransID	ans..64	M	ransaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte
4	Code	n8	M	PaygateEVO E Pay-Antwortcode
5	authenticationResponse	JSON	M	Antwort-Objekt als Rückgabe zur Authentifizierungsanfrage beim ACS

	Parameter	Format	Bedingung	Beschreibung
6	MAC	an64	M	Hash Message Authentication Code (HMAC) mit SHA-256 Algorithmus

3.2.4.3 Browser Challenge

Wenn eine Challenge angeordnet wird (siehe `acsChallengeMandated`), erfolgt die Browser Challenge im Browser des Karteninhabers. Zum Erzeugen einer Challenge ist es erforderlich, den Wert `base64EncodedChallengeRequest` über ein HTML-iFrame an die ACS URL zu posten.

```
<form name="challengeRequestForm" method="post" action="acsChallengeURL">
  <input type="hidden" name="creq" value="ewogICAgInRocmVlRFNTZXJ2ZXJUcmFuc0lEIjogI-
jhhODgwZGMwLWQyZDItNDA2Ny1iY2IxLWIwOGQxNjkwYjI2ZSIsCiAgICAi-
YWNzVHJhbnNJRCI6ICJkN2MxZWU5OS05NDc4LTQ0YTYtYjFmMi0zOTFlMjljNmIzNDAiLAogI-
CAgIm1lc3NhZ2VUeXBlIjogIkNSZXEiLAogICAgIm1lc3NhZ2VWZXJzaW9uIjogIjIuMS4wIiwKICAgICJ-
jaGFsbGVuZ2VXaW5kb3dTaXplIjogIjAxIiwKICAgICJtZXNzYWdlRXh0ZW5zaW9uIjog-
WwoJCXsKCQkJIm5hbWUi-
OiAiZW12Y29tc2dleHRJbkNoYWxsZW5nZSIsCgkJCSJpZCI6ICJ0YzhRdG00NjVMbjFGWDBuWn-
ByQSIsCgkJCSJjcml0aWNhbGl0eUluZGljYXRvciI6IGZhbHNlLAoJCQki-
ZGF0YSI6ICJtZXNzYWdlRXh0ZW5zaW9uRGF0YUluQ2hhbGxlbmdlIgoJCX0KICAgIF0KfQ==">
</form>
```

Code Block 7 Challenge Request

Sie können die Operationen `init3DSChallengeRequest` oder `createIFrameAndInit3DSChallengeRequest` aus dem nca3DSWebSDK verwenden, um die Challenge-Nachricht an den Browser des Karteninhabers zu übermitteln.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <script src="nca-3ds-web-sdk.js" type="text/javascript"></script>
  <title>Init 3DS Challenge-Anfrage - Beispiel</title>
</head>
<body>
<!-- Dieses Beispiel zeigt, wie Challenge-Anfragen für verschiedenen Fenstergrößen
initialisiert werden. -->
<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">
  // All Container laden
  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-chal-
lenge-request', iFrameContainerFull);

  //   nca3DSWebSDK.createIFrameAndInit3DSChallengeRequest(acsUrl,   creqData,   chal-
lengeWindowSize, 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>
```

Code Block 8 Init 3DS Challenge Request - Example

Sobald die Challenge des Karteninhabers abgeschlossen, abgebrochen oder per Zeitüberschreitung beendet ist, weist der ACS den Browser an, die Ergebnisse per Post an die in der Challenge-Anfrage angegebene Benachrichtigungs-URL zu senden und eine Ergebnis-Anfrage (RReq) über den Directory Server an den 3DS Server zu senden.

Beachten Sie bitte, dass die in der Challenge-Anfrage übergebene Benachrichtigungs-URL auf EVO E-PAY zeigt und nicht verändert werden darf.

3.2.5 Autorisierung

Nachdem die erfolgreiche Authentifizierung des Karteninhabers oder der Nachweis der versuchten Authentifizierung/Verifizierung bereitgestellt ist, setzt EVO E-PAY die Zahlungsautorisierung automatisch fort.

Falls die Authentifizierung des Karteninhabers nicht erfolgreich war oder der Nachweise der versuchten Authentifizierung/Verifizierung nicht bereitgestellt werden kann, setzt EVO E-PAY nicht mit einer Autorisierungsanfrage fort.

In beiden Fällen liefert EVO E-PAY eine endgültige Benachrichtigung an die vom Händler angegebene `URL-Notify` mit den Datenelementen gemäß nachstehender Tabelle.

3.2.5.1 Zahlungs-Benachrichtigung

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	MsgVer	ans..5	M	Message-Version. Zulässige Werte: • `2.0`
3	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
4	XID	an32	M	Von EVO E-PAY vergebene ID für die zu einer Zahlung ausgeführte Operation
5	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte
6	schemeReferenceID	ans..64	C	Kartensystemspezifische Transaktions-ID, die für nachfolgende Zahlungen mit hinterlegten Daten, verzögerte Autorisierungen und Wiedereinreichungen erforderlich ist
7	TrxTime	an21	M	Transaction time stamp in format DD.MM.YYYY HH:mm:ssff.
8	Status	a..20	M	Status der Transaktion. Zulässige Werte: • `Authorized` • `OK` (Sale) • `PENDING` • `FAILED` Im Falle von **nur Authentifizierung** ist der **Status** entweder `OK` oder `FAILED`.
9	Description	ans..1024	M	Textliche Beschreibung des Codes
10	Code	n8	M	PaygateEVO E Pay-Antwortcode
11	card	JSON	M	Kartendaten
12	ipInfo	JSON	O	Objekt mit IP-Informationen
13	threeDSData	JSON	M	Authentifizierungsdaten
14	resultsResponse	JSON	C	Falls der Authentifizierungsprozess eine Challenge des Karteninhabers enthalten hat, werden zusätzliche Informationen über das Ergebnis der Challenge bereitgestellt
15	MAC	an64	M	Hash Message Authentication Code (HMAC) mit SHA-256 Algorithmus

3.2.5.2 Browser Zahlungs-Antwort

Zusätzlich werden nachstehende Datenelemente im JSON-Format im Body der HTTP-Antwort zum Browser des Karteninhabers übertragen. Beachten Sie bitte, dass die Datenelemente (d.h. `MID`, `Len`, `Data`) base64-codiert sind.

Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	MID	string	M	HändlerID, die von EVO Payments vergeben wird
2	Len	integer	M	Länge des unverschlüsselten Strings `Data`
3	Data	string	M	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
}
```

Händler sollten diese Datenelemente zur Entschlüsselung und für den Abgleich mit der Zahlungs-Benachrichtigung am ihren Server weiterleiten. Basierend auf dem Zahlungsergebnis kann der Händler-Server eine entsprechende Antwort an den Browser des Karteninhabers senden (z.B. Erfolgsseite).

Entschlüsseltes Objekt Data

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
3	TransID	ans..64	M	Transaktionsnummer des Händlers

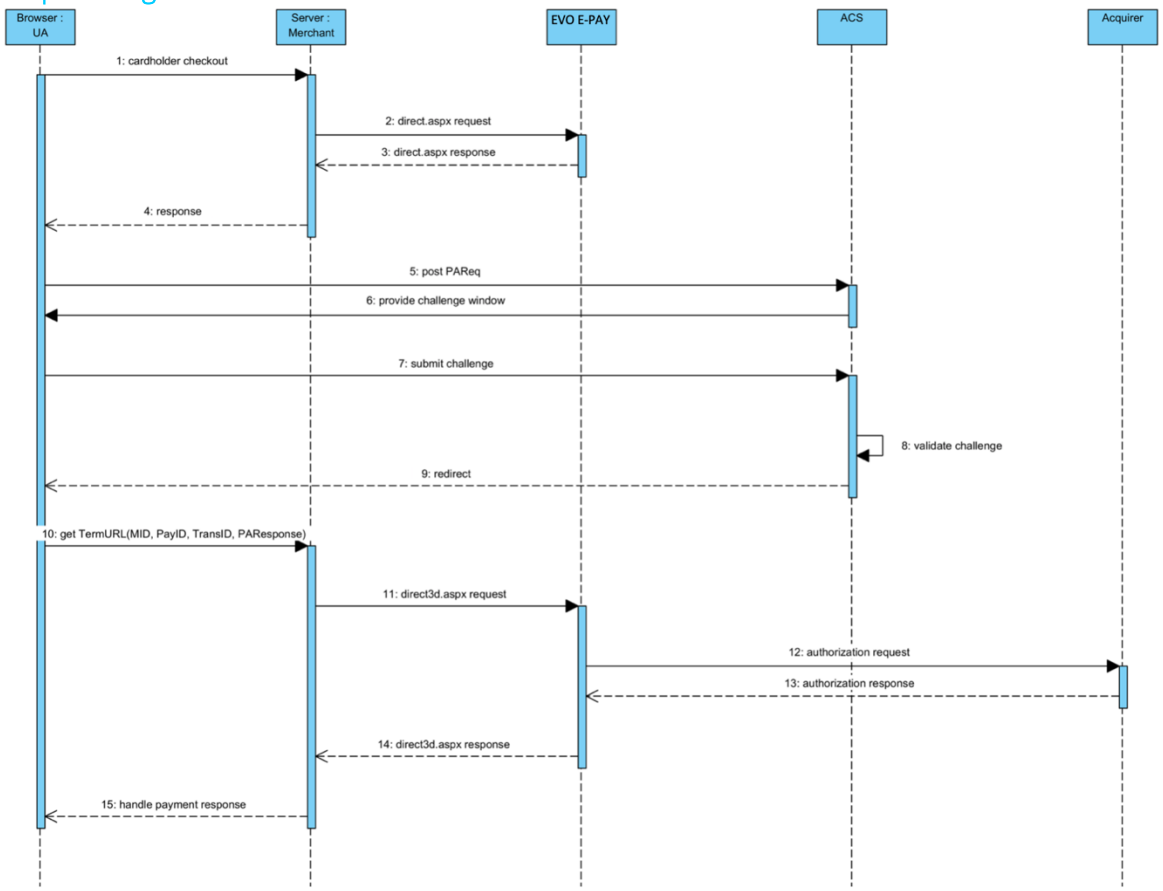
Beispiel für entschlüsseltes Objekt Data

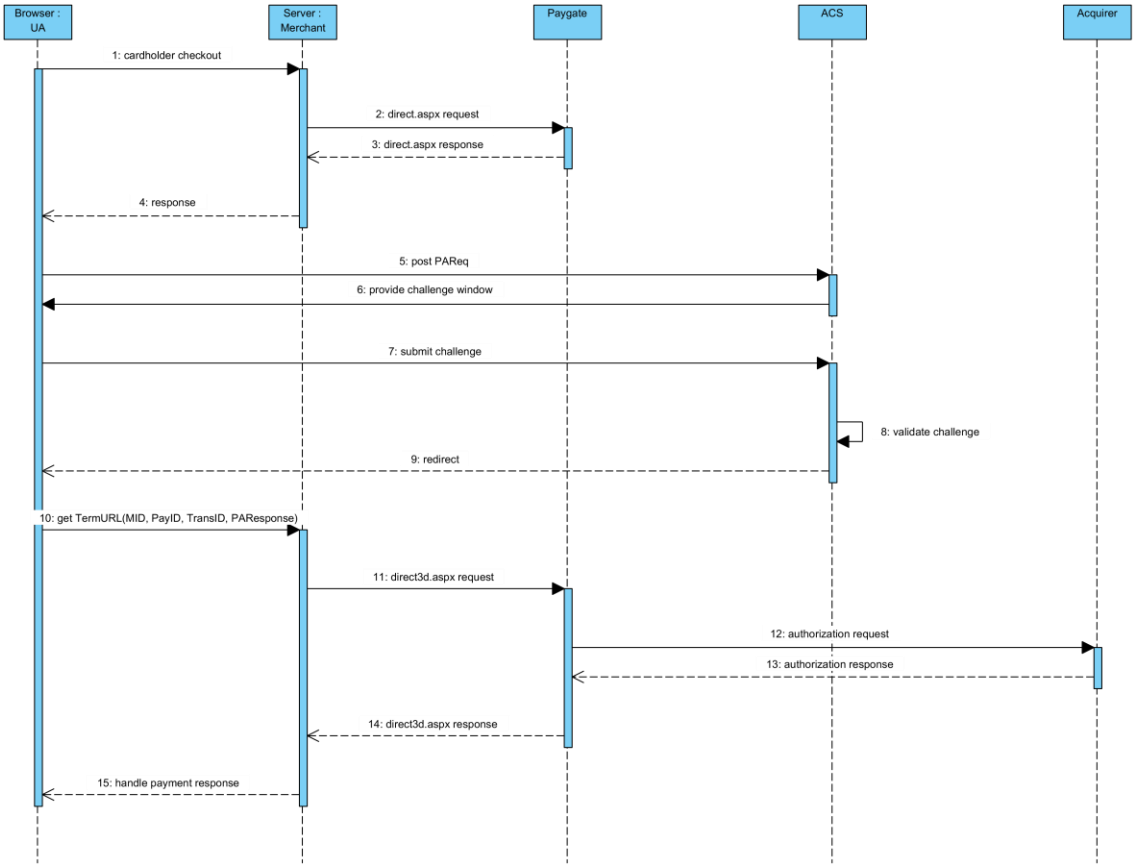
```
MID=YourMID&PayID=PayIDassignedbyPaygateEVOEPay&TransID=YourTransID
```

3.2.6 3DS 1.0 Fallback

Falls der Access Control Server (ACS) der Bank des Karteninhabers keine EMV 3DS Protokoll-Version unterstützt (d.h. 2.0 oder höher, siehe `acsStartProtocolVersion`), wird das Element `threeDSMethodData-Form` des Objekts versioningData in der Zahlungsantwort Null.

Sequenzdiagramm





3.2.6.1 3DS 1.0 Authentifizierung

Um eine 3DS 1.0 Authentifizierungsanfrage über den Browser des Karteninhabers auszuführen, ist es erforderlich, ein Formular aus den in `threeDSLegacy` bereitgestellten Datenelementen zu konstruieren und es an die `acsURL` zu posten.

Die an den ACS gesendeten Formularfelder sind in nachfolgender Tabelle aufgeführt:

	Formular-element	Beschreibung
1	PAReq	Ein konstruiertes, Base64-codiertes und komprimiertes Feld mit den Feldern der Payer Authentication Request Message. Der verwendete Kompressionsalgorithmus ist eine Kombination von LZ77- und Huffman-Codierung gemäß RFC 1951.
2	TermURL	Die Händler-URL, wohin der ACS den Karteninhaber nach Abschluss der Authentifizierung weiterleitet. Beachten Sie, dass EVO E-PAY die Felder `PayID`, `TransID` und `MID` im Anfrage-String zur Basis-URL hinzufügt. Bitte ändern Sie die TermURL nicht!
3	MD	Das Feld MD (d.h. Händlerdaten) kann beliebige Daten transportieren, die der Händler fpr die Fortsetzung der Sitzung benötigt. Beachten Sie bitte, dass dieses Feld im Formular vorhanden sein muss, auch wenn es nicht verwendet wird.

```
<html>
  <head>
    <script language=\"javascript\">
      <!--
        function sendpareq()
          {
            document.pareq_form.submit();
          }
      // -->
    </script>
  </head>

  <body onload="javascript:sendpareq();">
    <form           action="https://pit.3dsecure.net/VbVTestSuiteService/pit1/acsSer-
vice/paReq?summary=ZTIwOWMwYmEtNTVhOC00NDExLThkZDktYzllODk1NmZlNDQ0"    method="POST"
name="pareq_form">
      <input type="hidden" name="PaReq" value="eJxVUst22jAQ/RUfL7rpMZKFiQ0dK4dXgAVOT-
muSpjvVGsApfkSWA+TrK/Fo0t29M6M7M3cEt4di57yhav-
KqjF2/Q10Hy6ySebmJ3VV650Wu02hRSrGrSozdIzbuLYd0qxAnPzBrFXJYY-
tOIDTq5jN1aCIEioyzywkhILwh7gddnFD1JMVyv15HfYz2Xw8PwO75yuPTmp-
nWHAblSo6myrSg1B5G9jhYJD266jHWBXCgUqBYTPk4fR4+M+jdAz-
gEoRYG8zrXGRn+dFb/nzhdR1N+ccQXklIOsa-
kutjpyF5tWVQKt2fKt1PSBkv993sqqoW13VHYlAbA7Ix0gPrUWN0Trkkv+aLVnyvjkuZ6tD8vS8Tya7l/un-
BXt+n8ZAbAVIoZGbMSPaY4HjB4MuHQR9IKc4iMIOwX1KzXpnDLVtMfyU+BwA47syd-
zryfhiZHa4M8FCbM5kKY+U/DBKbjKfGD9PQQiAfC4zn1uFMG+vm+V06bad/Zi+rn6rrJ20xWt4P49h6fiqw8r
nxyo/8s74lQKwEuZyTXP6CQf/9kb8b1MvQ">
      <input type="hidden" name="TermUrl" value="http://localhost:40405/test/3DTer-
mURL.aspx?PayID=dc67820e15f049c9b6c1f0420729da8a&TransID=20180524-162741-084&MID=gus-
tav">
      <input type="hidden" name="MD" value="Optional merchant session data">
    </form>
  </body>
</html>
```

Code Block 9 Sample: PAReq form passed through the Cardholder to the ACS URL

Sobald die Authentifizierung abgeschlossen oder vom Karteninhaber abgebrochen worden ist, leitet der ACS den Karteninhaber über seinen Browser zur `TermURL` weiter, wie sie bei der anfänglichen Zahlungsanfrage angegeben ist.

Die Zahler-Authentifizierungs-Antwort (`PaRes`) wird mittels **HTTP POST** Methode übertragen, während `MID`, `PayID` und `TransID` im HTTP-Anfrage-String gesendet werden (d.h. **HTTP GET**).

Zur TermURL übertragene Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
3	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte

	Parameter	Format	Bedingung	Beschreibung
4	PARes	--	M	Die vom ACS gesendete PARes-Nachricht (Payer Authentication Response) in Reaktion auf die PAReq ungeachtet dessen, ob die Authentifizierung erfolgreich ist

3.2.6.2 Autorisierung

Um eine mit 3DS 1.0 authentifizierte Zahlung zu autorisieren, müssen Sie die Parameter der nachfolgenden Tabelle per POST an https://spg.evopayments.eu/pay/direct3d.aspx übermitteln.

Anfrage-Elemente

	Parameter	Format	Bedingung	Beschreibung
1	MerchantID	ans..30	M	HändlerID, die von EVO Payments vergeben wird
2	PayID	ans32	M	Von EVO E-PAY vergebene ID für die zu einer Zahlung ausgeführte Operation
3	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte
4	PAResponse	--	M	Die vom ACS gesendete PARes-Nachricht (Payer Authentication Response)

Antwort-Elemente

	Parameter	Format	Bedingung	Beschreibung
1	MID	ans..30	M	Merchant identifier assigned by EVO Payments.
2	PayID	ans32	M	Von EVO Payments vergebene ID für die Zahlung/Transaktion
3	XID	ans64	M	Von EVO E-PAY vergebene ID für die zu einer Zahlung ausgeführte Operation
4	TransID	ans..64	M	Transaktionsnummer des Händlers, die für jede Zahlung eindeutig sein sollte
5	Status	a..20	M	Status der Transaktion. Zulässige Werte: • `Authorized` • `OK` (Sale) • `FAILED`
6	Description	ans..1024	M	Textliche Beschreibung des Codes
7	Code	n8	M	PaygateEVO E-Pay-Antwortcode
8	card	JSON	C	Kartendaten
9	ipInfo	JSON	O	Objekt mit IP-Informationen
10	threeDSData	JSON	M	Authentifizierungsdaten

3.2.6.3 Felder der Payer Authentication Request

Das Nachrichtenfeld Payer Authentication Request (PAReq) ist ein von EVO Payments Merchant Server Plug-in (MPI) konstruiertes Datenelement.

Das MPI baut die XML PAReq im kanonischen Format gemäß DTD. Es führt den XML-Stream zu einem RFC1951-konformen Kompressor, der einen RFC1950-konformen Ausgangs-Stream erzeugt, der Base64-codiert wird.

Für Informatiosnzwecke sind die PAReq Datenelemente in der nahstehenden Tabelle aufgeführt.

PAReq

	Datenelement	Bedingung	Beschreibung
1	Message Version Number	M	Message-Versionsnummer, wie sie in der Verify Enrollment Response (VERes) erhalten wurde. Zulässige Werte: • `1.0.1` • `1.0.2`
2	Acquirer Bank Identification Number (BIN)	M	Dieses Feld muss zur verwendeten Acquirer-BIN bei der Verify Enrollment Request passen
3	Merchant Identifier (ID) Number	M	Dieses Feld muss zur verwendeten Merchant ID bei der Verify Enrollment Request passen. Dieses Feld muss auch zur vom Acquirer verwendeten Merchant ID gegenüber dem Kartennetzwerk für Autorisierungen und Abrechnung passen.
4	Merchant Name	M	Dieses Feld muss den Namen des Online-Händlers enthalten, bei dem der Karteninhaber einkauft. Die Maximallänge beträgt 25 Zeichen. Der Händlername muss dem eingereichten Namen für Autorisierung und Abrechnung entsprechen.
5	Merchant Country Code	M	Dieses Feld muss den dreistelligen Ländercode gemäß ISO 3166 enthalten
6	Merchant URL	M	Dieses Feld muss die vollständige URL der Händler-Webseite enthalten
7	Transaction Identifier	M	Eindeutige Transaktionsnummer des Händlers. Enthält einen 20 Byte großen statistischen eindeutigen Wert, der Base64-codiert ist und zu einem Ergebnis mit 28 Byte führt.
8	Purchase Date & Time	M	Datum und Uhrzeit des Kaufs in GMT im folgenden Format: JJJJMMTT HH:MM:SS.
9	Purchase Amount	M	Dieses Feld muss den Wert des Kaufs vom Karteninhaber enthalten. Es ist ein Wert mit bis zu 12 Stellen und ohne Nachkommastellen.
10	Purchase Currency	M	Der entsprechende dreistellige Währungscode gemäß ISO 4217 für die Transaktionswährung zwischen Karteninhaber und Händler muss verwendet werden.
11	Currency Exponent	M	Die kleinste Währungseinheit gemäß ISO 4217
12	Order Description	O	Kurze Beschreibung der gekauften Artikel durch den Händler. Die Maximalgröße beträgt 125 Zeichen, aber der Händler sollte beim Anlegen dieses Feldes die Eigenschaften vom Gerät des Karteninhabers berücksichtigen.
13	Recurring Payment Data	C	Ein Element Recur muss angegeben werden, wenn Händler und Karteninhaber wiederkehrende Zahlungen vereinbart haben
14	Installment Payment Data	C	Eine Ganzzahl größer als eins gibt die Maximalanzahl der erlaubten Autorisierungen für Ratenzahlungen an. Sie muss angegeben werden, wenn Händler und Karteninhaber Ratenzahlungen vereinbart haben.
15	Account Identifier	M	Der Inhalt dieses Feldes ist ein für den ACS nützlicher Daten-String; er darf die PAN nicht offenlagen und mit einem Algorithmus erzeugt werden, der glaubhaft eindeutige Werte erzeugt, selbst wenn dieselbe PAN präsentiert wird.
16	Card Expiry Date	M	Vom Karteninhaber an den Händler übermitteltes Ablaufdatum (JJMM)
17	Message Extension	O	Alle nötigen Daten zur Unterstützung der Anforderungen, die nicht anderweitig in der PAReq-Nachricht definiert sind, müssen in einer Nachrichten-Erweiterung transportiert werden

Wiederkehrende Zahlungsdaten

	Datenelement	Bedingung	Beschreibung
1	Recurring Frequency	M	Ganzzahl, welche die Mindestanzahl von Tagen zwischen Autorisierungen angibt
2	Recurring Expiry	M	Datum, nach dem keine weiteren Autorisierungen mehr erfolgen sollen. (Format JJJJMMTT).

4. JSON-Objekte

Beachten Sie bitte, dass alle JSON-Objekte **Base64**-codiert sein müssen.
EVO E-PAY validiert JSON-Objekte bei allen Requests, die den Parameter "MsgVer=2.0" enthalten. Dies passiert unabhängig davon, ob auch wirklich 3DSecure2 auf Ihrer MerchantID aktiv ist.
Bitte stellen Sie sicher, dass keine leeren Parameter oder Objekte gesendet werden. EVO E-PAY geht in solchen Fällen von einem Fehler aus und lehnt die Transaktion ab.

4.1 accountInfo

Die Kontoinformationen enthalten optionale Informationen über das Konto des Karteninhabers beim Händler.

Die Datenelemente in den Kontoinformationen des Karteninhabers, die zur Festlegung eines Zeitraums verwendet werden können entweder als *spezifisches Datum oder als ein Näherungsindikator* angegeben werden. 3DS Anforderer können beide Formate nutzen.

4.1.1 Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	accountIdentifier	string	O	Die Konto-ID des Karteninhabers in der Händler-Umgebung / Webseite (z.B. Kundennummer)
2	authenticationInformation	object	O	Dieses Element enthält optionale Informationen darüber, wie sich der Karteninhaber bei der Anmeldung zur Händler-Umgebung (z.B. Webseite) authentifiziert hat.
3	accountAgeIndicator	string	O	Zeitdauer, wie lange der Kunde das Zahlungsmittel / Zahlungskonto beim Händler hat. Zulässige Werte: • `guestCheckout` • `thisTransaction` • `lessThan30Days` • `from30To60Days` • `moreThan60Days`
4	accountChangeDate	string	O	Datum der letzten Änderung des Zahlungsmittels (Kontos) des Kunden beim Händler einschließlich Rechnungs- oder Lieferadresse, neues Zahlungskonto oder neu hinzugefügte(r) Benutzer (JJJJ-MM-TT).
5	accountChangeIndicator	string	O	Zeitdauer seit der letzten Änderung der Kontoinformationen des Kunden beim Händler einschließlich Rechnungs- oder Lieferadresse, neues Zahlungskonto oder neu hinzugefügte(r) Benutzer. Zulässige Werte: • `thisTransaction` • `lessThan30Days` • `from30To60Days` • `moreThan60Days`
6	accountCreationDate	string	O	Datum, an dem der Kunde das Konto beim Händler eröffnet hat im Format JJJJ-MM-TT
7	passwordChangeDate	string	O	Datum der letzten Kennwortänderung oder des Rücksetzens des Kundenkontos beim Händler im Format JJJJ-MM-TT.
8	passwordChangeDateIndicator	string	O	Gibt die Zeitdauer seit der Kennwortänderung oder seit dem Rücksetzen des Kundenkontos an. Zulässige Werte: • `thisTransaction` • `lessThan30Days` • `from30To60Days` • `moreThan60Days` • `noChange`
9	nbrOfPurchases	integer	O	Anzahl der Käufe in den letzten 6 Monaten
10	addCardAttemptsDay	integer	O	Anzahl der Versuche zum Hinzufügen einer Karte in den letzten 24 Stunden
11	nbrTransactionsDay	integer	O	Anzahl der Transaktionen (erfolgreich und abgebrochen) in den letzten 24 Stunden
12	nbrTransactionsYear	integer	O	Anzahl der Transaktionen (erfolgreich und abgebrochen) im letzten Jahr
13	paymentAccountAge	string	O	Datum, an dem das Zahlungskonto im Kundenkonto registriert worden ist, im Format JJJJ-MM-TT
14	paymentAccountAgeIndicator	string	O	Gibt die Zeitdauer an, wie lange das Zahlungskonto im Kundenkonto registriert ist.

	Parameter	Format	Bedingung	Beschreibung
				Zulässige Werte: • `thisTransaction` • `lessThan30Days` • `from30To60Days` • `moreThan60Days` • `guestCheckout`
15	shipAddressUsageDate	string	O	Datum, wann die für diese Transaktion angegebene Lieferadresse erstmalig verwendet wurde, im Format JJJJ-MM-TT
16	shipAddressUsageIndicator	string	O	Gibt an, wann die für diese Transaktion angegebene Lieferadresse erstmalig verwendet wurde. Zulässige Werte: • `thisTransaction` • `lessThan30Days` • `from30To60Days` • `moreThan60Days`
17	suspiciousAccActivity	boolean	O	Gibt an, ob der Händler in dem Kundenkonto verdächtige Aktivitäten (einschließlich früheren Betrugs) festgestellt hat

4.1.1.1 authenticationInformation

	Parameter	Format	Bedingung	Beschreibung
1	authenticationData	string	C	Dieses Datenelement kann spezielle Beglaubigungsdaten der Authentifizierung wie FIDO enthalten, falls zutreffend
2	authenticationMethod	string	M	Dieses Datenelement gibt den vom Karteninhaber zur Authentifizierung beim Händler verwendeten Mechanismus an. Zulässige Werte: • `guest` • `merchantCredentials` • `federatedID` • `issuerCredentials` • `thirdPartyAuthentication` • `FIDO` • `signedFIDO` • `SRCassuranceData`
3	authenticationTimestamp	string	M	Datum und Uhrzeit (siehe RFC 3339) der Authentifizierung des Karteninhabers in **UTC**. JJJJ-MM-TT**T**HH:MM:SS+00:00

4.1.2 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/accountInfo.json",
  "title": "accountInfo",
  "description": "Kundenkonto-Informationen",
  "type": "object",
  "properties": {
    "accountIdentifier": {
      "type": "string",
      "maxLength": 64
    },
    "authenticationInformation": {
      "type": "object",
      "properties": {
        "authenticationData": {
          "type": "string",
          "maxLength": 20000
        },
        "authenticationMethod": {
          "type": "string",
          "enum": ["guest", "merchantCredentials", "federatedID", "issuerCredentials",
"thirdPartyAuthentication", "FIDO", "signedFIDO", "SRCassuranceData"]
        },
        "authenticationTimestamp": {
          "type": "string",
          "format": "date-time"
        }
      },
      "required": ["authenticationMethod", "authenticationTimestamp"],
      "additionalProperties": false
    },
    "accountAgeIndicator": {
      "type": "string",
      "enum": ["guestCheckout", "thisTransaction", "lessThan30Days", "from30To60Days",
"moreThan60Days"],
      "description": "Zeitdauer, wie lange der Kunde das Konto beim Händler hat."
    },
    "accountChangeDate": {
      "type": "string",
      "format": "full-date",
      "description": "JJJJ-MM-TT"
    },
    "accountChangeIndicator": {
      "type": "string",
      "enum":       ["thisTransaction",       "lessThan30Days",       "from30To60Days",
"moreThan60Days"],
      "description": "Zeitdauer seit der letzten Änderung der Kundenkonto-Informatio-
nen."
    },
    "accountCreationDate": {
      "type": "string",
      "format": "full-date",
      "description": "JJJJ-MM-TT"
    },
    "passwordChangeDate": {
      "type": "string",
      "format": "full-date",
```

```
      "description": "JJJJ-MM-TT"
    },
    "passwordChangeDateIndicator": {
      "type": "string",
      "enum": ["noChange", "thisTransaction", "lessThan30Days", "from30To60Days",
"moreThan60Days"],
      "description": "Gibt die Zeitdauer seit der Kennwortänderung oder seit dem Rück-
setzen des Kundenkontos an."
    },
    "nbrOfPurchases": {
      "type": "integer",
      "maximum": 9999,
      "description": "Anzahl der Einkäufe in den letzten 6 Monaten."
    },
    "addCardAttemptsDay": {
      "type": "integer",
      "maximum": 999,
      "description": "Anzahl der Versuche zum Hinzufügen einer Karte in den letzten 24
Stunden."
    },
    "nbrTransactionsDay": {
      "type": "integer",
      "maximum": 999,
      "description": "Anzahl der Transaktionen (erfolgreich und abgebrochen) in den
letzten 24 Stunden."
    },
    "nbrTransactionsYear": {
      "type": "integer",
      "maximum": 999,
      "description": "Anzahl der Transaktionen (erfolgreich und abgebrochen) im letzten
Jahr."
    },
    "paymentAccountAge": {
      "type": "string",
      "format": "full-date",
      "description": "Datum, an dem das Zahlungskonto im Kundenkonto registriert worden
ist, im Format JJJJ-MM-TT."
    },
    "paymentAccountAgeIndicator": {
      "type": "string",
      "enum": ["guestCheckout", "thisTransaction", "lessThan30Days", "from30To60Days",
"moreThan60Days"],
      "description": "Gibt die Zeitdauer an, wie lange das zahlungskonto im Kundenkonto
registriert ist."
    },
    "shipAddressUsageDate": {
      "type": "string",
      "format": "full-date",
      "description": "Datum, wann die für diese Transaktion angegebene Lieferadresse
erstmalig verwendet wurde, im Format JJJJ-MM-TT."
    },
    "shipAddressUsageIndicator": {
      "type": "string",
      "enum": ["thisTransaction", "lessThan30Days", "from30To60Days",
"moreThan60Days"],
      "description": "Gibt an, wann die für diese Transaktion angegebene Lieferadresse
erstmalig verwendet wurde."
```

```
    },
    "suspiciousAccActivity": {
      "type": "boolean",
      "description": "Gibt an, ob der Händler in dem Kundenkonto verdächtige Aktivi-
täten (einschließlich früheren Betrugs) festgestellt hat."
    }
  },
  "additionalProperties": false
}
```

4.1.3 Beispiel

```
{
  "accountIdentifier": "joe.bloggs@acme.com"
  "authenticationInformation": {
    "authenticationMethod": "merchantCredentials",
    "authenticationTimestamp": "2021-10-05T04:36:18+00:00"
  },
  "accountAgeIndicator": "moreThan60Days",
  "accountChangeDate": "2019-01-23",
  "accountChangeIndicator": "from30To60Days",
  "accountCreationDate": "2016-01-01",
  "passwordChangeDate": "2018-06-08",
  "passwordChangeDateIndicator": "lessThan30Days",
  "nbrOfPurchases": 4,
  "addCardAttemptsDay": 0,
  "nbrTransactionsDay": 0,
  "nbrTransactionsYear": 5,
  "paymentAccountAge": "2018-03-20",
  "paymentAccountAgeIndicator": "thisTransaction",
  "shipAddressUsageDate": "2017-10-14",
  "shipAddressUsageIndicator": "moreThan60Days",
  "suspiciousAccActivity": true
}
```

4.2 address

4.2.1 Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	city	string	C	Stadt. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).
2	country	object	C	Alpha-3 Ländercode gemäß ISO 3166-1:2013. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).
3	addressLine1	object	C	Erste Zeile der Straßenadresse. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).

	Parameter	Format	Bedingung	Beschreibung
4	addressLine2	string	C	Zweite Zeile der Straßenadresse (z.B. Apartment, Suite, Etage, Postfach usw.). Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).
5	addressLine3	string	C	Dritte Zeile der Straßenadresse. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).
6	postalCode	string	C	PLZ oder andere Postleitzahl. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).
7	state	string	C	Alpha-2 Code des Bundesstaates oder der Provinz gemäß ISO 3166-2. Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken oder State in diesem Land nichtzutreffend ist. Für Versanddetails kann dieses Datenelement nicht verfügbar sein (z.B. digitale Güter).

4.2.1.1 addressLine1

	Parameter	Format	Bedingung	Beschreibung
1	street	string	M	Straßenname
2	streetNumber	string	C	Hausnummer

4.2.2 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/address.json",
  "title": "address",
  "description": "Adresse",
  "type": "object",
  "properties": {
    "city": {
      "type": "string"
    },
    "country": {
      "type": "object",
      "properties": {
        "countryName": {
          "type": "string",
          "description": "Name des Landes."
        },
        "countryA2": {
          "type": "string",
          "description": "ISO-3166 Alpha-2 Code."
        },
        "countryA3": {
          "type": "string",
          "description": "ISO 3166-1:2013 Alpha-3"
        },
        "countryNumber": {
          "type": "string",
          "description": "ISO-3166 numerischer Code."
        }
      },
      "required": ["countryA3"],
      "additionalProperties": false
    },
    "addressLine1": {
      "type": "object",
      "properties": {
        "street": {
          "type": "string"
        },
        "streetNumber": {
          "type": "string"
        }
      },
      "required": ["street"],
      "additionalProperties": false
    },
    "addressLine2": {
      "type": "string"
    },
    "addressLine3": {
      "type": "string"
    },
    "postalCode": {
      "type": "string"
    },
    "state": {
      "type": "string",
```

```
        "minLength": 2,
        "maxLength": 2,
        "description": "Alpha-2 Code des Bundesstaates oder der Provinz gemäß ISO 3166-
2:2013, sofern zutreffend"
      }
    },
    "required": ["country", "addressLine1", "postalCode"],
    "additionalProperties": false
  }
```

4.2.3 Beispiel

```
{
  "city": "New York",
  "country": {
    "countryA3": "USA"
  },
  "addressLine1": {
    "street": "Park Avenue",
    "streetNumber": "270"
  },
  "postalCode": "10017-2070",
  "state": "NY"
}
```

4.3 card

- card:request
- card:response

4.3.1 card:request

4.3.1.1 Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	securityCode	string	O	Kartenprüfnummer
2	expiryDate	string	M	Kartenverfallsdarum im Format JJJJMM.
3	startDate	string	C	Kartenanfangsdatum im Format JJJJMM (nur für einige Debitkarten in GB zutreffend)
4	cardholder-Name	string	M	Name des Karteninhabers, wie auf der Karte angegeben
5	issueNumber	string	C	Ausgabenummer der Karte (nur für einige Debitkarten in GB zutreffend)
6	number	string	M	Pseudokartennummer (PKN)/ Karten-Token
7	brand	string	M	Kartennetzwerk

4.3.1.2 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id":       "https://www.computop-paygate.com/https://spg.evopayments.eu/pay/schemas/card.json",
  "title": "card",
  "description": "Karteninformationen",
  "type": "object",
  "properties": {
    "securityCode": {
      "type": "string",
      "minLength": 3,
      "maxLength": 4
    },
    "expiryDate": {
      "type": "string",
      "description": "JJJJMM",
      "minLength": 6,
      "maxLength": 6
    },
    "startDate": {
      "type": "string",
      "description": "JJJJMM (nur für einige Debitkarten in GB zutreffend)",
      "minLength": 6,
      "maxLength": 6
    },
    "cardholderName": {
      "type": "string",
      "maxLength" : 45,
      "minLength" : 2,
      "description" : "Name des Karteninhabers, wie auf der Karte angegeben. Alphanumerische Sonderzeichen gemäß EMV Book 4, “Appendix B”."
    },
    "issueNumber" : {
      "type" : "string",
      "maxLength" : 2,
      "minLength" : 1,
      "description" : "Nur für einige Debitkarten in GB zutreffend"
    },
    "number": {
      "type" : "string",
      "maxLength" : 19,
      "minLength" : 12
    },
    "brand": {
      "type": "string",
      "enum": [
        "MasterCard",
        "VISA",
        "AMEX",
        "DINERS",
        "CBN",
        "JCB",
        "Dankort",
        "Maestro",
        "Cartes Bancaires",
        "DISCOVER",
        "Bancontact",
```

```
          "Hipercard",
          "Elo",
          "Aura",
          "Carte 4Etoiles",
          "AirPlus",
          "CUP",
          "NARANJA",
          "SHOPPING",
          "CABAL",
          "ARGENCARD",
          "CENCOSUD",
          "KOOKMIN",
          "KEB",
          "BC",
          "SHINHAN",
          "SAMSUNG",
          "HYUNDAI",
          "LOTTE",
          "1euro",
          "echequevacances",
          "cofidis3xcb",
          "cofidis4xcb",
          "facilypay-3x",
          "facilypay-3xsansfrais",
          "facilypay-4x",
          "facilypay-4xsansfrais",
          "RuPay"
        ]
      }
    },
    "required": ["expiryDate", "number", "brand"],
    "additionalProperties": false
}
```

4.3.1.3 Beispiel

```
{
  "securityCode": "569",
  "expiryDate": "202208",
  "cardholderName": "William Thomas",
  "number": "4186665161011901",
  "brand": "VISA"
}
```

4.3.2 card:response

4.3.2.1 Datenelemente

	Parameter	Format	Bedingung	Beschreibung
1	cardholderName	string	C	Das Vorhandensein hängt von der Händler-Konfiguration ab. Name des Karteninhabers, wie auf der Karte angegeben.

	Parameter	Format	Bedingung	Beschreibung
2	number	string	C	Das Vorhandensein hängt von der Händler-Konfiguration ab. Falls vorhanden, enthält dieses Element entweder die maskierte Kartennummer oder das PaygateEVO E-Pay-Kartentoken.
3	expiryDate	string	C	Vorhanden, falls number das PaygateEVO E-Pay-Kartentoken enthält
4	bin	object	M	Bankleitzahl (BIN) einschließlich Kontenbereich, falls zutreffend
5	brand	string	M	Name des Kartennetzwerkes (z.B. 'Visa', 'Mastercard')
6	product	string	C	Kartenproduktname (falls verfügbar) (z.B. 'Business Premium Debit').
7	source	string	C	Kartenfinanzierungsquelle (falls verfügbar) Zulässige Werte: • `DEBIT` • `CREDIT` • `DEFERRED DEBIT` • `PREPAID` • `CHARGE`
8	type	string	C	Der Kartentyp gibt das zur Karte gehörige Programm, Anwendung oder Karten-Level an, falls vorhanden (z.B. Classic, Standard, Gold, Business usw.)
9	country	JSON	M	Land, in dem die Karte ausgestellt ist
10	issuer	string	C	Kartenaussteller (falls verfügbar)

bin

	Parameter	Format	Bedingung	Beschreibung
1	accountBIN	string	M	Die ersten sechs Ziffern der Kontonummer, die auch als Bankleitzahl bekannt sind (**BIN**)
2	accountRangeLow	string	C	Die Kontonummer am unteren Ende des Kontenbereichs
3	accountRangeHigh	string	C	Die Kontonummer am oberen Ende des Kontenbereichs

4.4 credentialOnFile

	Parameter	Format	Bedingung	Beschreibung
1	type	object	M	Art der Zahlung mit hinterlegten Zugangsdaten
2	initialPayment	boolean	M	Gibt an, ob eine Kartentransaktion mit hinterlegten Daten die erste Transaktion einer Reihe (Einrichtung) oder eine nachfolgende Transaktion ist

4.4.1 type

	Parameter	Format	Bedingung	Beschreibung
1	recurring	object	C	Wiederkehrende Zahlunge sind eine Reihe von Transaktionen, die gemäß einer Vereinbarung zwischen einem Karteninhaber und einem Händler erfolgen, wobei der Karteninhaber Waren oder Dienstleistungen über einen Zeitraum durch eine Anzahl separater Transaktionen kauft. Beachten Sie bitte, das im Kontext von PSD2 und SCA die Anforderungen der Europäischen Bankenaufsichtsbehörde (EBA) wiederkehrende Zahlungen als eine Reihe von Transaktionen mit dem gleichen Betrag und dem gleichen Zahlungsempfänger beschreiben.

	Parameter	Format	Bedingung	Beschreibung
2	unscheduled	string	C	Wert, der die Partei angbit, welche eine Transaktion mit hinterlegten Zahlungsdaten auslöst, die nicht nach einem festen Zeitplan erfolgt. Zulässige Werte: • `CIT` = Vom Kunden ausgelöste Transaktion • `MIT` = Vom Händler ausgelöste Transaktion

4.4.1.1 recurring

	Parameter	Format	Bedingung	Beschreibung
1	recurringFrequency	integer	M	Gibt die Anzahl der Tage zwischen den Autorisierungen an
2	recurringStartDate	string	O	Bestimmt das Datem der ersten Autorisierunge gemäß dem wiederkehrenden Mandat
3	recurringExpiryDate	string	M	Datum, nach dem keine weiteren Autorisierungen mehr ausgeführt werden sollen

4.4.2 Schema


```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/credentialOnFile.json",
  "title": "credentialOnFile",
  "description": "Transaktionen mit hinterlegten Zugangsdaten",
  "type": "object",
  "properties": {
    "type": {
      "type": "object",
      "properties": {
        "recurring": {
          "type": "object",
          "properties": {
            "recurringFrequency": {
              "type": "integer",
              "minimum": 1,
              "maximum": 9999,
              "description": "Gibt die Mindestanzahl von Tagen zwischen wiederkehren-
den Autorisierungen an"
            },
            "recurringStartDate": {
              "type": "string",
              "format": "full-date",
              "description": "JJJJ-MM-TT"
            },
            "recurringExpiryDate": {
              "type": "string",
              "format": "full-date",
              "description": "JJJJ-MM-TT"
            }
          },
          "required": ["recurringExpiryDate", "recurringFrequency"],
          "additionalProperties": false
        },
        "unscheduled": {
          "type": "string",
          "enum": ["CIT", "MIT"]
        }
      },
      "oneOf": [
        {"required": ["recurring"]},
        {"required": ["installments"]},
        {"required": ["unscheduled"]}
      ],
      "additionalProperties": false
    },
    "initialPayment": {
      "type": "boolean"
    }
  },
  "required": ["type", "initialPayment"],
  "additionalProperties": false
}
```

4.4.3 Beispiel wiederkehrend

```
{
  "type": {
    "recurring": {
      "recurringFrequency": 30,
      "recurringStartDate": "2019-09-14",
      "recurringExpiryDate": "2020-09-14"
    }
  },
  "initialPayment": true
}
```

4.4.4 Beispiel ungeplante CIT

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": false
}
```

4.5 customerInfo

	Parameter	Format	Bedingung	Beschreibung
1	consumer	object	C	Objekt zur Beschreibung von Privatkunden. Erforderlich, wenn der Kunde eine natürliche Person ist.
2	business	object	C	Objekt zur Beschreibung von Geschäftskunden. Erforderlich, wenn der Kunde eine juristische Person ist.
3	phone	object	C	Telefonnummer. Erforderlich (falls verfügbar), sofern nicht Markt- oder regionale Mandate die Übermittlung dieser Information beschränken.
4	mobilePhone	object	C	Mobiltelefonnummer. Erforderlich (falls verfügbar), sofern nicht Markt- oder regionale Mandate die Übermittlung dieser Information beschränken.
5	email	string	C	E-Mail-Adresse. Erforderlich, sofern nicht Markt- oder regionale Vorgaben die Übermittlung dieser Information einschränken.

4.5.1 consumer

	Parameter	Format	Bedingung	Beschreibung
1	salutation	string	O	Anrede Zulässige Werte: • `Mr` • `Mrs` • `Miss`
2	firstName	string	M	Vorname des Kunden
3	lastName	string	M	Nachname des Kunden
4	birthDate	string	O	Geburtsdatum des Kunden im Format JJJJ-MM-TT

4.5.2 business

	Parameter	Format	Bedingung	Beschreibung
1	legalName	string	M	Firmenname
2	dbaName	string	O	Geschäfte tätigens als
3	registrationNumber	string	O	Unternehmens-Registrierungsnummer

4.5.3 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/customerInfo.json",
  "title": "customerInfo",
  "description": "Kundeninformationen",
  "type": "object",
  "properties": {
    "consumer": {
      "properties": {
        "salutation": {
          "type": "string",
          "enum": ["Mr", "Mrs", "Miss"]
        },
        "firstName": {
          "type": "string",
          "maxLength": 30
        },
        "lastName": {
          "type": "string",
          "maxLenght": 30
        },
        "birthDate": {
          "type": "string",
          "format": "full-date",
          "description": "JJJJ-MM-TT"
        }
      },
      "required": ["firstName", "lastName"],
      "additionalProperties": false
    },
    "business": {
      "properties": {
        "legalName": {
          "type": "string",
          "maxLenght": 50
        },
        "dbaName": {
          "type": "string",
          "maxLenght": 50,
          "description": "Geschäfte tätigens als. Unternehmensname, wie er üblicher-
weise den Kunden bekannt ist."
        },
        "registrationNumber": {
          "type": "string",
          "maxLenght": 20
        }
      },
      "required": ["legalName"],
      "additionalProperties": false
    },
    "phone": {
      "type": "object",
      "properties": {
        "countryCode": {
          "type": "string",
          "minLength": 1,
          "maxLenght": 3
```

```
        },
        "subscriberNumber": {
          "type": "string",
          "maxLenght": 12
        }
      },
      "required": ["countryCode", "subscriberNumber"],
      "additionalProperties": false
    },
    "mobilePhone": {
      "type": "object",
      "properties": {
        "countryCode": {
          "type": "string",
          "minLength": 1,
          "maxLenght": 3
        },
        "subscriberNumber": {
          "type": "string",
          "maxLenght": 12
        }
      },
      "required": ["countryCode", "subscriberNumber"],
      "additionalProperties": false
    },
    "email": {
      "type": "string",
      "maxLenght": 50,
      "format": "idn-email"
    }
  },
  "oneOf": [
    {"required": ["consumer"]},
    {"required": ["business"]}
  ],
  "additionalProperties": false
}
```

4.5.4 Beispiel

```
{
  "consumer": {
    "salutation": "Mr",
    "firstName": "Napoleon",
    "lastName": "Bonaparte",
    "birthDate": "1769-08-15"
  },
  "mobilePhone": {
    "countryCode": "33",
    "subscriberNumber" : "12345678910"
  },
  "email": "napoleon.bonaparte@france.com"
}
```

4.6 ipInfo

	Parameter	Format	Bedingung	Beschreibung
1	ipAddress	string	M	IP-Adresse
2	country	object	M	Land der IP-Herkunft
3	state	string	C	Bundesstaaten und Provinzen (das ist die erste Ebene der administrativen Gliederung) in allen Ländern, wo es sie gibt
4	city	string	M	Stadt in lokaler Schreibweise
5	longitude	string	M	Geographische Länge des ermittelten Ortes als Gleitkommazahl im Bereich von -180 to 180, wobei positive Zahlen Osten und negative Zahlen West bedeuten
6	latitude	string	M	Geographische Breite des ermittelten Ortes als Gleitkommazahl im Bereich von -90 to 90, wobei positive Zahlen Norden und negative Zahlen Süden bedeiten. Breite und Länge werden von der Stadt oder der Postleitzahl ausgehend ermittelt.

4.6.1 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://www.computop-paygate.com/schemas/ipInfo.json",
  "title": "ipInfo",
  "description": "IP-Informationen",
  "type": "object",
  "properties": {
    "ipAddress": {
      "type": "string",
      "oneOf": [{"format": "ipv4"},{"format": "ipv6"}]
    },
    "country": {
      "type": "object",
      "properties": {
        "countryName": {
          "type": "string"
        },
        "countryA2": {
          "type": "string",
          "minLength": 2,
          "maxLength": 2
        },
        "countryA3": {
          "type": "string",
          "minLength": 3,
          "maxLength": 3
        },
        "countryNumber": {
          "type": "string",
          "minLength": 3,
          "maxLength": 3
        }
      },
      "required": ["countryName", "countryA2", "countryA3", "countryNumber"],
      "additionalproperties": false
    },
    "state": {
      "type": "string"
    },
    "city": {
      "type": "string"
    },
    "longitude": {
      "type": "string"
    },
    "latitude": {
      "type": "string"
    }
  },
  "required": ["ipAddress", "country", "city", "longitude", "latitude"],
  "additionalproperties": false
}
```

4.6.2 Beispiel


```
{
  "ipAddress": "178.37.173.82",
  "country": {
    "countryName": "poland",
    "countryA2": "pl",
    "countryA3": "pol",
    "countryNumber": "616"
  },
  "state": "wielkopolskie",
  "city": "poznan",
  "longitude": "16.83739",
  "latitude": "52.4136"
}
```


4.7 merchantRiskIndicator

	Parameter	Format	Bedingung	Beschreibung
1	deliveryEmail	string	O	Für elektronische Lieferung die E-Mail-Adresse, an die die Ware geliefert wurde
2	deliveryTimeframe	string	O	Gibt den Zeitrahmen für die Warenlieferung an
3	giftCardAmount	integer	O	Für den Kauf von Prepaid- oder Geschenkkarten der Gesamtkaufbetrag der Preipaid- oder Geschenkkarten in der kleinsten Währungseinheit
4	giftCardCount	integer	O	Für den Kauf von Prepaid- oder Geschenkkarten die Gesamtanzahl der gekauften einzelnen Prepaid- oder Geschenkkarten/-Codes
5	giftCardCurr	string	O	Für den Kauf von Prepaid- oder Geschenkkarten der dreistellige Währungscode der Geschenkkarte gemäß ISO 4217
6	preOrderDate	string	O	Für einen vorbestellten Kauf das erwartete Datum, an dem die Ware verfügbar sein wird (JJJJ-MM-TT)
7	preOrderPurchaseIndicator	boolean	O	Gibt an, ob der Kunde eine Bestellung für eine Ware mit zukünftiger Verfügbarkeit oder zukünftigem Veröffentlichungsdatum aufgibt
8	reorderItemsIndicator	boolean	O	Gibt an, ob der Kunde eine zuvor bereits gekaufte Ware erneut bestellt
9	shippingAddressIndicator	string	O	Gibt die für die Transaktion gewählte Liefermethode an. Wenn in dem Verkauf einer oder mehrere Artikel enthalten sind, verwenden Sie den Code Shipping Indicator für die physischen Waren, oder wenn alles digitale Güter sind, verwenden Sie den Code Shipping Indicator zur Beschreibung des teuersten Artikels.

4.7.1 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/merchantRisk.json",
  "title": "merchantRisk",
  "description": "Händler-Risikoinformationen",
  "type": "object",
  "properties": {
    "deliveryEmail": {
      "type": "string",
      "maxLength": 50,
      "format": "idn-email",
      "description": "Für elektronische Lieferung die E-Mail-Adresse, an die die Ware
geliefert wurde"
    },
    "deliveryTimeframe": {
      "type": "string",
      "enum": ["electronicDelivery", "sameDayDelivery", "nextDayDelivery", "twoOrMore-
DaysDelivery"]
    },
    "giftCardAmount": {
      "type": "integer",
      "maximum": 999999999999
    },
    "giftCardCount": {
      "type": "integer",
      "maximum": 99,
      "description": "Anzahl der für diesen Kauf genutzten Prepaid- oder Geschenkkar-
ten"
    },
    "giftCardCurr": {
      "type": "string",
      "minLength": 3,
      "maxLength": 3,
      "description": "Dreistelliger Währungscode der Geschenkkarte gemäß ISO 4217"
    },
    "preOrderDate": {
      "type": "string",
      "format": "full-date",
      "description": "Das erwartete Datum, wann die Ware verfügbar sein wird, im Format
JJJJ-MM-TT"
    },
    "preOrderPurchaseIndicator": {
      "type": "boolean",
      "description": "Gibt an, ob der Kunde eine Bestellung für eine Ware mit zukünf-
tiger Verfügbarkeit oder zukünftigem Veröffentlichungsdatum aufgibt"
    },
    "reorderItemsIndicator": {
      "type": "boolean",
      "description": "Gibt an, ob der Kunde eine zuvor bereits gekaufte Ware erneut
bestellt"
    },
    "shippingAddressIndicator": {
      "type": "string",
      "enum": [
        "shipToBillingAddress",
        "shipToVerifiedAddress",
        "shipToNewAddress",
```

```
          "shipToStore",
          "digitalGoods",
          "noShipment",
          "other"
        ],
        "description": "Gibt die für die Transaktion gewählte Liefermethode an. Wenn in
dem Verkauf einer oder mehrere Artikel enthalten sind, verwenden Sie den Code Shipping
Indicator für die physischen Waren, oder wenn alles digitale Güter sind, verwenden Sie
den Code Shipping Indicator zur Beschreibung des teuersten Artikels."
      }
    },
    "additionalProperties": false
}
```

4.7.2 Beispiel

```
{
  "deliveryEmail": "joe.bloggs@acme.com",
  "deliveryTimeframe": "twoOrMoreDaysDelivery",
  "giftCardAmount": 5000,
  "giftCardCount": 2,
  "giftCardCurr": "EUR",
  "preOrderDate": "2020-03-15",
  "preOrderPurchaseIndicator": true,
  "reorderItemsIndicator": true,
  "shippingAddressIndicator": "shipToStore"
}
```

4.8 priorAuthenticationInfo

	Parameter	Format	Bedingung	Beschreibung
1	priorAuthenticationData	string	O	Daten, die eine spezifischen vom Händler ausgeführten Authentifizierungsprozess wie FIDO unterstützten und dokumentieren
2	priorAuthenticationMethod	string	O	Zur vorherigen Authenrisierungen verwendeter Mechanismus des Karteninhabers. Zulässige Werte: • `frictionless` • `ACSchallenge` • `AVSverified` • `other`
3	priorAuthenticationTimestamp	string	O	Datum und Uhrzeit (siehe RFC 3339) der vorherigen Authentifizierung des Karteninhabers in **UTC**. JJJJ-MM-TT**T**HH:MM:SS+00:00
4	priorAuthenticationReference	string	O	Dieses Datenelement enthält eine ACS Transaktions-ID für eine vorherige authentiriserte Transaktion (beispielsweise die erste wiederkehrende Transaktion, die vom Karteninhaber authentifiziert wurde).

4.8.1 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id":            "https://spg.evopayments.eu/pay/schemas/priorAuthenticationInfor-
mation.json",
  "title": "Vorherige Authentifizierungs-Informationen",
  "type": "object",
  "properties": {
    "prioAuthenticationData": {
      "type": "string",
      "maxLength": 2048
    },
    "priorAuthenticationMethod": {
      "type": "string",
      "enum": ["frictionless", "ACSchallenge", "AVSverified", "other"]
    },
    "priorAuthenticationTimestamp": {
      "type": "string",
      "format": "date-time"
    },
    "priorAuthenticationReference": {
      "type": "string",
      "maxLength": 36
    }
  },
  "additionalProperties": false
}
```

4.8.2 Beispiel

```
{
  "priorAuthenticationMethod": "frictionless",
  "priorAuthenticationTimestamp": "2021-10-05T04:36:18+00:00",
  "priorAuthenticationReference": "d7c1ee99-9478-44a6-b1f2-391e29c6b340"
}
```

4.9 resultsResponse

Beachten Sie bitte, dass alle nachstehenden Datenelemente in `resultsResponse` vorhanden sind, aber je nach *Bedingung* einen *Leerstring* enthalten können.

	Parameter	Format	Bedingung	Beschreibung
1	threeDSServerTransID	string	M	EVO E-PAY PayID im kanonischen Format gemäß IETF RFC 4122.
2	acsTransID	string	M	Vom ACS vergebene universelle, eindeutige Transaktions-ID zur Identifikation einer Einzeltransaktion
3	acsRenderingType	object	C	Erforderlich, sofern nicht ACS Entkoppelte Bestätigung = true
4	authenticationType	string	C	Erforderlich, wenn der Transaktionsstatus = Y oder N ist.

	Parameter	Format	Bedingung	Beschreibung
				Gibt die Art der Authentifizierungsmethode an, die der Issuer als Challenge für den Karteninhaber verwenden will. Erforderlich, wenn der Transaktionsstatus = C oder D ist. Zulässige Werte: • `01 = static` • `02 = dynamic` • `03 = oob` Zukünftige Implementierung. Ab Protokollversion 2.2.0 aufwärts - • `04 = decoupled`
5	authentication-Value	string	C	Erfoderlich, wenn Transaktionsstatus = Y oder A
6	challengeCancel	string	C	Indikator, der darüber informiert, dass die Authentifizierung abgebrochen wurde. Zulässige Werte: • `01` = Karteninhaber wählte “Abbrechen” • `02` = Reserviert für zukünftige EMVCo Verwendung (Werte ungültig, solange sie nicht durch EMVCo definiert sind) • `03` = Zeitüberschreitung der Transaktion — Entkoppelte Authentifizierung • `04` = Zeitüberschreitung der Transaktion am ACS — andere Zeitüberschreitungen • `05` = Zeitüberschreitung der Transaktion am ACS — Erste CReq vom ACS nicht empfangen • `06` = Transaktionsfehler • `07` = Unbekannt • `08` = Zeitüberschreitung der Transaktion am SDK
7	dsTransID	string	M	Vom DS vergebene universelle, eindeutige Transaktions-ID zur Identifikation einer Einzeltransaktion
8	eci	string	C	Vom ACS oder DS bereitgestellter zahlungssystemspezifischer Wert, der das Ergebnis des Versuchs zur Authentifizierung des Karteninhabers angibt. Die Anforderungen für das Vorhandensein dieses Felds sind DS-spezifisch.
9	interactionCounter	string	M	Gibt die Anzahl der durch den Karteninhaber versuchten Authentifizierungs-Zyklen an
10	messageCategory	string	M	Identifiziert die Kategorie der Nachricht für einen bestimmten Anwendungsfall. Zulässige Werte: • `01` = PA • `02` = NPA
11	messageExtension	string	C	Nötige Daten zur Unterstützung von Anforderungen, die nicht anderweitig in der 3-D Secure Nachricht definiert sind, werden in einer Nachrichten-Erweiterung transportiert. Bedingungen sind von jedem DS festzulegen.
12	messageType	string	C	Identifiziert die Art der gescheiterten Nachricht im Fehlerfall. Zulässige Werte: • `ARes` • `RReq`
13	messageVersion	string	M	Kennung für die Protokollversion
14	sdkTransID	string	M	Zukünftige Verwendung. Vom 3DS SDK vergebene universelle, eideutige Transaktion-ID zur Ifentifikation einer Einzeltransaktion.

	Parameter	Format	Bedingung	Beschreibung
15	transStatus	string	M	Gibt an, ob sich eine Transaktion als eine authentifizierte Transaktion qualifiziert. Zulässige Werte: • `Y` = Authentifizierungs-Überprüfung erfolreich • `N` = Nicht authentifiziert /Konto nicht verifiziert; Transaktion abgelehnt • `U` = Authentifizierung/ Kontoverifizierung konnte nicht ausgeführt werden; technisches oder anderes Problem, wie in ARes oder RReq angegeben • `A` = Verarbeitung der Versuche ausgeführt; Nicht authentifiziert/verifiziert, aber Nachweis der versuchten Authentifizierung/Verfizierung ist bereitgestellt • `C` = Challenge erfoderlich; zusätzliche Authentifizierung mittels CReq/CRes ist erforderlich • `D` = Challenge erfoderlich; entkoppelte Authentifizierung bestätigt • `R` = Authentifizierung/ Kontoverifizierung abgelehnt; Issuer lehnt Authentifizierung/Verifizierung ab und fordert, dass keine Autorisierung versucht wird • `I` = Nur zur Information; 3DS Requestor Challenge-Präferenz anerkannt
16	transStatusReason	string	C	Gibt Informationen darüber, warum das Feld Transaktionsstatus den angegebenen Wert hat. Erforderlich, wenn das Feld Transaktionsstatus = N, U oder R ist. Zulässige Werte: • `01` = KartenAuthentifizierung gescheitert • `02` = Unbekanntes Gerät • `03` = Nicht unterstütztes Gerät • `04` = Überschreitet das Limit für die Authentifizierungshäufigkeit • `05` = Abgelaufene Karte • `06` = Ungülstige Kartennummer • `07` = Ungültige Transaktion • `08` = Kein Kartendatensatz • `09` = Sicherheitsfehler • `10` = Gestohlene Karte • `11` = Betrugsverdacht • `12` = Transaktion für den Karteninhaber nicht erlaubt • `13` = Karteninhaber nicht für den Service angemeldet • `14` = Zeitüberschreiung der Transaktion am ACS • `15` = Geringes Vertrauen • `16` = Mittleres Vertrauen • `17` = Hohes Vertrauen • `18` = Sehr hohes Vertrauen • `19` = Überschreitet das ACS Maximum der Challenges • `20` = Nicht zahlungswirksame Transaktion nicht unterstützt • `21` = 3RI-Transaktion nicht unterstützt • `22` = ACS technisches Problem • `23` = Entkoppelte Authentifizierung vom ACS gefordert, aber vom 3DS Requestor nicht angefragt • `24` = 3DS Requestor maximale Ablaufzeit bei Entkopplung überschritten • `25` = Entkoppelte Authentifizierung hatte unzureichend Zeit für die Authentifizierung des Karteninhabers. ACS macht keinen Versuch

	Parameter	Format	Bedingung	Beschreibung
				• `26` = Authentifizierung versucht, aber vom Karteninhaber nicht ausgeführt
17	whiteListStatus	string	C	**Zukünftige Verwendeng.** Erst ab Protokollversion 2.2.0 aufwärts unterstützt. Ermöglicht die Übermittlung von Statuswerten Vertrauenswürdige Empfänger/Whitelist. Zulässige Werte: • `Y` = 3DS Requestor steht beim Karteninhaber auf der Whitelist • `N` = 3DS Requestor steht beim Karteninhaber nicht auf der Whitelist • `E` = Nach Festlegung vom Issuer nicht wählbar • `P` = Ausstehende Bestätigung vom Karteninhaber • `R` = Karteninhaber abgelehnt • `U` = Whitelist-Status unbekannt, nicht verfügbar oder nicht zutreffend
18	whiteListStatusSource	string	C	**Zukünftige Verwendeng.** Erst ab Protokollversion 2.2.0 aufwärts unterstützt. Dieses Datenelement wird von dem System ausgefüllt, dass den Whitelist-Status setzt. Zulässige Werte: • `01` = 3DS Server • `02` = DS • `03` = ACS

4.9.1 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://www.computop-paygate.com/schemas/resultsResponse.json",
  "type": "object",
  "properties": {
    "threeDSServerTransID": {
      "type": "string",
      "maxLength": 36
    },
    "acsTransID": {
      "type": "string",
      "maxLength": 36
    },
    "acsRenderingType": {
      "type": "object",
      "properties": {
        "acsInterface": {
          "type": "string",
          "enum": ["native", "html", ""],
          "description": "Die ACS-Schnittstelle, die dem Karteninhaber die Challenge
zeigen wird."
        },
        "acsUiTemplate": {
          "type": "string",
          "enum": ["text", "singleSelect", "multiSelect", "oob", "other",""],
          "description": "Bestimmt das Format der UI-Vorlage, die der ACS dem Kunden
zuerst zeigt."
        }
      },
      "required": ["acsInterface", "acsUiTemplate"],
      "additionalProperties": false
    },
    "authenticationType": {
      "type": "string",
      "enum": ["01", "02", "03", "04", ""]
    },
    "authenticationValue": {
      "type": "string",
      "maxLength": 28
    },
    "challengeCancel": {
      "type": "string",
      "enum": ["01", "02", "03", "04", "05", "06", "07", "08", ""]
    },
    "dsTransID": {
      "type": "string",
      "maxLength": 36
    },
    "eci": {
      "type": "string",
      "maxLength": 2
    },
    "interactionCounter": {
      "type": "string",
      "maxLength": 2
    },
    "messageCategory": {
```

```
      "type": "string",
      "enum": ["01", "02"]
    },
    "messageExtension": {
      "type": "string",
      "maxLength": 81920
    },
    "messageVersion": {
      "type": "string",
      "minLength": 5,
      "maxLength": 8
    },
    "sdkTransID": {
      "type": "string",
      "maxLength": 36
    },
    "transStatus": {
      "type": "string",
      "enum": ["Y", "N", "U", "A", "C", "D", "R", "I", ""]
    },
    "transStatusReason": {
      "type": "string",
      "enum": ["01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12",
"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24", "25", "26",
""]
    }
  },
  "required": ["threeDSServerTransID", "acsTransID", "acsRenderingType", "authentica-
tionType", "authenticationValue", "challengeCancel", "dsTransID", "eci", "interac-
tionCounter", "messageCategory", "messageExtension", "messageVersion", "sdkTransID",
"transStatus", "transStatusReason"],
  "additionalProperties": false
}
```

4.9.2 Beispiel


```
{
  "threeDSServerTransID":"9e944d5d-56f3-461d-a393-80a666d346d1",
  "acsTransID":"1e43b52f-3623-4e5d-8917-41c5c15b7218",
  "acsRenderingType":{
    "acsInterface":"01",
    "acsUiTemplate":"01"
  },
  "authenticationType":"02",
  "authenticationValue":"JAmi21makAifmwqo2120cjq1AAA=",
  "challengeCancel":"",
  "dsTransID":"c626e8a0-f2ba-42b3-aa6d-620658421f3a",
  "eci":"05",
  "interactionCounter":"01",
  "messageCategory":"01",
  "messageExtension":"",
  "messageVersion":"2.1.0",
  "sdkTransID":"",
  "transStatus":"Y",
  "transStatusReason":""
}
```


4.10 threeDSConfig

Dieses Datenelement kann verwendet werden, um die für die Merchant ID hinterlegten Konfigurationsdaten zu überschreiben.

	Parameter	Format	Bedingung	Beschreibung
1	loginID	string	M	3DS Login-ID
2	acquirerBIN	string	M	Bankleitzahl (BIN) des Acquirerss
3	merchantCategoryCode	string	M	Kategoriecode des Händlers (MCC).
4	merchantCountry	object	M	Händlerland
5	merchantName	string	M	Händlername

4.10.1 Schema


```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/threeDSConfig.json",
  "title": "3DS Config",
  "description": "3DS Konfigurationsdaten",
  "type": "object",
  "properties": {
    "loginID": {
      "type": "string"
    },
    "acquirerBIN": {
      "type": "string",
      "minLength": 6,
      "maxLength": 8
    },
    "merchantCategoryCode": {
      "type": "string",
      "minLength": 4,
      "maxLength": 4
    },
    "merchantCountry": {
      "type": "object"
    },
    "merchantName": {
      "type": "string"
    }
  },
  "additionalProperties": false,
  "required": ["loginID", "acquirerBIN", "merchantCategoryCode", "merchantCountry",
"merchantName"]
}
```


4.10.2 Beispiel


```
{
  "loginID": "123456ABCDEF",
  "acquirerBIN": "123456",
  "merchantCategoryCode": "5965",
  "merchantCountry": {
    "countryA3": "pol"
  },
  "merchantName": "ACME INC."
}
```


4.11 threeDSData

- threeDSData:response

4.11.1 threeDSData:response

	Parameter	Format	Bedingung	Beschreibung
1	authenticationStatus	boolean	M	Gibt an, ob ein Karteninhaber authentifiziert ist oder nicht
2	acsProtocolVersion	string	M	Die zur Authentifizierung verwendete Protokoll-Version
3	authenticationValue	string	C	Zahlungssystemspezifischer Wert als Nachweis der Authentifizierung
4	eci	string	M	Zahlungssystemspezifischer E-Commerce-Indikator
5	threeDSServerTransID	string	C	Nur 3DS 2.0. EVO E-PAY PayID im kanonischern Format gemäß IETF RFC 4122
6	acsXID	string	C	Nur 3DS 1.0. Vom ACS vergebene Transaktions-ID

4.11.1.1 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://www.computop-paygate.com/schemas/threeDSData_response.json",
  "title": "3DS Data",
  "description": "3DS-Daten",
  "type": "object",
  "properties": {
    "authenticationStatus": {
      "type": "boolean"
    },
    "acsProtocolVersion": {
      "minLength": 5,
            "maxLength": 8
    },
    "authenticationValue": {
            "type": "string",
            "maxLength": 28
        },
    "eci": {
      "type": "string",
      "minLength": 2,
      "maxLength": 2
    },
    "threeDSServerTransID": {
      "type": "string",
            "maxLength": 36
        },
    "ACSXID": {
      "type": "string",
      "maxLength": 40
    }
  },
  "additionalProperties": false,
  "required": ["authenticationStatus", "acsProtocolVersion", "eci"]
}
```

4.12 threeDSPolicy

	Parameter	Format	Bedingung	Beschreibung
1	skipThreeDS	string	O	Gibt an, ob und unter welchen Bedingungen die Authentifizierung übersprungen werden soll oder ob lediglich eine Datenfreigabe erfolgen soll (dataOnly). Standardmäßig werden alle Transaktionen und SCA-Ausnahmen über EMV 3DS verarbeitet, sofern nicht anders angegeben. Zulässige Werte: • `thisTransaction` • `outOfScope` • `dataOnly`
2	threeDSExemption	object	O	Objekt, das die angeforderten SCA-Ausnahmen genauer angibt

4.12.1 threeDSExemption

	Parameter	Format	Bedingung	Beschreibung
1	exemptionReason	string	M	Bezeichnet die Art der SCA-Ausnahme (z.B. Acquirer TRA oder MIT), die anzuwenden ist. Zulässige Werte: • `transactionRiskAnalysis` • `delegatedAuthority` • `merchantInitiatedTransaction` • `lowValue` Beachten Sie bitte, dass Acquirer-Ausnahmen und vom Händler initiierte Transaktionen (MIT) auch durch eine Autorisierung ohne Authentifizierung angefordert werden können (d. h. EMV 3DS oder EMV 3DS Data Only). Beachten Sie bitte, dass merchantInitiatedTransaction ist nur in Kombination mit credentialOnFile gültig ist.
2	merchantFraudRate	integer	O	Die Händler-Betrugsrate in bps berücksichtige alle Händler-Seiten und Kartenumsätze und wird nach PSD2 RTS Artikel 19 berechnet. Die Händler-Betrugsrate ist optional und muss vom Acquirer berechnet werden. Die Übermittlung dieses Wertes könnte vorteilhaft sein, um das Vertrauen des ACS/Issuers in die lauefende Transaktion zu erhöhen. Issuer könnten ihn auch für die Entscheidung verwenden, ob ein Händler für die Whitelist-Ausnahme in Betracht kommt.

4.12.2 Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://spg.evopayments.eu/pay/schemas/threeDSPolicy.json",
  "title": "threeDSPolicy",
  "description": "3DS Policy",
  "type": "object",
  "properties": {
    "skipThreeDS": {
      "type": "string",
      "enum": ["thisTransaction", "outOfScope", "dataOnly"]
    },
    "threeDSExemption": {
      "type": "object",
      "properties": {
        "exemptionReason": {
          "type": "string",
          "enum": ["transactionRiskAnalysis", "delegatedAuthority", "merchantInitiat-
edTransaction", "lowValue"]
        },
        "merchantFraudRate": {
          "type": "integer",
          "minimum": 1,
          "maximum": 99
        }
      },
      "required": ["exemptionReason"],
      "additionalProperties": false
    }
  },
  "additionalProperties": false
}
```

4.12.3 Beispiel

```
{
  "skipThreeDS": "outOfScope",
  "threeDSExemption": {
    "exemptionReason": "merchantInitiatedTransaction",
    "merchantFraudRate": 4
  }
}
```

5. Wichtige Hinweise

- 3DS Authentication Hosting
- Amazon Payments MFA
- Dynamische Rechnungs-Deskriptoren
- Hinterlegte Zugangsdaten
- Konto-Verifizierung
- Mehrparteien-E-Commerce / Agenten-Modell
- Nicht zahlungswirksame Authentifizierungen für Card Add (Hinzufügen von Kartendaten)
- Obligatorisch und bedingt erforderliche Datenelemente für EMV 3DS

- schemeReferenceID

5.1 Dynamische Rechnungs-Deskriptoren

5.1.1 Allgemeine Anforderungen

Das Element `billingDescriptor` dient zum Überschreiben des Händlernamens, der gegebenenfalls an die Bank des Karteninhabers gesendet wird.
Der Händlername ist der wichtigste Faktor für den Karteninhaber, um eine üblicherweise auf seinem Kontoauszug gedruckte Transaktion zu erkennen. Er sollte den üblichen Namen einer Firma (d.h. der Name 'handelt als' (DBA)) anstatt der juristischen Bezeichnung enthalten, woran der Karteninhaber den Händler erkennen würde, um Verwechslungen zu vermeiden und Anfragen nach Kopien zu minimieren.
Standardmäßig leitet Acquire den Händlernamen weiter, den sie im Händlerkonto hinterlegt haben. Beachten Sie bitte, dass die Kartenorganisationen strenge Regeln für Händlernamen auf Kontoauszügen festgelegt haben.
Es gibt jedoch eine Reihe spezifischer Ausnahmen, wo abhängig von Anwendungsfall und Branche (z.B. Fluglinien, Bahnlinien, Autovermietungen, Tankstellen usw.) ergänzende Daten zum DBA-Namen hinzugefügt werden können.

5.1.2 Formatierung des Händlernamens

Die Autorisierungs- und Clearing-Systeme der Kartenorganisationen bieten unterschiedlichen Größen für Händlernamen an. Der kleinste gemeinsame Nenner sind 22 Zeichen. Daher passen Händlernamen mit mehr als 22 Zeichen nicht in das Feld für den Händlernamen und müssen in einer Art und Weise abgekürzt werden, die für den Karteninhaber noch erkennbar bleibt.

5.1.2.1 Kauf von Waren oder Dienstleistungen

Für normale Käufe von Waren oder Dienstleistungen können zusätzliche Informationen hinter dem Händlernamen und einem Sternchen (*) eingefügt werden, um eine Bestellnummer, Referenznummer oder anderen Angaben zur Identifikation einer Transaktion anzugeben.

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
G	R	E	A	T		B	R	A	N	D		L	T	D	*	0	8	1	5	3	7

Für Autovermiter und Hotelhändler darf der Händlername nicht gekürzt werden, um ergänzende Informationen im Feld des Händlernamens unterzubringen.

5.1.2.2 No-Show-Transaktionen

Sie dürfen nach dem Händlernamen auch die Wörter "NO SHOW" enthalten.

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
H	.	C	A	L	I	F	O	R	N	I	A		N	O		S	H	O	W		

5.1.2.3 Kauf eines Flug-Tickets (oder Passagier-Eisenbahnfahrkarte in Region USA)

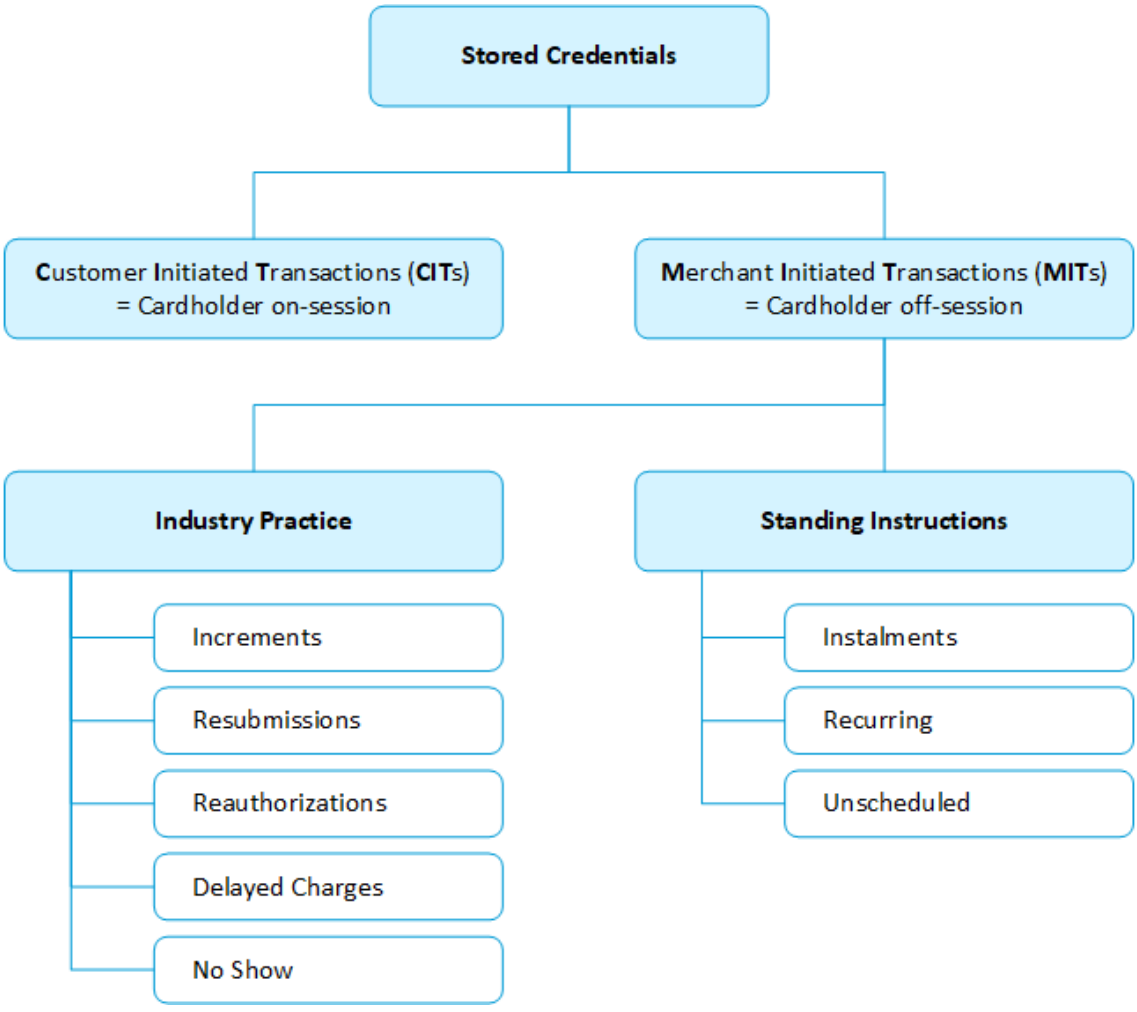
Alle folgenden Dinge müssen enthalen sein:

- Ein abgekürzter Name der Fluglinie (oder US Eisenbahn) in den ersten 11 oder 12 Zeichen
- Gegebenenfalls ein Leerzeichen an Position 12
- Ab Position 13 eine Kennung für das Flug- (oder US Eisenbahn-) Ticket

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
F	L	Y		L	O	W		P	L	C		1	2	3	4	5	6	7	8	9	0

5.2 Hinterlegte Zugangsdaten

Wann immer Karten-Zugangsdaten (d.h. Name des Karteninhabers, Kartennummer/Token und/oder Ablaufdatum) für eine zukünftige Verwendung gespeichert werden, ist eine vorherige Zustimmung durch den Karteninhaber erforderlich.
Während der Einrichtung eines solchen Mandats sollte der Karteninhaber über den genauen Grund für die Speicherung der Zugangsdaten beim Händler informiert werden. Das bedeutet, eine Autorisierungsanfrage zur Einrichtung eines Mandats für hinterlegte Zugangsdaten muss auch die Art der möglichen nachfolgenden Transaktionen angeben.
Diese nachfolgenden Transaktionen mit hinterlegten Zahlungsdaten, denen der Karteninhaber zugestimmt hat, werden ganz allgemein in vom Kunden ausgelöste Transaktionen (Customer Initiated Transactions / CITs) und vom Händler ausgelöste Transaktionen (Merchant Initiated Transactions / MITs) kategorisiert.



Der maßgebliche Unterschied zwischen **CITs** und **MITs** ist, dass Letztere außerhalb des Geltungsbereiches der RTS für die SCA sind. Das liegt daran, dass der Karteninhaber regelmäß nicht in einer Sitzung ist und daher praktisch für eine Authentifizierung nicht zur Verfügung steht.

Es gibt verschiedene Anwendungsfälle für MITs, die allgemein in Transaktionen gemäß einer bestimmten Branchenpraxis sowie ständige Anweisungen eingeteilt werden können.

In EVO E-PAY sind CITs und MITs für ständige Anweisungen durch das JSON-Objekt credentialOnFile markiert.

Beachten Sie bitte, dass alle außerplanmäßigen MIT-Transaktionen in 3DS 2.0 nicht unterstützt werden und daher direkt zur Autorisierung gesendet werden, ohne in die EVO E-PAY 3DS Sequenz zu gelangen.
Wiederkehrende MIT-Transaktionen werden jedoch über das 3DS 2.0 Protokoll zum Issuer gesendet, um bestmögliche Akzeptanzraten zu garantieren.

Beachten Sie bitte, dass Sie mit jeder anfänglichen CIT, welche ein Mandat für nachfolgenden MITs einrichtet, eine `schemeReferenceID` erhalten, die bei nachfolgenden Transaktionen übergeben werden muss, um die Sequnz zu verknüpfen.
Nachdem die SCA am 14. September 2019 verpflichtend geworden ist, können durch Zustimmung vom Karteninhaber gedeckte MITs weiterhin ohne eine `schemeReferenceID` verarbeitet werden, wenn die Mandate dafür vor diesem Datum eingerichtet wurde (d.h. Bestandsschutz). Übergeben Sie bitte keine Platzhalterwerte. EVO E-PAY verwendet automatisch entsprechende Werte im Autorisierungs-Protokoll, um den sogenannten Bestandsschutz anzuzeigen.

Vom Karteninhaber ausgelöste Transaktion (Cardholder Initiated Transaction / CIT)

Eine vom Karteninhaber ausgelöste Transaktion ist jede Transaktion, an der der Karteninhaber aktiv teilnimmt. Das kann entweder an einem Terminal in einem Geschäft oder beim Bezahlvorgang online sein oder mit hinterlegten Zahlungsdaten, bei denen der Karteninhaber zuvor der Speicherung beim Händler zugestimmt hat.

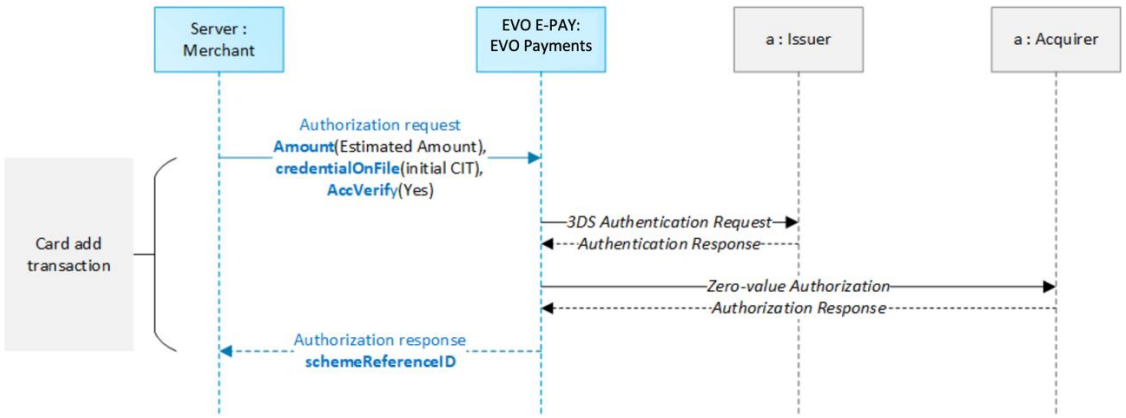
Vom Händler ausgelöste Transaktion (Merchant Initiated Transaction / MIT)

Jede Transaktion, die sich auf eine vorherige vom Karteninhaber ausgelöste Transaktion bezieht, aber ohne aktive Teilnahme des Karteninhabers durchgeführt wird. Im Ergebnis kann der Händler keine Validierung durch den Karteninhaber ausführen. In allen Fällen muss sich eine vom Händler ausgelöste Transaktion auf die originale Interaktion mit dem Karteninhaber beziehen.

5.2.1 Echtzeit-Service über mobile App mit Zahlung nach Service-Abschluss

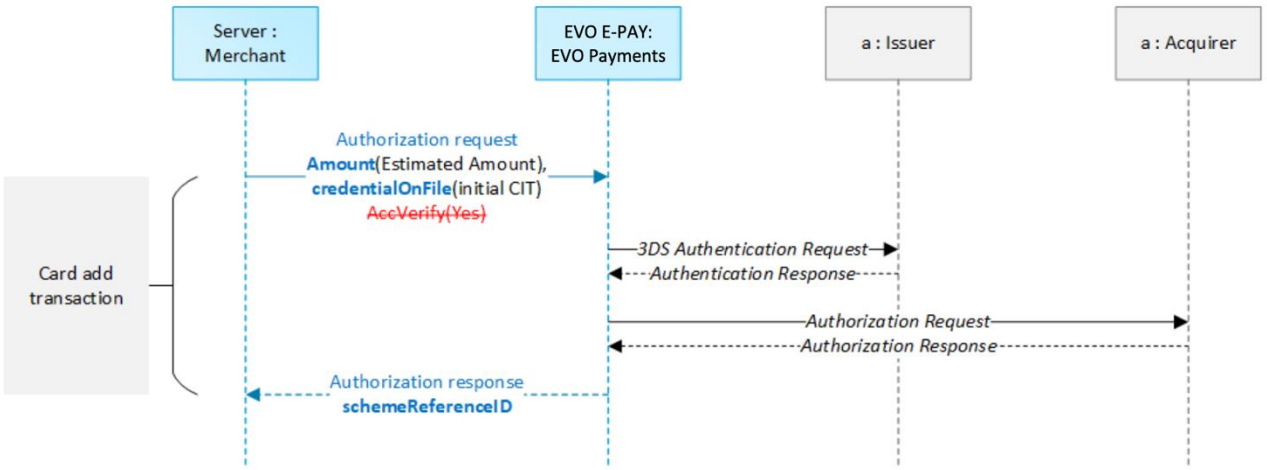
In vielen Szenarien der gemeinsamen Nutzung wie Car-Sharing oder Bike-Sharing ist das Mobilgerät des Kunden ein entscheidender Bestandteil für die Dienstleistungserbringung und das Zahlungssystem. Kartenzugangsdaten werden für optimale Benutzererfahrung häufig im Konto des Karteninhabers beim Dienstleister gespeichert.
Die Seqeunz der auszuführenden Schritte ist in den nachfolgenen Diagramm skizziert.

5.2.1.1 Hinzufügen der Karte als Teil einer nicht zahlungswirksamen Transaktion (NPA)

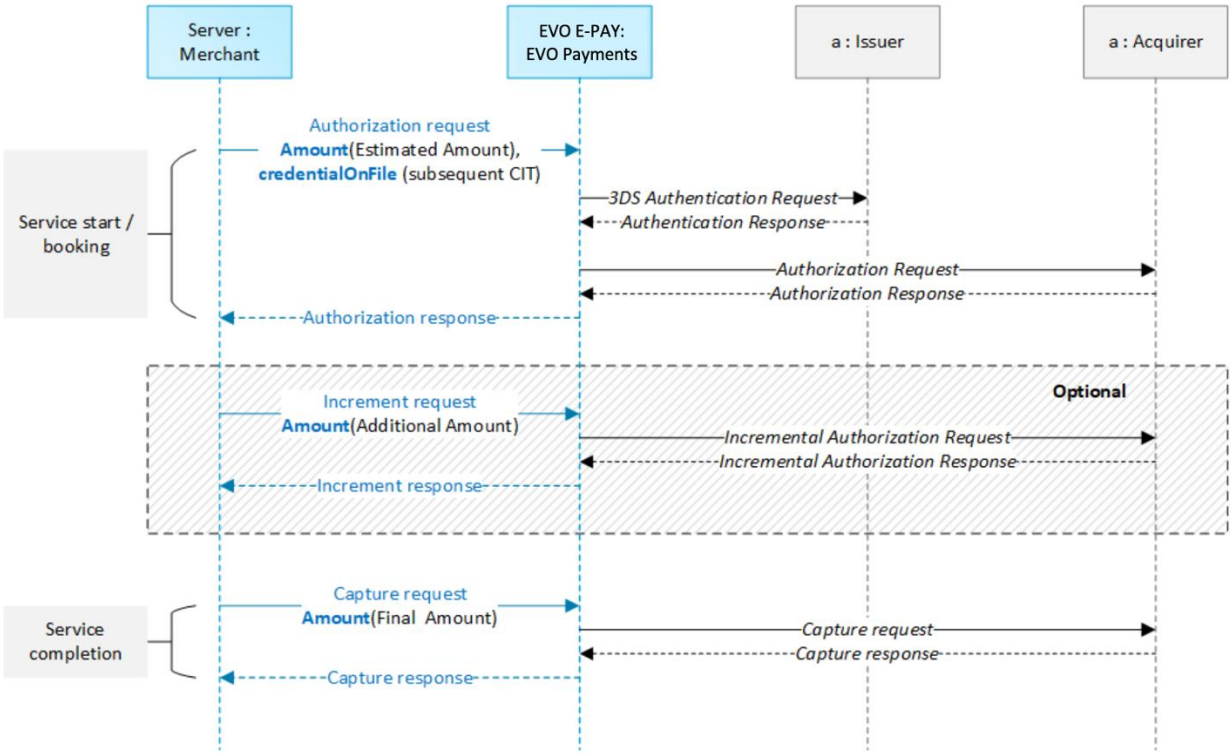


Falls das Hinzufügen der Karte (**Card Add**) NICHT Teil einer Zahlungstransaktion ist, muss obligatorisch eine Kontoverifizierung durchgeführt werden (siehe `AccVerify`).

5.2.1.2 Hinzufügen der Karte als Teil einer Zahlungstransaktion



5.2.1.3 Service-Bereitstellung



Außerplanmäßige **MITs** mit hinterlegten Zugangsdaten (UCOF) sind nicht für Szenarien anwendbar, wo der Karteninhaber zum Zeitpunkt des Service-Abschlusses in einer aktiven Sitzung und daher für eine Authentifizierung verfügbar ist. Das ist beispielsweise regelmäßig der Fall bei Apps für Car-Sharing oder den Ruf einer Fahrt.

Falls der geschätzte Betrag geringer als der endgültige Betrag ist, ist es empfehlenswert, eine vollständige Stornierung des ursprünglich autorisierten Betrags auszuführen und eine neue Autorisierung für den endgültigen Betrag einzureichen.

5.2.1.4 Amount

Hinzufügen einer Karte (Card Add)
Ein Nullwert oder ein geschätzter Wert. Beachten Sie bitte, dass der Betrag normalerweise dem Karteninhaber während der Authentifizierungs-Challange angezeigt wird und daher im Erwartungsbereich des Kunden liegen sollte.

Service-Bereitstellung
Der Betrag der Autorisierungsanfrage The amount in the authorization request should be an estimated for the service provision according to resonable customer expectations. Once the service has been completed incremental authorizations may be used before the final amount is captured.

5.2.1.5 credentialOnFile

Hinzufügen einer Karte (Card Add)
Das UCOF-Flag wird übermittelt, um ein Mandat zum Speichern von Zugangsdaten einzurichten und die anfängliche `schemeReferenceID` zu erhalten. Der Kartenherausgeber ist verpflichtet, währnd der Authentifizierung eine Verstärkung auszuführen.

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": true
}
```

Service-Bereitstellung
Das CIT-Flag wird übermittelt, um UCOF-Transaktionen ohnen Kartenprüfnummer zu ermöglichen.

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": false
}
```

5.2.1.6 AccVerify

Alle Transaktionen zum Hinzufügen von Karten (Card Adds), die nicht Teil einer Zahlungstransaktion sind, erfordern eine Konto-Verifizierung.

5.2.2 Verzögerte Lieferung

In manchen Fällen kann ein Händler eine Bestellung von einem Kunden erhalten, die nicht innerhalb der Haltedauer einer Autorisierung von 7 (d.a. abschließende Autorisierung) beziehungsweise 30 Tagen (d.h. Vorautorisierung) lieferbar ist.
Das ist üblicherweise der Fall für:

- individuell konfigurierte Produkte wie Fahrräder, Computer-Server, Möbel oder anderer angefertigte Artikel außerhalb der Standardspezifikationen, die im Auftrag gebaut werden (BTO)
- Vorbestellungen kommender Produkte wie neuer Telefonmodelle
- ausverkaufte Artikel

> Beachten Sie bitte, falls die Autorisierung deutlich später als die anfängliche Bestellung erfolgt, ist es eine gute Praxis, dem Karteninhaber ein paar Tage vor der Autorisierung eine Erinnerung zu schicken, um die Wahrscheinlichkeit zu erhöhen, dass die Gelder verfügbar sind.

Um eine mögliche Haftungsverschiebung der 3DS-Authentifizierung zu erhalten, ist es empfehlenswert, zwei Schritte im Prozess zu befolgen:

1. Anfängliches Hinzufügen der Karte als Teil einer nicht zahlungswirksamen Transaktion (NPA)
2. Nachfolgende außerplanmäßige COF MIT mit Authentifizierungsdaten aus dem Schritt 1 (UCOF MIT)

5.2.2.1 Anfängliches Hinzufügen der Karte als Teil einer nicht zahlungswirksamen Transaktion (NPA)

Um das Mandat für eine hinterlegte Karte einzurichten und den karteninhaber zu authentifizieren, übermitteln Sie bitte eine Autorisierungsanfrage an EVO E-PAY.

Betrag

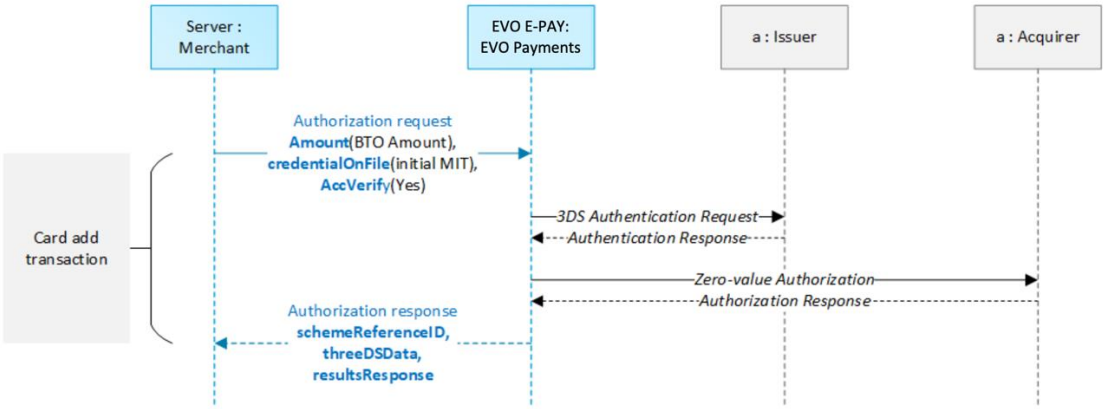
Der angegebene `Amount` wird innerhalb des 3DS Authentifizierungssprozesses verwendet und dem Kunden bei der Challenge des Karteninhabers angezeigt. Es sollte der endgültige Betrag sein, der dem Kunden belastet wird, nachdem die Bestellung ausgeführt ist, da dies zugleich der Maximalbetrag für die Haftungsverscheibung ist.

COF

Die Challenge des Karteninhabers wird durch den Indikator `credentialOnFile` erzwungen, der die Transaktion als Einrichtung eines Mandats für nachfolgenden MITs kennzeichnet.

Konto-Verifizierung

Um eine sofortige Autorisierung des vollen Bestellbetrags auf dem Konto des Karteninhabers zu unterdrücken und die Regeln des Kartensystems für Transaktionen zum Hinzufügen einer Karte einzuhalten, ist es erforderlich, eine Konto-Verifizierung (alias eine Nullwert-Autorisierung) durchzuführen, indem der Parameter `AccVerify` mit dem Wert 'Yes' übermittelt wird.



Das JSON-Objekt `threeDSData` in der Antwort enthält zusammen mit dem bedingten (d.h. Challenge-Ablauf) JSON-Objekt `resultsResponse` die nötigen Authentifizierungsdaten für die nachfolgende MIT Autorisierungsanfrage, sobald die Bestellung lieferbereit ist.

5.2.2.2 UCOF MIT

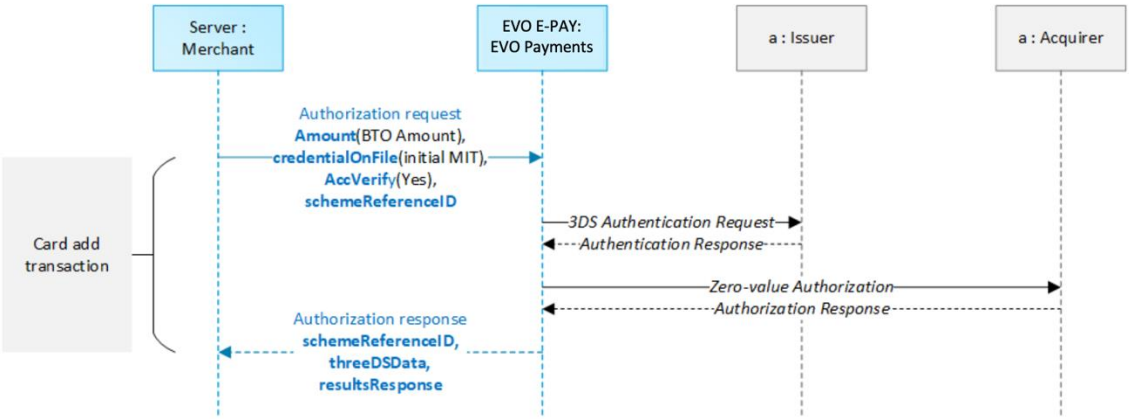
Sobald das Produkt oder die Dienstleistung verfügbar wird und lieferbereit ist oder zu jedem anderem Datum, das eine Autorisierungsgenehmigung auf dem Konto des Karteninhabers ermöglicht, übermitteln Sie bitte eine als UCOF MIT markierte Autorisierungsanfrage in Kombination mit den Authentifizierungsdaten, die Sie bei der anfänglichen Transaktion zur COF-Einrichtung erhalten haben.

COF

Die Markierung `credentialOnFile` der MIT hindert den Issuer daran, eine Challenge des Karteninhabers anzufordern und verknüpft die Transaktion mittels der `schemeReferenceID` mit der anfänglichen COF-Transaktion.

threeDSData

Die Haftungsabsicherung lässt such durch Übermitteln des Objekts `threeDSData` einrichten, dass die Authentifizierungsdaten der anfänglichen COF CIT Transaktion enthält.



5.3 Konto-Verifizierung

Eine auch als Nullwert-Aotorisierung bekannte Konto-Verifizierung ist eine Anfrage zur Prüfung, ob ein Kartenkonto in gutem Ansehen ist (d.h. die Karte ist nicht gestohlen und das Kartenkonto kann für Zahlungen verwendet werden).

5.3.1 3DS und Konto-Verifizierung

Beachten Sie bitte, dass bei einer angefragten Konto-Verifizierung (d.h. AccVerify=Yes) in Kombination mit 3DS die Autorisierung stets mit einem Nullwert ausgeführt wird. Die Authentifizierung wird hingegen mit dem Betrag ausgeführt, der dem im Feld `Amount` übermittelten Betrag entspricht. Wenn der übermittelte `Amount` Null ist, erfolgt die zugehörige Haftungsumkehr ebenfalls für einen Nullwert.

5.3.1.1 Hinterlegte Zugangsdaten

Beachten Sie bitte, dass nicht zahlungswirksame Transaktionen, die `credentialOnFile` (alias Card Add) hinterlegen, eine Konto-Verifizierung erfordern.

5.4 Nicht zahlungswirksame Authentifizierungen für Card Add (Hinzufügen von Kartendaten)

Verwenden Sie bitte `AccVerify`, um eine Konto-Verifizierung anzufordern, wenn Sie ohne Zahlung eine Karte zu COF (Credential on file - hinterlegte Kartendaten) hinzufügen wollen.
Nicht zahlungswirksame Authentifizierungen für Transaktionen zum Hinzufügen von Kartendaten (Card Add) erfordern immer eine verstärkte Authentifizierung (d.h. Challenge).

Eine Bereitstellung wie Card Add ist allgemein zu betrachten als eine

> *Aktion über einen Remote-Kanal, die ein Risiko für Zahlungsbetrug oder anderen Missbrauch darstellen kann*

gemäß Artikel 97(1)(C) PSD2. Für diese Aktionen sind keine Ausnahmen vorgesehen.

Wenn eine Karte zu COF hinzugefügt und gleichzeitig eine Zahlung angefordert wird, ist nur eine starke Kundenauthentifizierung (SCA) erforderlich.

Anwendungsfall	Flags
COF hinzufügen ohne Zahlung	`AccVerify` `credentialOnFile`
COF hinzufügen als Teil einer Zahlung	`credentialOnFile`

5.5 Obligatorisch und bedingt erforderliche Datenelemente für EMV 3DS

Beachten SIe bitte, dass einige als bedingt erforderlichen Datenelemente in EMV 3DS weiter spezifiziert sind als:

> *Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken.*

Das betrifft beispielsweise Datenelemente wie email, phone number oder billing address.
Unter Berücksichtigung des rechtlichen Umfelds und abhängig vom Issuer *(oder exakter vom Anbieter des Access Control Servers (ACS) des Issuers)* kann diese Spezifikation interpretiert werden als streng erforderlich im Europäischen Wirtschaftsraum (EEA).

Im Gegensatz zur EMV 3DS Protokoll-Spezifikation:

> *A.1 Fehlende erforderliche Felder*
> *. . .*
> *Sofern nicht ausrücklich angegeben, gibt die empfangenden Komponente eine Fehlermeldung zurück, wenn ein erforderliches Feld fehlt . . . Das trifft zu, gleichgültig ob das Feld immer Pflicht oder bedingt erforderlich ist.*

haben die Kartensysteme verfügt, dass Issuer EMV 3DS-Nachrichten nicht ablehnen dürfen, wenn eines oder mehrere bedingte Felder fehlen. Die Kartensysteme schreiben aber auch vor, dass Händler bedingte Felder in EMV 3DS-Nachrichten gemäß den geltenden Datenschutzgesetzen senden müssen.
Folgende Elemente gelten als zentral, und es wird dringend empfohlen, diese Daten anzugeben:

- Informationen zum Karteninhaber
  - Name
  - E-Mail-Adresse
  - Telefonnummer
  - Mobiltelefonnummer
  - Rechnungsadresse
  - Lieferadresse

Als allgemeine Regel ist es dringend empfohlen, bedingt erforderlich Datenelemente stets zu übermitteln, um unnötige Reibereien und Ablehnungen zu vermeiden.

5.6 schemeReferenceID

Das ist eine eindeutige direkt von den Kartensystemen wie VISA und MC bereitgestellte Transaktions-ID, um eine Transaktion im gesamten Zahlungs-Ökosystem eindeutig zu referenzieren. Sie wurde anfänglich von VISA gemäß deren Framework-Spezifikationen wie COF (Credential On File) und MIT (Merchant Initiated Transactions) eingeführt und ist relevant für Anwendungsfälle mit Transaktionsarten wie Wiederkehrend, U-COF (MIT), Inkrementell, Verzögerte Autorisierung, Wiedervorlage usw.
Mit der Veröffentlichung der EMV 3DS-Spezifikationen entstand auch für Mastercard die Anforderung, eine derartige eindeutige ID zu verwenden, welche sie "traceID" oder "grandfathering ID" nannten. Die Logik dahinter ist, dass sich der Issuer auf diese ID verlassen kann, um die anfängliche Zahlung mit allen nachfolgenden zu verknüpfen, die sich in einem Dauerauftrag in in einem COF- oder MIT-Regime darauf beziehen. Das ermöglicht dem Issuer, für alle nachfolgenden Zahlungen abweichende Transakationsregeln anzuwenden (d.h. kein CVV/CVC, keine zusätzliche Authentifizierung in EMV 3DS).

In der derzeitigen Situation wurde den Händlern für Mastercard/Maestro-Transaktionen, bei denen die anfängliche Zahlung (Einrichtung einer Vereinbarung) vo dem 14. September 2019 erfolgt ist, in den Autorisierungsantworten keine "*schemeReferenceID*" bereitgestellt. EVO Payments fordert von jenen Händlern, die diese Anwendungsfälle betreffen, diesen Parameter bei allen nachfolgenden (COF/MIT) Transaktionen leer zu belassen.
Bei **anfänglichen Zahlungen** (Einrichtung einer Vereinbarung) nach dem 14. September 2019 müssen die Händler den in der Antwort als "*schemeReferenceID*" bereitgestellten Wert speichern und in allen nachfolgenden Zahlungen, die sich auf diese anfängliche Vereinbarung beziehen, an EVO E-PAY übermitteln. Bei VISA ist die "*schemeReferenceID*" äquivalent zum vorherigen EVO E-Pay-Parameter "*TID*", den die Händler derzeit gemäß COF & MIT Frameworks übermitteln.

6. Test-Karten

6.1 Kartennummern

	Visa	Mastercard	Test-Szenario
1	4000012892688323	5232125125401459	Browser-Challenge
2	4000016435940133	5232122189301469	Browser-Challenge
3	4000012699048523	5232127264637786	Browser reibungslos; fehlende DS Transaktions-ID
4	4000011744135012	5232122741507017	Nicht authentifizierter Browser reibungslos
5	4000019966199434	5232122422543299	Authentifizierter Browser reibungslos
6	4000015573198637	5232128083944791	Browser-Challenge fehlende ACS URL
7	4000017873485953	5232122596907270	Authentifizierungs-Protokollfehler
8	4000014730366880	5232124106987982	Browser-Challenge; authentifizierte Transaktion; fehlender Authentifizierungswert

6.2 Einmal-Passwörter (OTPs)

Hinweis: Bitte bestätigen Sie das Einmal-Passwort im Falle einer Challenge per Mausklick und nicht mit der Eingabetaste, da sonst sie Schaltfläche "Abbrechen" ausgewählt und die Authentifizierung abgebrochen wird.

	otpValue	transStatus	transStatusReason	ECI	authenticationValue
1	1234	Y		01	JAmi21makAifmwqo2120cjq1AAA=
2	1111	N	01	01	
3	2222	R	01	01	
4	3333	U	01	01	
5	6666	Y	01	01	
6	7777	A		01	JAmi21makAifmwqo2120cjq1AAA=
7	8888	N	10		
8	9999	N	08		
9	0001	N	01		
10	0002	N	02		
11	0003	N	03		
12	0004	N	04		
13	0005	N	05		
14	0006	N	06		
15	0007	N	07		
16	0009	N	09		
17	0010	N	10		
18	0011	N	11		

6.2.1 transStatus

	transStatus	Beschreibung
1	Y	Authentifizierungs-Verifizierung erfolgreich
2	N	Nicht authentifiziert /Konto nicht verifiziert; Transaktion abgelehnt

	transStatus	Beschreibung
3	U	Authentifizierung/ Konto-Verifizierung konnte nicht ausgeführt werden; technisches oder anderes Problem, wie in ARes oder RReq angegeben
4	A	Verarbeitung der Versuche ausgeführt; Nicht authentifiziert/verifiziert, aber Nachweis der versuchten Authentifizierung/Verfizierung ist bereitgestellt
5	C	Challenge erfoderlich; zusätzliche Authentifizierung mittels CReq/CRes ist erforderlich
6	D	Challenge erfoderlich; entkoppelte Authentifizierung bestätigt
7	R	Authentifizierung/ Kontoverifizierung abgelehnt; Issuer lehnt Authentifizierung/Verifizierung ab und fordert, dass keine Autorisierung versucht wird
8	I	Nur zur Information; 3DS Requestor Challenge-Präferenz anerkannt

6.2.2 transStatusReason

	transStatusReason	Beschreibung
1	01	KartenAuthentifizierung gescheitert
2	02	Unbekanntes Gerät
3	03	Nicht unterstütztes Gerät
4	04	Überschreitet das Limit für die Authentifizierungshäufigkeit
5	05	Abgelaufene Karte
6	06	Ungülstige Kartennummer
7	07	Ungültige Transaktion
8	08	Kein Kartendatensatz
9	09	Sicherheitsfehler
10	10	Gestohlene Karte
11	11	Betrugsverdacht
12	12	Transaktion für den Karteninhaber nicht erlaubt
13	13	Karteninhaber nicht für den Service angemeldet
14	14	Zeitüberschreiung der Transaktion am ACS
15	15	Geringes Vertrauen
16	16	Mittleres Vertrauen
17	17	Hohes Vertrauen
18	18	Sehr hohes Vertrauen
19	19	Überschreitet das ACS Maximum der Challenges
20	20	Nicht zahlungswirksame Transaktion nicht unterstützt
21	21	3RI-Transaktion nicht unterstützt
22	22	ACS technisches Problem
23	23	Entkoppelte Authentifizierung vom ACS gefordert, aber vom 3DS Requestor nicht angefragt
24	24	3DS Requestor maximale Ablaufzeit bei Entkopplung überschritten
25	25	Entkoppelte Authentifizierung hatte unzureichend Zeit für die Authentifizierung des Karteninhabers. ACS macht keinen Versuch
26	26	Authentifizierung versucht, aber vom Karteninhaber nicht ausgeführt

7. Begriffe und Definitionen

7.1 Obligatorische und bedingte Datenelemente

Bedingungen in 3DS 2.0 sind oft beschrieben als *'Erforderlich, sofern nicht Markt- oder regionale Mandate das Senden dieser Informationen beschränken. '*
Das betrifft zum Beispiel die E-Mail-Adresse des Karteninhabers. Beachten Sie bitte, dass einige Anbieter von ACS-Software sowie manche Issuer diese Datenelemente innerhalb des EWR als technische obligatorisch ansehen können, da derzeit keine Beschränkungen bekannt sind. Daher ist es dringend empfohlen, diese Datenelemente falls möglich zu übermitteln.

7.2 Bedingungs-Codes

Code	Bedingung
M	**Pflicht** (engl.: mandatory). Bedeutet, dass das Datenelement in der Nachricht enthalten sein soll.
O	**Optional**. Das Datenelement kann oder kann nicht in einer Nachricht enthalten sein.
C	**Bedingt** (engl.: conditional). Das Datenelement soll enthalten sein (d.h. obligatorisch), wenn angegebene Bedingungen erfüllt sind.

7.3 Definitionen

Begriff	Definition
Autorisierung	Eine Autorisierung ist eine Genehmigung oder Garantie von Geldmitteln durch den Kartenaussteller an den Acquirer.
Autorisierungs-Avis	Der Acquirer informiert den Kartenaussteller über eine bereits gegebene Autorisierung (z.B. Stimmenautorisierung).
Buchung	Buchung ist der Prozess des Verbindes von genehmigtem Betrag und Autorisierungs-Code einer Transaktion zur Umwandlung in einen abrechenbaren Transaktions-Datensatz. Im Wesentlichen ist es die Anweisung, die Geldmittel vom Konto des Schuldners abzuziehen. Der Acquirer übermittelt nromalerweise eine Buchungsdatei an das Kartennetzwerk (duales Nachrichtensystem). In Systemen mit Host-Buchung sendet der Händler normalerweise eine Nachricht Buchungs-Avis an den annehmenden Host. Bei Systemen mit Terminal-Buchung übermittelt entweder der Kartenakzeptant (z.B. Händler) eine Buchungsdatei (am gebräuchlichsten) zum Acquirer oder führt einen Batch-Upload aus. Die vom Kartenakzeptanten übermittelten Buchungsdatensätze werden üblicherweise beim annehmenden Host validiert und dann zur Buchungsdatei des Acquirers für das entsprechenden Kartennetzwerk hinzugefügt.
Verkauf	Ein Verkauf ist eine Anweisung vom Händler an den Acquirer, in einer einzelnen Nachricht eine Autorisierung sowie eine Buchung der am Verkaufspunkt abgeschlossenen Transaktion anzufordern. Das bedeutet, eine erfolgreiche Autorisierung wird automatisch ohne weitere Anweisungen oder Abschlussnachrichten zur Buchungsdatei des Acquirers hinzugefügt. Einige Systeme mit Terminal-Buchung können jedoch erfordern, dass Verkaufs-Transaktionen in die Buchungsdatei oder in den Bacth-Upload einbezogen werden.
Terminal-Buchung	Terminal-Buchung bedeutet, dass das Terminal Autorisierungen, Verkäufe und Stornierungen den ganzen Tag über an den Host übermittelt. Das Terminal speichert alle diese Transaktionen ebenso wie alle lokal (offline) ausgeführten Transaktionen, so dass das Terminal am Ende des Verarbeitungstages vom Händler eine Batch-Übermittlung ausführen kann. Verarbeitung per Terminal-Buchung gibt dem Händler die Möglichkeit, Offline-Transaktionen auszuführen, die nur im Batch enthalten sind. Offline-Transaktionen umfassen zum Beispiel Rückgaben, Zwischenverkäufe und Trinkgeldkorrekturen.

Begriff	Definition
Host-Buchung	Bei der Verarbeitung mit Host-Buchung werden die Transaktions-Batches vom annehmenden Host verwaltet. Händler übermitteln die Transaktionen so zum Host, wie sie am Verkaufspunkt erfolgen, und der Host zeichnet die Transaktionen in einer Batch-Datei auf. In Nachrichtenprotokollen auf Basis von ISO 8583 wird das oft als Buchungs-Avis bezeichnet. Die Batchdatei wird dann entweder auf Anforderung vom Händlersystem (manuelle Batch-Freigabe) oder jeden Tag zu einer geplanten Zeit geschlossen (Host Auto-close). Die Option Auto-close ist am gebräuchlichsten.
Wiederkehrend	Wiederkehrende Transaktionen sind eine Reihe von Transaktionen, die nach einer Vereinbarung zwischen dem Karteninhaber und einem Händler verarbeitet werden, wobei der Karteninhaber über einen Zeitraum Waren oder Dienstleistungen in einer Reihe separater Transaktionen erwirbt.

8. Syntax

EVO E-PAY Antwort-Codes sind 8-stellig und gemäß der nachstehend beschriebenen Syntax aufgebaut.
Format: N8, (NNNNNNNN)

- N (Status)
- NNN (Kategorie)
- NNNN (Detail)

8.1 Beispiel

22060203

- 2 Fehler
- 206 3DS Kreditkartenadapter für Autorisierungs-Protokoll GICC
- 0203 Kartenmarke unterstützt kein 3DS

8.2 Status-Codes (1)

Code	Bedeutung	Beschreibung
0	Ok	Operation erfolgreiche abgeschlossen
2	Fehler	Operation gescheitert
4	Fataler Fehler	Operation gescheitert und verarbeitete Daten möglicherweise verloren
6	Andauernd / Transient	Operation ist nicht abgeschloissen. Der endgültige Status wird asynchron übertragen.
7	EMV 3DS Info	Zwischenzustände in der EMV 3DS Sequenz

8.3 Kategorie (2-4)

Der Kategorie-Code ist ein 3-stelliger Wert, der den Zahlungs-Adapter oder das Zahlungs-Protokoll angibt. Für 3DS 2.0 liegen diese Kategorie-Codes abhängig vom Kartenverbinder im Bereich von 100 bis 299.

8.4 Detail (5-8)

Status	Kategorie	Detail	Beschreibung
2	xxx	0101	Empfangene Nachricht ungültig
2	xxx	0102	Nachrichten-Versionsnummer nicht unterstützt
2	xxx	0103	Limit gesendete Nachrichten erreicht. Nur für PReq verwendet
2	xxx	0201	Erforderliches Element fehlt
2	xxx	0202	Critical message extension not recognized
2	xxx	0203	Format bei einem oder mehreren Elementen ist gemäß Spezifikationen ungültig
2	xxx	0204	Doppeltes Datenelement
2	xxx	0301	Transaktions-ID nicht erkannt
2	xxx	0302	Fehler der Datenentschlüsselung
2	xxx	0303	Zugriff verweigert, ungültiger Endpunt
2	xxx	0304	ISO-Code ungültig
2	xxx	0305	Transaktionsdaten ungültig
2	xxx	0306	Händler-Kategoriecode ist für das Zahlungssystem ungültig
2	xxx	0307	Seriennummer ungültig
2	xxx	0402	Zeitüberschreitung der Transaktion
2	xxx	0403	Kurzzeitiger Systemausfall
2	xxx	0404	Permanenter Systemausfall
2	xxx	0405	Systemverbindungsfehler
2	xxx	0911	Spezifischer Fehlercode von UnionPay. Vorhanden, wenn die Datenfelder-Relevanzprüfung scheitert (ECI-Wert und AV-Erscheinungsbild sind inkonsistent mit dem Transaktionsstatus).
2	xxx	0912	Spezifischer Fehlercode von UnionPay. Vorhanden bei duplizierter Transaktions-ID (Die Transaktions-ID sollte für alle AReq-Anfragen eindeutig sein).
2	xxx	0985	3DS 2.0 wird von der Karte nicht unterstützt. Der Händler muss dem Fallback-Prozess folgen.
2	xxx	3002	Ungültiger Parameter BROWSERINFO
2	xxx	3006	Ungültiger Parameter BILLINGADDRESS
7	000	0000	3DS 2.0 Versionierung erfolgreich
7	000	0001	Authentifizierungs-Antwort --> Challenge vorgeschrieben

9. ECI Codes EN

Der ECI-Wert wird vom Issuer ACS bereitgestellt. Es gibt die Authentifizierungsstufe an, die für die Transaktion durchgeführt wurde. Der von der Authentifizierung empfangene ECI-Wert wird in der Autorisierungsanforderung weitergeleitet und bestimmt auch, ob eine Transaktion eine Haftungsumkehr (Liability Shift) erhält.

9.1 Visa

ECI	Beschreibung	3DS Version(en)	Merchant Liability Shift/ Händler-Haftungsumkehr
06	Der Händler hat versucht, den Karteninhaber zu authentifizieren	3DS 1.0	Ja

ECI	Beschreibung	3DS Version(en)	Merchant Liability Shift/ Händler-Haftungsumkehr
	Für 3DS 1.0.2 kann der ECI Wert 06 nach Ermessen des Issuers als Authentifizierungsantwort vom Issuer-ACS verwendet werden. Beispielsweise können Issuer, die eine risikobasierte Authentifizierung verwenden, einen ECI = 06 für eine Transaktion bereitstellen, für die keine Karteninhaber-Authentifizierung erforderlich ist. Dies wird auch als "frictionless" Authentifizierung bezeichnet. Diese Issuer können einen ECI = 05 für Transaktionen vorsehen, die erfolgreich zur Authentifizierung aufgefordert wurden. Bei 3DS 2.0 kann der ECI 06-Wert nur verwendet werden, um anzuzeigen, dass ein Händler versucht hat, den Karteninhaber zu authentifizieren.	EMV 3DS (2.0)	
05	1. Karteninhaber-Authentifizierung erfolgreich (dies schließt eine erfolgreiche Authentifizierung mit risikobasierter Authentifizierung und / oder einem dynamischen Kennwort ein)	3DS 1.0 EMV 3DS (2.0)	Ja
07	Nicht authentifizierte E-Commerce-Transaktion	3DS 1.0 EMV 3DS (2.0)	Nein

9.2 Mastercard

ECI	Beschreibung	3DS Version(en)	Merchant Liability Shift/ Händler-Haftungsumkehr
00	Nicht authentifizierte E-Commerce-Transaktion	3DS 1.0 EMV 3DS (2.0)	Nein
01	Der Händler hat versucht, den Karteninhaber zu authentifizieren und hat einen Authentifizierungswert (Accountholder Authentication Value (AVV)) erhalten.	3DS 1.0 EMV 3DS (2.0)	Ja
02	Karteninhaber-Authentifizierung erfolgreich (dies schließt eine erfolgreiche Authentifizierung mit risikobasierter Authentifizierung und / oder einem dynamischen Kennwort ein)	3DS 1.0 EMV 3DS (2.0)	Ja
04	Data share only: Nicht authentifizierte E-Commerce-Transaktion, aber der Händler hat beschlossen, Daten über den 3DS-Fluss mit dem Issuer zu teilen, um die Genehmigungsrate für die Autorisierung zu verbessern	EMV 3DS (2.0)	Nein
06	Acquirer Ausnahme	EMV 3DS (2.0)	Nein
07	Wiederkehrende Zahlungen können für die erste oder nachfolgende Transaktion gelten Wenn dieser Wert bei der initialen wiederkehrenden Zahlungen eingeht, hat der Händler eine Haftungsumkehr • Nachfolgende Transaktionen gelten als MIT und die Haftung verbleibt beim Händler	EMV 3DS (2.0)	Ja

10. 3DS 2.0 Händler-Anwendungsfälle & Testen von 3-D Secure 2.0

Was Sie in diesem Kapitel erwarten können?
Wegen der verschiendenen Szenarien, die mit 3-D Secure 2.X auftreten können, untergliedern wir das Kapitel in drei thematische Bereiche: 2. Allgemeine technische Anpassungen (für alle Händler relevant) 3. Anwendungsfälle für Transaktionskennzeichnung (unterschiedliche Behandlung je nach Händler-Szenario) 4. Test von 3-D Secure 2.X via EVO Payments

1. Allgemeine technische Anpassungen

Welche Anfragearten betrifft 3-D Secure 2.0/SCA?

- Die betroffenen Anfragearten sind: Autorisierung und Verkauf
- Buchung und Gutschrift sind von den Änderungen ausgenommen

Wie wird die Datenübertragung an/von EVO Payments in Zukunft aussehen?

- ANFRAGE: Während der Implementierung von 3-D Secure 2.0 und der nötigen Lieferung größerer Datenmengen empfehlen wir Ihnen, unserer Formulare via Form-POST Methode aufzurufen. Beachten Sie bitte, dass die Option iFrame weiterhin verfügbar ist. Der Hintergrund sind mögliche Browserbeschränkungen, die dazu führen können, dass der gesendete Datenstring abgeschnitten wird.
  Beispiel:

```
<body>
    <form action="https://spg.evopayments.eu/pay/payssl.aspx"   method="post" id="form1">
        <input type="hidden" name="Len" value="371" />
        <input type="hidden" name="Data" value=
        "EF98523E2F6DF933C6098284B9C885DDBE1D5E800862CB5214D7AAEE36B7BD99F3BD8A188E6EF1EC8004D9FFDD1F517778ACD97F693A
        523807ACC1C20BE2D75B6695045C0C87DA25794BFD4B9C6098284B9C885DDBE1D5E800862CB52A16B55552D3341B117AA379FCC871EA8
        70E25B07ABB04A083407259292080B35D417995E49AB36F1083E3D5B5CE0C275DBBE26607870FF822DF6B9734FD3072E2C196B1CA" />
        <input type="hidden" name="MerchantID" value="Ihre_MID" />
        <input type="submit" value="senden" />
    </form>
</body>
</html>
```

- ANTWORT:
  Beachten Sie bitte auch die Änderung der abschließenden Weiterleitung zu URLSuccess | URLFailure. Diese wird im Fall von Transaktionen mit 3-D Secure 2.0 als ein Body POST ausgeführt. Deshalb sollten Sie sowohl ein GET als auch eine POST Antwort an URLSuccess | URLFailure empfangen können.

Wie kann ich zwischen 3DS 1.0 und 2.0 wählen?

- WICHTIG: Um 3-D Secure 1.0 oder 3-D Secure 2.0 testen oder nutzen zu können, müssen wir 3-D Secure an EVO E-PAY in Ihrem Namen konfigurieren. Bitte wenden Sie sich an den zuständigen EVO Implementation-Manager, falls Sie diesen Prozess noch nicht begonnen haben.
- Standardmäßig erfolgt jede Zahlung gemäß dem Prozess von 3-D Secure 1.0.
- Wenn Sie das Verfahren für 3-D Secure 2.0 nutzen wollen, verwenden Sie bitte den Aufrufparameter MsgVer=2.0. Das gilt für Tests ebenso wie für die spätere produktive Nutzung.
  - Parameter: MsgVer
    Wert: 2.0

Die Verwendung von JSON-Objekten wird Pflicht

- Beachten Sie bitte, dass mit der Implementierung von 3-D Secure 2.0 eine obligatorische Erweiterung der vorhandenen Parameter verbunden ist. Aus diesem Grund erwartet und antwortet EVO E-PAY mit relevanten zusätzlichen Daten als JSON-Objekt.
  Das JSON-Objekt muss Base64-codiert sein und regulär zusammen mit allen anderen Parametern in den verschlüsselten Blowfish-Daten an EVO E-PAY übertragen werden.
- Bitte übermitteln Sie nur JSON-Objekte mit Werten. Leere oder mit Null gefüllte Objekte/Parameter führen zu einer Ablehnung.

JSON Beispiel-Anfrage

https://spg.evopayments.eu/pay/payssl.aspx?MerchantID=Generic3DSTest&len=1800&data=CDC44E5A9D2C8A559CEDF1CCA97C9FBD3D90E046BFBF96F06ADA9A00FB0BC3494317E8D924FF44729671B93348B477F880541ACFF12C8E3A868CD55FEA95219C245CF7F4716FCF3462167A8B63D11424FA7BD30891504F8465C56805975115EB71C0A04E5D7466D771495035749FFF94D3087529F578DEF518003EA1422F6DE7B7DFD78A0DD695550623A42BF41A422EC219012318FE26D2B757F12BDFE046EA4CB8D35079ABAB6859691FEE1B034834714950357 49FFF94D3087529F578DEF518003EA1422F6DE7D4E20259A484D23A9EFC7F4ADB209DD67D8EDE5BD2AC0CB2682D7CF26A6624A54BCF4E93219ADD89ABA6214820D4BAA5A9A184DD7F8AF3E2BE98C5B63113276B023B92DA5AADCCD7387B71B6651A0E7E4E42F8790122386AA9A184DD7F8AF3E23CFEC0086B59B6A9D98EB96DFDA496D97D85706A4A810056FE48AB878EFC1E976DB7504D402F4B96778B45ADE1DF3E217EFFBA566359677AB73F514F1E75F11DBE3E15983BA530E7D5B13A87D1A2ED19A9A184DD7F8AF3E21D32D652A71B2A49A58F3A30256097DA11388C26E7CBEB12E65B31C485C94DE8179CEACDE9237EF4C426A05E594E28069E10B19AE173D25A93A546845C5D78C44112031D6D5DE9B4ABA6214820D4BAA5A9A184DD7F8AF3E260C35EC59CD2FAA2435CD631BFC801AA7C72A1BAE39879C0BF733EDC45DD99F3A9A184DD7F8AF3E2DDA25A6458507ACE3B3CAAC3A4B293A9C6177F7F00EBFB6924050D9DF661DE8EC204863D819ABF9564498E9F2D72BEFF2E040214C4961D8737821BA1F638BE05FB01E1B382733FC42D6B04AB80D66218C75E691B9475C5F6CF13AD357057BC6B5864EE113DF2272EF6572101D5E45CB634F3E941FA7B3EA7E636EAEF751C67C82F8E8D9B618E69826221B2A42D7F694D9E10B19AE173D25A6EB48BD63BFFE0FAFC78722BD9FFA39623B5D40494B96D2A9E10B19AE173D25A188DA61C8E3401850C400A3144C3547808A0C82C7B8E9863D017852B02FBFE6D62983EBC372B1A8108D832C13F92E88535C213D0FDA1B1A5C426A05E594E28069E10B19AE173D25A92AD74641E23F21D1D66F1B352AF-CCD408B1727FACC2405AA9A184DD7F8AF3E29B3106F31EE7D473A854D99576FDD5620141A96DEF638FCE4362F90866AED8044E42F8790122386AA9A184DD7F8AF3E20F6BF2E070199426696A900FEEBC7848B6F72D445F2CB9F0ED160CC32B1A3C40C426A05E594E28069E10B19AE173D25A201E55FC81E8F7CD78FFD98E342897C11AB2BE505B3E8421C63E936DCCF29058C31D72A3697DA2C89EFC7F4ADB209DD67D8EDE5BD2AC0CB2682D7CF26A6624A54BCF4E93219ADD89ABA6214820D4BAA5A9A184DD7F8AF3E2BE98C5B63113276B023B92DA5AADCCD7387B71B6651A0E7E4E42F8790122386AA9A184DD7F8AF3E23CFEC0086B59B6A9D98EB96DFDA496D93F669AB8A34E11706F7B3F762241F749A9A184DD7F8AF3E286587E20CD9A354709F67B1501183CFC5D6FD3FD6E23B0D4FA9746B8925D4A4FA9A184DD7F8AF3E21D32D652A71B2A49A58F3A30256097DA11388C26E7CBEB120758D07B77A47DB34E359C7AE383D69BC426A05E594E28069E10B19AE173D25A93A546845C5D78C44112031D6D5DE9B4ABA6214820D4BAA5A9A184DD7F8AF3E260C35EC59CD2FAA2435CD631BFC801AA7C72A1BAE39879C0BF733EDC45DD99F3A9A184DD7F8AF3E2DDA25A6458507ACE3B3CAAC3A4B293A9C6177F7F00EBFB6924050D9DF661DE8EC204863D819ABF956449 8E9F2D72BEFF2E040214C4961D8737821BA1F638BE05FB01E1B382733FC46AA58C2847221D78069144B06DE3755A6C88EADD3B3FCCD6F6572101D5E45CB634F3E941FA7B3EA7B08783F57D9AD1BAB2071FAB9B93B3C13FF102AD44B6A493B5C341FB37BF525B0A0E4F490BE1D46A4C5B8F691A2020868119A0AEB9E9BCD4F9D783FEA316723E17976FBB4909040AE279D66AF13B8441582CB00BB30835AB6401E5CDDF295F533AEE31D2677314D288F2C15BFB16837EF4A779C1E39E4AA1CEE13FABDB2B89D9A7A89ED81EC005BCD416330CFCE5CF716A316FDF29A9CFF3F25490656C800BCA582CB00BB30835ABABD19D247E68289A52F1387D978126C967F9BBB890618AF5A0E5136C7DC2892D2460687217D2779B5836D3F1FFAE8F3B582CB00BB30835ABEE02C59E0AAF8C913339B61F9DDFB7DAC4FF2460869E4876C5DFD5D39E79330D427654226D9E37E72D7A4C332F59563DF70B3A840877E2B1BF739A2347A73347F7DA9F100EEBC189ADE92F98B

E65BCBE29FE1A3DFE89E44EE-
EBF9C902BBAA7C2F68CBC48C724B889A53EA148988B56CC52D52743C045F57844F6607DDEA75FE613E
AC80E2C02BCEA89B71E52E64D7538DC9B82EB2740B82C698F43B6A62D770935233D5F10E593D051951
1BAAD615B0035D7524B097C29BA39EBE-
BEDB93425EB7824B9CCDB1397E716993ED326500615B4B1853A59F760A0E06373BDFE1CC6695A93B15
851F56428&template=evo_responsive&language=en

```
MerchantID=Generic3DSTest

&MsgVer=2.0

&TransID=1234567890

&RefNr=20200124

&Amount=100

&Currency=EUR

&URLNotify=

https://www.shop.com

&URLSuccess=

https://www.shop.com

&URLFailure=

https://www.shop.com

&billToCustomer=ew0KICAgICJjb25zdW1lciI6IHsNCiAgICAgICAgInNhbHV0YXRpb24iO-
iAiTXIiLA0KICAgICAgICAiZmlyc3ROYW1lIjogIk5hcG9sZW9uIiwNCiAgICAgICAgImxhc3ROYW1lIjo-
gIkJvbmFwYXJ0ZSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc2OS0wOC0xNSINCiAgICB9LA0KI-
CAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cnlDb2RlIjogIjMzIiwNCiAgICAgICAgI-
CAgInN1YnNjcmliZXJOdW1iZXIiIDogIjEyMzQ1Njc4OTEwIg0KICAgIH0sDQogICAgImVtYWlsIjo-
gIm5hcG9sZW9uLmJvbmFwYXJ0ZUBmcmFuY2UuY29tIg0KfQ==

&billingAddress=ew0KICAgICJjaXR5IjogIkFqYWNjaW8iLA0KICAgICJjb3VudHJ5Ijogew0KICAgI-
CAgICAiY291bnRyeUEzIjogIkZSQSINCiAgICB9LA0KICAgICJhZGRyZXNzTGluZTEiOiB7DQogICAgI-
CAgICJzdHJlZXQiOiAiRXhpbGVzdHJlZXQiLA0KICAgICAgICAic3RyZWV0TnVtVmVyIjogIjI3MCINCi-
AgICB9LA0KICAgICJwb3N0YWxDb2RlIjogIjIwMTY3Ig0KfQ==

&shipToCustomer=ew0KICAgICJjb25zdW1lciI6IHsNCiAgICAgICAgInNhbHV0YXRpb24iO-
iAiTXIiLA0KICAgICAgICAiZmlyc3ROYW1lIjogIkx1ZHdpZyIsDQogICAgICAgICJsYXN0TmFt-
ZSI6ICJYVklJSSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0xMS0xNyINCiAgICB9LA0KI-
CAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cnlDb2RlIjogIjMzIiwNCiAgICAgICAgI-
CAgInN1YnNjcmliZXJOdW1iZXIiIDogIjEyMzQ1Njc4OTEwIg0KICAgIH0sDQogICAgImVtYWlsIjo-
gIkx1ZHdpZ0Byb3lhbC5mcmFuY2UuY29tIg0KfQ==

&shippingAddress=ew0KICAgICJjaXR5IjogIlBhcmlzIiwNCiAgICAiY291bnRyeSI6IHsNCiAgICAgI-
CAgImNvdW50cnlBMyI6ICJGUkEiDQogICAgfSwNCiAgICAiYWRkcmVzc0xpbmUxIjogew0KICAgICAgI-
CAic3RyZWV0IjogIlBsYWNlIGRlIGxhIENvbmNvcmRlIiwNCiAgICAgICAgInN0cmVldE51bWJlciI6ICI-
xIg0KICAgIH0sDQogICAgInBvc3RhbENvZGUiOiAiNzUwMDEiDQp9

&credentialOnFile=ew0KICAgICJ0eXBlIjogew0KICAgICAgICAidW5zY2hlZHVsZWQiOiAiQ0lUIg0KI-
CAgIH0sDQogICAgImluaXRpYWxQYXltZW50IjogdHJ1ZQ0KfQ==

&OrderDesc=Test:0000

billToCustomer=ew0KICAgICJjb25zdW1lciI6IHsNCiAgICAgICAgInNhbHV0YXRpb24iOiAiTXIiLA0KI-
CAgICAgICAiZmlyc3ROYW1lIjogIk5hcG9sZW9uIiwNCiAgICAgICAgImxhc3ROYW1lIjo-
gIkJvbmFwYXJ0ZSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc2OS0wOC0xNSINCiAgICB9LA0KI-
```

CAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cnlDb2RlIjogIjMzIiwNCiAgICAgI-
CAgInN1YnNjcmliZXJOdW1iZXIiIDogIjEyMzQ1Njc4OTEwIg0KICAgIH0sDQogICAgImVtYWlsIjo-
gIm5hcG9sZW9uLmJvbmFwYXJ0ZUBmcmFuY2UuY29tIg0KfQ==

{

"consumer":
{ "salutation": "Mr", "firstName": "Napoleon", "lastName": "Bonaparte", "birthDate":
"1769-08-15" }

,

"mobilePhone":
{ "countryCode": "33", "subscriberNumber" : "12345678910" }

,

"email": "napoleon.bonaparte@france.com"

}

billingAddress=ew0KICAgICJjaXR5IjogIkFqYWNjaW8iLA0KICAgICJjb3VudHJ5Ijogew0KICAgICAgI-
CAiY291bnRyeUEzIjogIkZSQSINCiAgICB9LA0KICAgICJhZGRyZXNzTGluZTEiOiB7DQogICAgI-
CAgICJzdHJlZXQiOiAiRXhpbGVzdHJlZXQiLA0KICAgICAgICAic3RyZWV0TnVtVyIjogIjI3MCINCiA-
gICB9LA0KICAgICJwb3N0YWxDb2RlIjogIjIwMTY3Ig0KfQ==

{

"city": "Ajaccio",

"country":
{ "countryA3": "FRA" }

,

"addressLine1":
{ "street": "Exilestreet", "streetNumber": "270" }

,

"postalCode": "20167"

}

shipToCustomer=ew0KICAgICJjb25zdW1lciI6IHsNCiAgICAgICAgIn-
NhbHV0YXRpb24iOiAiTXIiLA0KICAgICAgICAiZmlyc3ROYW1lIjogIkx1ZHdpZyIsDQogICAgI-
CAgICJsYXN0TmFtZSI6ICJYVklJSSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0xMS0xNyINCiA-
gICB9LA0KICAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cnlDb2RlIjogIjMzIiwNCiAgI-
CAgICAgInN1YnNjcmliZXJOdW1iZXIiIDogIjEyMzQ1Njc4OTEwIg0KICAgIH0sDQogICAgImVtYWlsI-
jogIkx1ZHdpZ0Byb3lhbC5mcmFuY2UuY29tIg0KfQ==

{

"consumer":
{ "salutation": "Mr", "firstName": "Ludwig", "lastName": "XVIII", "birthDate": "1755-
11-17" }

```
,

"mobilePhone":
{ "countryCode": "33", "subscriberNumber" : "12345678910" }

,

"email": "Ludwig@royal.france.com"

}

shippingAddress=ew0KICAgICJjaXR5IjogIlBhcmlzIiwNCiAgICAiY291bnRyeSI6IHsNCiAgICAgI-
CAgImNvdW50cnlBMyI6ICJGUkEiDQogICAgfSwNCiAgICAiYWRkcmVzc0xpbmUxIjogew0KICAgICAgI-
CAic3RyZWV0IjogIlBsYWNlIGRlIGxhIENvbmNvcmRlIiwNCiAgICAgI-
CAgInN0cmVldE51bWJlciI6ICIxIg0KICAgIH0sDQogICAgInBvc3RhbENvZGUiOiAiNzUwMDEiDQp9

{

"city": "Paris",

"country":
{ "countryA3": "FRA" }

,

"addressLine1":
{ "street": "Place de la Concorde", "streetNumber": "1" }

,

"postalCode": "75001"

}

credentialOnFile=ew0KICAgICJ0eXBlIjogew0KICAgICAgICAidW5zY2hlZHVsZWQi-
OiAiQ0lUIg0KICAgIH0sDQogICAgImluaXRpYWxQYXltZW50IjogdHJ1ZQ0KfQ==

{

"type":
{ "unscheduled": "CIT" }

,

"initialPayment": true

}
```

Schlüssel Parameter / Objekt

- Wenn Sie nicht Ihre eigene Vorlage verwenden, haben wir Ihre ersten Tests eine neue Vorlage. Sie müssen lediglich das "Template=evo_responsive" zu den verschlüsselten Daten hinzufügen und der vom Kunden eingegebene cardholderName wird automatisch von EVO Payments für den Prozess 3D 2.0 übernommen. Für das geplante / kommende Rollout von 3DS-2.0 wird EVO Payments die Standardvorlagen entsprechend anpassen und Ihnen zur Verfügung stellen.

- Wenn Sie Ihre eigene Händlervorlage verwenden und die Abfrage das Karteninhabers dort bisher nicht integriert ist, müssen Sie cardholderName selbst integrieren.
- Beispiel einer XSL-Datei:

```
<!-- Cardholdername -->
  <div class="row ccholder">
    <span class="label">
      <xsl:value-of select="EVOEPay/language/strCCHolder"/>
    </span>
    <div class="input">
      <input type="text" value="" id="creditCardHolder" name="creditCardHolder">
        <xsl:attribute     name="value"><xsl:value-of     select="EVOEPay/creditCard-
Holder"/></xsl:attribute>
      </input>
    </div>
  </div>
```

- Beispiel einer XML-Datei:

```
For each language used:

<strCCHolder>Cardholdername</strCCHolder>
```

- Für PaySSL.aspx ist der cardholderName ein Schlüssel-Wert-Paar.

JSON-Objekt – accountInfo

- Je mehr Daten Sie uns übermitteln, desto größer ist die Wahrscheinlichkeit, dass eine reibungslose Zahlungswbwicklung erfolgt.
  Sie sollten daher prüfen, welche Daten Sie bereits haben, und intern bestimmen, welche Daten Sie übertragen möchten.

JSON Objekt – customerInfo (billToCustomer | shipToCustomer)

- Beachten Sie bitte, dass die Übermittlung von Adressdaten für 3-D Secure 2.0 obligatorisch ist. WICHTIG: Wenn Lieferadresse und Rechnungsadresse nicht übereinstimmen, müssen beide Adressen übermittelt werden! Bei digitalen Gütern ist die Rechnungsadresse ausreichend.

JSON-Objekt – merchantRiskIndicator

- Wir empfehlen dringend, den merchantRiskIndicator (Liefermethode) zu übermitteln. Die Lieferart wird im JSON-Objekt merchantRiskIndicator im JSON-Parameter shippingAddressIndicator übermittelt.
  Das kann eine positive Auswirkung für eine reibungslose Zahlungsabwicklung haben.

2. Anwendungsfälle für Transaktions-Kennzeichnung

Szenario 01 – Kreditkarten-Einmalzahlung

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- Jede Zahlung ist eine Einmalzahlung und deshalb immer eine neue ausgelöste Zahlung
- Sie verwenden keine Pseudokartennummer zur Speicherung und Wiederverwendung der Kartendaten

Anmeldedaten sind hinterlegt (CoF)

- Sie müssen 3-D Secure verwenden
- Es sind keine weiteren Anpassungen nötig

Szenario 02 – Kreditkarten-Abonnements

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- Kunden schließen bei Ihnen ein Abonnement ab, dass IMMER denselben Betrag und das gleiche Zahlungsintervall hat

- Sie nutzen die Pseudokartennummer, um die Kartendaten zu speichern und wiederzuverwenden
- WICHTIG: Die folgende Anfangszahlung unterliegt der Haftungsverschiebung für Sie als Händler. Im Falle der nachfolgenden Zahlung verfällt diese jedoch, so dass es dort **keine** Haftungsverschiebung gibt.

Anmeldedaten sind hinterlegt (CoF) – Anfangszahlung des Abonnements

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Nötige Anpassungen:
  - Beispiel:
    - JSON-Objekt credentialOnFile mit JSON-Parameter recurring (3 Schlüssel enthalten)
    - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "true"
    -
    - Beispiel für Anfangszahlung des Abonnements:

```
{
  "type": {
    "recurring": {
      "recurringFrequency": 30,
      "recurringStartDate": "2019-09-14",
      "recurringExpiryDate": "2020-09-14"
    }
  },
  "initialPayment": true
}
```

Anmeldedaten sind hinterlegt (CoF) – Folgezahlung des Abonnements

- Gilt für Direct.aspx
- 3-D Secure ist **NICHT** obligatorisch
- Nötige Anpassungen:
  - Beispiel:
    - Senden Sie bitte die schemereferenceID der Anfangszahlung mit, sodass nachfolgende Systeme die beiden Transaktionen entsprechend verknüpfen können.
    - JSON-Objekt credentialOnFile mit JSON-Parameter recurring (3 Schlüssel enthalten)
    - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "false"
    - Beispiel für Folgezahlung des Abonnements:

```
{
  "type": {
    "recurring": {
      "recurringFrequency": 30,
      "recurringStartDate": "2019-09-14",
      "recurringExpiryDate": "2020-09-14"
    }
  },
  "initialPayment": false
}
```

Szenario 03 – Kreditkarte Wiederkehrende Zahlung / Anzahlung / Schlusszahlung

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- Die Kunden kaufen wiederholt in ihrem Shop ein und verwenden dieselben Kreditkartendaten
- Sie nutzen die Pseudokartennummer, um die Kartendaten zu speichern und wiederzuverwenden
- WICHTIG: Die folgende Anfangszahlung unterliegt der Haftungsverschiebung für Sie als Händler. Im Falle der nachfolgenden Zahlung verfällt diese jedoch, so dass es dort **keine** Haftungsverschiebung gibt.

Anmeldedaten sind hinterlegt (CoF) – Anfängliche wiederkehrende Zahlung

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Nötige Anpassungen:
    - Beispiel:
        - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "CIT"
        - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "true"
        - Beispiel einer anfänglichen wiederkehrenden Zahlung:

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": true
}
```

Anmeldedaten sind hinterlegt (CoF) – Folgende wiederkehrende Zahlung

- Gilt für Direct.aspx
- 3-D Secure ist NICHT obligatorisch
- Nötige Anpassungen:
    - Beispiel:
        - Senden Sie bitte die schemereferenceID der Anfangszahlung mit, sodass nachfolgende Systeme die beiden Transaktionen entsprechend verknüpfen können.
        - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "MIT"
        - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "false"
        - Beispiel für folgende wiederkehrende Zahlung:

```
{
  "type": {
    "unscheduled": "MIT"
  },
  "initialPayment": false
}
```

Szenario 04 – Kreditkarten-Kontoverifizierung

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- In diesem Szenario wollen Sie nur die Kreditkarte des Kunden validieren
- Sie verwenden die Pseudokartennummer, um die Kartendaten zu speichern und wiederzuverwenden
- WICHTIG: Derzeit und zukünftig wollen die Schemes/Kartenmarken die Händler daran hindern, die Validierung der Kartendaten mit einem Minimalbetrag durchzuführen (z.B. 1-Cent-Autosierung). Deshalb bieten Ihnen EVO E-PAY die entsprechende "NullWertAuthentifizierung" an. Dies erfolgt durch Übergabe des zusätzlichen Parameters "AccVerify" in den verschlüsselten Daten – für Details siehe Beispiel unten. Stellen Sie bitte sicher, dass Ihr Kreditkarten-Acquirer diese Funktion unterstützt.

Anmeldedaten sind hinterlegt (CoF) – Validation Request

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Nötige Anpassungen:
    - Beispiel:
        - Senden Sie bitte den Parameter AccVerify=Yes in den verschlüsselten Daten mit (weitere Details finden Sie bitte in unserem Programmierhandbuch)
        - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "CIT"
        - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "true"

- Beispiel der Kontoverifizierung:

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": true
}
```

Szenario 05 – Kreditkarten Token speichern / Formular vorausfüllen

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- Kunden kaufen in Ihrem Shop ein und Sie speichern die Kreditkartendaten in Form einer Pseudokarten-nummer
- Wenn der Kunde wiederkommt, füllen Sie das Kreditkartenformular mit den gespeicherten Daten vor

Anmeldedaten sind hinterlegt (CoF) – Anfangszahlung für Token-Speicherung

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Nötige Anpassungen:
  - Beispiel:
    - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "CIT".
    - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "true"
    - Beispiel Anfangszahlung für Token-Speicherung:

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": true
}
```

Anmeldedaten sind hinterlegt (CoF) – Folgezahlung für Token-Speicherung

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Nötige Anpassungen:
  - Beispiel:
    - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "CIT"
    - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "false"
    - Beispiel Folgezahlung für Token-Speicherung:

```
{
  "type": {
    "unscheduled": "CIT"
  },
  "initialPayment": false
}
```

Szenario 06 – Kreditkarte wiederkehrende Zahlung inkl. Haftungsverschiebung (z.B. Reisebranche)

- WICHTIG: Das folgende Szenario gilt nur für PCI-zertifizierte Systeme

- Es gibt mehrere Szenarien für die Reisebranche, bei denen wiederkehrende Zahlungen auch der Haftungsverschiebung unterliegen
- Beispiel:
    - Der Kunde bucht ein Hotelzimmer über eine Buchungsplattform, gibt seine Kartendaten ein und führt 3-D Secure 2.0 aus. Das wird über einen separaten PSP verarbeitet. Diese Transaktion dient nur zur Validierung der Kartendaten -NullWertAuthentifizierung-.
      Das führt zu einem Authentifizierungsstatus = CAVV, den die zentrake Buchungsplattform dann dem Hotelbetreiber meldet (und jedem anderen Dienstleister wie einer Autovermietung, Versicherung usw.). Der Hotelbetreiber macht eine Zahlung OHNE 3DS 2.0 über das EVO E Pay, fügt aber den CAVV und alle weiteren Daten hinzu. Die zweite Transaktion enthält auch die entsprechende Haftungsverschiebung.
- Grundlage dafür, dass das funktioniert und die Haftungsverschiebung erfolgt, ist die Übermittlung des Authentifizierungsstatus (CAVV). Dieser wird über eine sogenannte "Externe 3DS Authentifizierung" bestimmt. Zwei Schritt sind notwendig:
    a. Das externe Händlersystem, welches die erste Zahlung (AccVerify/NullWertAuthentifizierung) ausgelöst hat, speichert den Authentifizierungsstatus
    b. Nachfolgend kann eine wiederkehrende Zahlung über EVO E-PAY erfolgen. In diesem Fall muss der Händler sowohl das JSON-Objekt threeDSData in den JSON-Daten als auch die originalen Kartendaten der anfänglichen Authentifizierung (Card-JSON) einbeziehen. Die Kartendaten müssen deshalb zwecks PCI-Compliance in ihrer originalen Form von der Buchungsplattform an alle relevanten Dienstleister / Agenturen übermittelt werden.
    Für diesen Zweck erklärt ein separater Abschnitt die nötigen Schritte.
- Alle nötigen technischen Informationen sind im Abschnitt Mehrparteien-E-Commerce / Agentur-Modell zu finden.

Szenario 07 – Kreditkarten-MoTo (MailOrder / TelephoneOrder) via PaySSL

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an, die telefonisch erfasst wird.
- Die Kreditkartendaten werden in einer separaten Callenter-Anwendung eingegeben, die eine Zahlung über EVO E-PAY mittels PaySSL.aspx auslöst.
- Sie verwenden die Pseudokartennummer, um die Kartendaten zu speichern und wiederzuverwenden
- WICHTIG: MoTo-Zahlungen unterliegen nicht der Haftungsverschiebung, da 3-D Secure nicht möglich ist. (Out of Scope)

Anmeldedaten sind hinterlegt (CoF) – Anfängliche MoTo-Zahlung

- Gilt für PaySSL.aspx
- 3-D Secure ist nicht möglich (Out of Scope)
- Nötige Anpassungen:
    - Beispiel:
        - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "MIT"
        - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "true"
        - Beispiel einer anfänglichen MoTo-Zahlung:

```
{
  "type": {
    "unscheduled": "MIT"
  },
  "initialPayment": true
}
```

Anmeldedaten sind hinterlegt (CoF) – Folgende MoTo-Zahlung

- Gilt für die automatisierte Zahlungsauslösung via Direct.aspx
- 3-D Secure ist nicht möglich (Out of Scope)
- Nötige Anpassungen:

- Beispiel:
  - Senden Sie bitte immer die schemereferenceID der Anfangszahlung mit, sodass nachfolgende Systeme die beiden Transaktionen entsprechend verknüpfen können
  - JSON-Objekt credentialOnFile mit JSON-Parameter unscheduled und dem Wert "MIT"
  - JSON-Objekt credentialOnFile mit JSON-Parameter initialPayment und dem Wert "false"
  - Beispiel einer folgenden MoTo-Zahlung:

```
{
  "type": {
    "unscheduled": "MIT"
  },
  "initialPayment": false
}
```

Szenario 08 – Kreditkarten MoTo (MailOrder / TelephoneOrder) über Virtuelles Terminal

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an, die telefonisch erfasst wird.
- Die Kreditkartendaten werden über ein virtuelles Terminal eingegeben.
- WICHTIG: MoTo-Zahlungen unterliegen nicht der Haftungsverschiebung, da 3-D Secure nicht möglich ist. (Out of Scope)

Anmeldedaten sind hinterlegt (CoF)

- Durch die Verwendung des virtuellen Terminals sind keine weiteren Anpassungen erforderlich.

Szenario 09 – Batch-Verarbeitung

- Wegen der ständig nötigen Anpassungen im Bereich der Batch-Abläufe wenden Sie sich bei Fragen bitte an den zuständigen EVO Implementation-Manager.

Szenario 10 – Erweitertes Transaktions-Management (ETM)

- Bei Nutzung des EVO Payments ETMs kümmert sich EVO E-PAY für Sie um die richtige Markierung der Transaktionen.

3. Test von 3-D Secure 2.0 über EVO Payments

Nutzen Sie die Chance, jetzt 3-D Secure 2.0 zu testen!
Während derzeit nicht alle nachgelagerten Systeme 3-D Secure für Testzwecke anbieten, können Sie in EVO E-PAY eine Testsimulation durchführen. Das ermöglicht Ihnen die Durchführung von 3-D Secure Authentifizierungen mit verschiedenen Rückgabewerten.
Gehen Sie für Tests bitte folgendermaßen vor:

- Aktivieren Sie 3-D Secure 2.0 für Ihre EVO Payments MerchantID. Wenn Sie unsicher sind, ob das bereits aktiviert ist, wenden Sie sich bitten an den zuständigen EVO Implementation-Manager.
- In der verschlüsselten Datenanfrage verwenden Sie den Standardparameter OrderDesc mit dem Wert "Test:0000". Das gibt Ihnen eine entsprechend erfolgreiche Autorisierung nach einer erfolgreichen Authentifizierung.
  WICHTIG: Im Simulationsmodus wird die schemereferenceID der anfänglichen Zahlung nicht zurückgegeben, weil diese ID von den nachgelagerten Systemen generiert wird. Diese Systeme sind bei der Simulation nicht beteiligt.
- Führen SIe die 3-D Secure Authentifizierung durch
- Verwenden SIe bitte **NUR** die verfügbaren Testkarten (Ablaufdatum immer in der Zulunft + CVV/CVC kann jeden Wert enthalten)
- Je nach gewünschtem Szenario (z.B. Browser 3-D Secure 2.0 Challenge, reibungslose Browser-Authentifizierung usw.) verwenden SIe bitte die entsprechenden Einmal-Passwörter