

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

Was Sie in diesem Kapitel erwarten können?

Wegen der verschiedenen Szenarien, die mit 3-D Secure 2.X auftreten können, untergliedern wir das Kapitel in drei thematische Bereiche:

1. Allgemeine technische Anpassungen (für alle Händler relevant)
2. Anwendungsfälle für Transaktionskennzeichnung (unterschiedliche Behandlung je nach Händler-Szenario)
3. Test von 3-D Secure 2.X via Computop

Allgemeine technische Anpassungen

Welche Anfragearten betrifft 3D-Secure 2.0/SCA?

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

- Allgemeine technische Anpassungen
 - Welche Anfragearten betrifft 3D-Secure 2.0/SCA?
 - Wie sieht die Datenübertragung mit 3-D Secure 2.x aus?
 - Wie kann ich zwischen 3DS 1.0 und 2.0 wählen?
 - Die Verwendung von JSON-Objekten wird Pflicht
 - JSON Beispiel-Anfrage - verschlüsselt es "data"
 - JSON Beispiel-Anfrage - unverschlüsselte Request-Parameter
 - Schlüssel Parameter / Objekt
- Anwendungsfälle für Transaktions-Kennzeichnung
 - Szenario 01 – Kreditkarten-Einmalzahlung
 - Szenario 02 – Kreditkarten-Abonnements
 - Szenario 03 – Kreditkarte Wiederkehrende Zahlung / Anzahlung / Schlusszahlung
 - Szenario 04 – Kreditkarten-Kontoverifizierung
 - Szenario 05 – Kreditkarten Token speichern / Formular vorausfüllen – PayNow-Schnittstelle
 - Szenario 05 – Kreditkarten Token speichern / Formular vorausfüllen – PaySSL-Schnittstelle
 - Szenario 06 – Kreditkarte wiederkehrende Zahlung inkl. Haftungsverschiebung (z.B. Reisebranche)
 - Szenario 07 – Kreditkarten-MoTo (MailOrder / TelephoneOrder) via PaySSL, Direct oder PayNow
 - Szenario 08 – Kreditkarten MoTo (MailOrder / TelephoneOrder) über Virtuelles Terminal

- Szenario 09 – Batch-Verarbeitung
- Szenario 10 – Erweitertes Transaktions-Management (ETM)
- Test von 3-D Secure 2.x

Wie sieht die Datenübertragung mit 3-D Secure 2.x aus?

- 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://www.computop-paygate.com/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 unserem Paygate in Ihrem Namen konfigurieren. Bitte wenden Sie sich an [Computop Helpdesk](#), 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 das Paygate 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 des Computop Paygate übertragen werden.
- Bitte übermitteln Sie nur **JSON-Objekte** mit Werten. Leere oder mit Null gefüllte Objekte/Parameter führen zu einer Ablehnung.

JSON Beispiel-Anfrage - verschlüsseltes "data"

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

```

MerchantID=Generic3DSTest&len=1800&data=CDC44E5A9D2C8A559CEDF1CCA97C9FBD3D90E046BFBF96F06ADA9A00FB0BC3494317E8D92
4FF44729671B93348B4717F805441ACFF12C8E3A868CD5FEA95219C25CF7F4126FCF34621677A8B63D112424FA7BD30B891504F8465C5680597
5115EB71C0A0E5D7466D771495035749FF94D3087529F58F5E18003EA1422F6DE7B7DF78A0D69550623A4B2BF41A422CE219012318
FE26D2B757F12BDFE046EA4CB8D35079ABAB6859691FEE1B03483471495035749FFF94D3087529F578DEF518003EA1422F6DE7D4E20259A4
84D23A9EFC7F74ADB209DD67D8DEDE5BD2AC0CB2682D7CF26A662A454BCF4E93219ADD89ABA621482D04BA5A9A184DD7F8AF3E2BCE98C
5B631132768023B92DA5AACDDC7387B71B6651A0E7E4E2F8790122386AA9A184DD7F8AF3E23CFEC0086B59A9D98B96FDA496D97D
85706A4A810056FE48AB878EFC1E976DB7504D02F4B96778B45ADE1DF3E217FEFFBA566359677AB73F514F1E75F11DBE3E15983B450E7D
5B13A87D1A2ED19A9A184DD7F8AF3E21D32D652A71B2A49A58F3A30256097DA11388C26E7CBEB12E65B31C48DC94DE8179CEACDE9237E
F4A426A05E594E28069E10B19AE173D25A93A546845C5D78C44112031D6D5DE9B4ABA621482D04BA5A9A184DD7F8AF3E260C35EC59CD2
FAA2435CD631BFC801AA7C72A1BAE39879C0BF733EDC45DD99F3A9A184DD7F8AF3E2DDA25A6458507ACE3B3CAAC3A4B293A9C6177F7F
00EBFB6924050D9DF661DE8EC204863D819ABF9564498E9F2D72BEFF2E040214C4961D8737821BA1F638BE05FB01E1B382733FC42D6B04A
B80D66218275691B9475C5FCF13AD350757BCB58564EE113DF2272EF657210D5E45CEB34F3E941FA7B3EA7E636AE9F751C67C2F8E8
D9B618E69826221D4A2D7F694D9E10B19AE173D25A93A546845C5D78C44112031D6D5DE9B4ABA621482D04BA5A9A184DD7F8AF3E260C35EC59CD2
DA61C8E3401850C400A3144C3547808A0C82C7B8E9863D017852B02FBFE6D62983EBC372B1A8108D832C13F92E88535C213D0FDA1B1A5C4
26A05E594E28069E10B19AE173D25A92AD74641E23F21D1D66F1B352AFCC4D08B1727FAC22045A9A184DD7F8AF3E29B3106F31EE7D473
A854D99576DFD5620141A96DEF638FE4362F90866AED8044E2F8790122386AA9A184DD7F8AF3E20FBEF2E070199426696A900FE05BC7848
B6F72D445F2CB9F0ED160CC32B1A3C40C426A05E594E28069E10B19AE173D25A201E55FC81E8F7CD78FFD98E342897C11AB2B0E00E5B3E84
21C63E936DCCF29058C31D72A3697DA2C89EFC7F4ADB209DD67D8DEDE5BD2AC0CB2682D7CF26A662A454BCF4E93219ADD89ABA621482D
04BA5A9A184DD7F8AF3E2BCE985B63113276B023B92DA5AACDDC7387B71B6651A0E7E4E2F8790122386AA9A184DD7F8AF3E23CFEC00
86B59B6A9D98B96FDA496D9E3F669AB8A34A11706F73B7622417F49A9A184DD7F8AF3E286587E20CD9A354709F67B1501183C25D6FD3
FD6E23B0D4FA9746B8925D4A4FA9A184DD7F8AF3E21D32D652A71B2A49A58F3A30256097DA11388C26E7CBEB120758D07B77A47DB34E35
9C7AE383D69BC426A05E594E28069E10B19AE173D25A93A546845C5D78C44112031D6D5DE9B4ABA621482D04BA5A9A184DD7F8AF3E260
C35EC59CD2FAA2435CD631BFC801AA7C72A1BAE39879C0BF733EDC45DD99F3A9A184DD7F8AF3E2DDA25A6458507ACE3B3CAAC3A4B29
3A9C6177F7F00EBFB6924050D9DF661DE8EC204863D819ABF9564498E9F2D72BEFF2E040214C4961D8737821BA1F638BE05FB01E1B38273
3FC46AA582847221D78069144B06DE3755A6C88EAD43B3FC6D6657210D5E45CEB34F3E941FA7B3EA7E636AE9F751C67C2F8E8
B93B3C13FF102424BA46B493B3C341FB37BF525B0A0D490CBE1D4664520B86F691A2020868119A0AEB9E9BCD4F9D783FEC316723E1797FB
B4909040AE279D66AF138B441582CB00BB30835AB6401E5CDDF29F5533AEE31D2677314D288F2C15BF816837FE4779C1E39E4AA1CEE13
FABDB2B89D9A7A89ED81EC005BCD416330CFCE5CF716A316FDF29A9CFF3F25490656C800BCA582CB00BB30835ABABD19D247E68289A52
F1387D978126C967F9BB8B90618FA5A0E5136C7D2C892D46068721D72779B5836D3CF1FFAE8F3B582CB00BB30835ABABED02C59E0AAF8C913
339B619FDDFB7D4AC4FF2460869E4876C5DFD5D39E79330D427654226D9E37E72D7A4332F39563D7F0B3A840877E2B1BF739A2347473347
F7DA9F100EEBC189ADE92F98BE65BCBE29DFE1A3DFE89E44EEEBF9C902DBAA7C2F68CB48C724B889A53EA148988B56CC52D52743C045
F57844F6607DDEA75FE613EAC80E2C02BCEA89B71E52E64D7538DC9B82EB2740B82C698F43B6A62D770935233D5F10E593D0519511BAAD
615B0035D7524B097C29BA39EBEBEDB93425EB7824B9CCDB1397E716993ED326500615B4B1853A59F760A0E06373BDFE1CC6695A93B1585
1F56428&template=ct_responsive_ch&language=en

```

```
MerchantID=Generic3DSTest
&MsgVer=2.0
&TransID=1234567890
&RefNr=20200124
&Amount=100
&Currency=EUR
&URLNotify=https://www.computop.com
&URLSuccess=https://www.computop.com
&URLFailure=https://www.computop.com
&billToCustomer=ew0KICAgICJjb25zdW1lc iI6IHsNCiAgICAgICAgInNh bHV0YXRpb24iOiAiTXIiLA0KICAgICAgICAiZmlyc3R0YW1lIjogIk5hcG9sZW9uIiwNCiAgICAgICAgImxhc3R0YW1lIjogIk JvbWFWYXJ0ZSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0wOC0xNSINCiAgICB9LA0KICAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cn lDb2RlIjogIjMzMCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIj IwMTY3Ig0KFq==&billingAddress=ew0KICAgICJjaXR5IjogIkFyQW Njaw8iLA0KICAgICJjb3VudHJ5Ijogew0KICAgICAgICAiY291bnRyeUEzIjogIkZSQ SINCiAgICB9LA0KICAgICJhZGRyZXNZTGluzTEiOiB7DQogICAgICAgICJzdHJlZXQiOiAiRXhp bGVzdHJlZXQiLA0KICAgICAgICAic3RyZWV0TnVtYmVyIjogIjI3MCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIjIwMTY3Ig0KFq==&shipToCustomer=ew0KICAgICJjb25zdW1lc iI6IHsNCiAgICAgICAgInNh bHV0YXRpb24iOiAiTXIiLA0KICAgICAgICAiZmlyc3R0YW1lIjogIkx1ZHdpZyIsDQogICAgICAgICJsyXN0TmFtZSI6ICJYVk lJSSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0xMS0xNyINCiAgIC B9LA0KICAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cn lDb2RlIjogIjMzMCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIj IwMTY3Ig0KFq==&shippingAddress=ew0KICAgICJjaXR5IjogIl BhcmllziIiwNCiAgICAiY291bnRyeSI6IHsNCiAgICAgICAgImNvdW50cn lBMiY6ICJGUKEiDQogICAgfSwNCiAgICAiYWRkc mVzc0xpbmUxIjogew0KICAgICAgICAic3RyZWV0IjogIlBsYWNlIGRlIG xhIE NvbmNvcmlIiWNCiAgICAgICAgInN0cmVldE5lbWJlc iI6ICxiG0KICAgIH0sDQogICAgInBvc3RhbnVzGUiOiAiZnUwMDEIQP9 &credentialOnFile=ew0KICAgICJ0eXBFIjogew0KICAgICAgICAidW5zY2hl ZHVzZWQiOiAiQ0lUig0KICAgIH0sDQogICAgImluaXRPYXw QYXltZW50IjogdHJ1ZQ0KFq==&OrderDesc=Test:0000
```

where:

```
biillToCustomer=ew0KICAgICJjb25zdW1lc iI6IHsNCiAgICAgICAgInNh bHV0YXRpb24iOiAiTXIiLA0KICAgICAgICAiZmlyc3R0YW1lIjogIk5hcG9sZW9uIiwNCiAgICAgICAgImxhc3R0YW1lIjogIk JvbWFWYXJ0ZSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0wOC0xNSINCiAgICB9LA0KICAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cn lDb2RlIjogIjMzMCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIj IwMTY3Ig0KFq==&billingAddress=ew0KICAgICJjaXR5IjogIkFyQW Njaw8iLA0KICAgICJjb3VudHJ5Ijogew0KICAgICAgICAiY291bnRyeUEzIjogIkZSQ SINCiAgICB9LA0KICAgICJhZGRyZXNZTGluzTEiOiB7DQogICAgICAgICJzdHJlZXQiOiAiRXhp bGVzdHJlZXQiLA0KICAgICAgICAic3RyZWV0TnVtYmVyIjogIjI3MCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIjIwMTY3Ig0KFq==&shipToCustomer=ew0KICAgICJjb25zdW1lc iI6IHsNCiAgICAgICAgInNh bHV0YXRpb24iOiAiTXIiLA0KICAgICAgICAiZmlyc3R0YW1lIjogIkx1ZHdpZyIsDQogICAgICAgICJsyXN0TmFtZSI6ICJYVk lJSSIsDQogICAgICAgICJiaXJ0aERhdGUiOiAiMTc1NS0xMS0xNyINCiAgIC B9LA0KICAgICJtb2JpbGVQaG9uZSI6IHsNCiAgICAgICAgImNvdW50cn lDb2RlIjogIjMzMCINCiAgICB9LA0KICAgICJwb3N0YXwxDb2RlIjogIj IwMTY3Ig0KFq==&shippingAddress=ew0KICAgICJjaXR5IjogIl BhcmllziIiwNCiAgICAiY291bnRyeSI6IHsNCiAgICAgICAgImNvdW50cn lBMiY6ICJGUKEiDQogICAgfSwNCiAgICAiYWRkc mVzc0xpbmUxIjogew0KICAgICAgICAic3RyZWV0IjogIlBsYWNlIGRlIG xhIE NvbmNvcmlIiWNCiAgICAgICAgInN0cmVldE5lbWJlc iI6ICxiG0KICAgIH0sDQogICAgInBvc3RhbnVzGUiOiAiZnUwMDEIQP9 &credentialOnFile=ew0KICAgICJ0eXBFIjogew0KICAgICAgICAidW5zY2hl ZHVzZWQiOiAiQ0lUig0KICAgIH0sDQogICAgImluaXRPYXw QYXltZW50IjogdHJ1ZQ0KFq==&OrderDesc=Test:0000
```

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

is base64-encoding of

```
{
  "city": "Ajaccio",
  "country": { "countryA3": "FRA" },
  "addressLine1": { "street": "Exilestreet", "streetNumber": "270" },
  "postalCode": "20167"
}
```

is base64-encoding of

```
{
  "consumer": { "salutation": "Mr", "firstName": "Ludwig", "lastName": "XVIII", "birthDate": "1755-11-17" },
  "mobilePhone": { "countryCode": "33", "subscriberNumber" : "12345678910" },
  "email": "Ludwig@royal.france.com"
}
```

is base64-encoding of

```
{
  "city": "Paris",
  "country": { "countryA3": "FRA" },
  "addressLine1": { "street": "Place de la Concorde", "streetNumber": "1" },
  "postalCode": "75001"
}
```

is base64-encoding of

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

- Wenn Sie nicht Ihre eigene Vorlage verwenden, haben wir Ihre ersten Tests eine neue Vorlage. Sie müssen lediglich das "Template=ct_responsive_ch" zu den verschlüsselten Daten hinzufügen und der vom Kunden eingegebene **cardholderName** wird automatisch von Computop für den Prozess 3D 2.0 übernommen. Für das geplante / kommende Rollout von 3DS-2.0 wird Computop die Standardvorlagen entsprechend anpassen und Ihnen zur Verfügung stellen.
- Wenn Sie Ihre eigene Händlervorlage verwenden und die Abfrage des 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="paygate/language/strCCHolder" />
    </span>
    <div class="input">
      <input type="text" value="" id="creditCardHolder" name="creditCardHolder">
      <xsl:attribute name="value"><xsl:value-of select="paygate/creditCardHolder" />
    </div>
  </div>
</div>

```

- Beispiel einer XML-Datei:

For each language used:

```
<strCCHolder>Cardholdername</strCCHolder>
```

- Für PaySSL.aspx | PayNow.aspx ist der [cardholderName](#) ein Schlüssel-Wert-Paar.
- Für Direct.aspx ist der [cardholderName](#) ein JSON-Parameter des JSON-Objekts "[Card](#)".

JSON-Objekt – [browserInfo](#)

- Für PaySSL.aspx erfasst Computop die Browserdaten für Sie und daher müssen Sie nichts unternehmen.
- Bei einer Server-zu-Server-Verbindung per Direct.aspx oder der Verwendung von PayNow.aspx müssen die individuellen zusätzlichen Anfragen im Shop implementiert werden.

JSON-Objekt – [accountInfo](#)

- Je mehr Daten Sie uns übermitteln, desto größer ist die Wahrscheinlichkeit, dass eine reibungslose Zahlungsabwicklung 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.

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** Pseudokartenummer 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 Pseudokartenummer, 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 + PayNow.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 Pseudokartenummer, 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 + PayNow.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-Autorisierung). Deshalb bietet Ihnen das Computop Paygate die entsprechende "NullWertAuthentisierung" 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 + PayNow.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 – PayNow-Schnittstelle



Die im folgenden Verlauf beschriebenen Abläufe unterliegen der Haftungsverschiebung für Sie als Händler.

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an
- Kunden kaufen in Ihrem Shop ein und Sie speichern die Kreditkartendaten in Form einer Pseudokartenummer

A: Wenn der Kunde wiederkommt, füllen Sie das Kreditkartenformular mit den gespeicherten Daten vor. Im Falle von CIT mit initial=false wird das Flagging verwendet, wenn der Händler die Pseudokartenummer über eine Template-Prefill-Option dem Kunden vorbelegt.

B: Wenn der Kunde wiederkommt, füllen Sie das Kreditkartenformular mit den gespeicherten Daten vor. Falls der Kunde die bestehende Kartenummer löscht und eine neue hinterlegt oder ggfs. eine weitere Karte hinzufügt, dann muss das Flagging für CIT erneut verwendet werden (initial=true), sofern der Kunde diese Karte auch vorbelegt haben möchte.

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

- Gilt für PayNow.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 PayNow.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"
 - Senden Sie bitte die [schemereferenceId](#) der Anfangszahlung mit (COF-CIT-TRUE), sodass nachfolgende Systeme die beiden Transaktionen entsprechend verknüpfen können.
 - Beispiel Folgezahlung für Token-Speicherung:

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

Szenario 05 – Kreditkarten Token speichern / Formular vorausfüllen – PaySSL-Schnittstelle

Die verpflichtende Steuerung der initialen und wiederkehrenden Zahlungen übernimmt das Computop Paygate für Sie. Damit wir die Funktion für Sie freischalten können, wenden Sie sich bitte an den [Computop Helpdesk](#).

Bitte beachten Sie die nachfolgenden Voraussetzungen im Bezug auf händlereigene oder Computop Standard-Templates (ct_responsive, Cards_v1).

- Gilt für PaySSL.aspx
- 3-D Secure ist obligatorisch
- Die Haftungsumkehr unterliegt der Haftungsverschiebung für Sie als Händler
- Nötige Anpassungen (händlereigenes Template):
 - Bitte integrieren Sie eine Prefill-Checkbox anhand des im weiteren Verlauf gelieferten Code-Snippets
 - Bitte aktivieren (einkommentieren) Sie den im XSL-File enthaltenen Code zur Vorbelegung der Pseudokartenummer
 - Entscheidet sich der Kunde, die Kartendaten zu speichern (klickt die Checkbox), bekommt das Händlersystem den zusätzlichen, verschlüsselten Response-Parameter "prefill=on" übermittelt. Wird die Checkbox nicht ausgewählt, gibt es keine zusätzliche Benachrichtigung über den prefill Parameter.
 - Möchten Sie nun zum registrierten Kunden das Formular mit den Kartendaten vorausgefüllt darstellen, übergeben Sie bitte im verschlüsselten Data das [Card JSON Objekt](#) und übermitteln die vorliegende Pseudokartenummer (Token), Ablaufdatum, Kartenmarke und den Kreditkarteninhaber via JSON.
 - Bitte übergeben Sie für den initialen Request als auch zur weiteren vorausgefüllten Zahlung KEIN Credential On File flagging.
- Nötige Anpassungen (Computop-Standardformular):
 - Damit die Prefill-Checkbox auf dem Computop-Formular dargestellt wird, geben Sie uns bitte über den [Computop Helpdesk](#) Bescheid. Dann konfigurieren wir die komplette Funktion (Checkbox + COF-Steuerung) für Sie.
 - Bitte übergeben Sie für den initialen Request als auch zur weiteren vorausgefüllten Zahlung KEIN Credential On File flagging.

- Entscheidet sich der Kunde, die Kartendaten zu speichern (klickt die Checkbox), bekommt das Händlersystem den zusätzlichen, verschlüsselten Response-Parameter "prefill=on" übermittelt. Wird die Checkbox nicht ausgewählt, gibt es keine zusätzliche Benachrichtigung über den prefill Parameter.
- Möchten Sie nun zum registrierten Kunden das Formular mit den Kartendaten vorausgefüllt darstellen, übergeben Sie bitte im verschlüsselten Data das [Card JSON Objekt](#) und übermitteln die vorliegende Pseudokartennummer (Token), Ablaufdatum, Kartenmarke und den Kreditkarteninhaber via JSON

JavaScript

```
// Adds a Change-Event to the checkbox called 'cbPrefill'
$("#cbPrefill").change(function() {
    if ($("#cbPrefill").is(':checked'))
    { // Wenn die Checkbox aktiviert wurde, wird der Wert des Hiddenfields mit den Namen 'prefill' auf 'on'
      gesetzt   $("input[type='hidden'][name='prefill']").val('on'); }

    else
    { // Wenn die Checkbox deaktiviert wurde, wird der Wert des Hiddenfields mit den Namen 'prefill' wieder
      geleert   $("input[type='hidden'][name='prefill']").val(''); }

});
// In case of retries (If form gets called a second time due to errors),
// the last status will be set
if ($("input[type='hidden'][name='prefill']").val() == 'on') {
    $("#cbPrefill").attr('checked', true);
}
```

XSL

```
<!-- Hiddenfield, which tells Paygate that prefill function should get activated -->
<!-- value="on" means that Paygate will return 'prefill=on' in the response to merchant -->
<!-- value="" means that Paygate will not return 'prefill=on' in the response to merchant -->
<input type="hidden" name="prefill"><xsl:attribute name="value"><xsl:value-of select="paygate/prefill"/><
/xsl:attribute></input>

<!-- Checkbox to (de)activate prefill function -->
<div id="div_cbPrefill" class="div_cbPrefill">
    <input type="checkbox" name="cbPrefill" id="cbPrefill"></input>
    <span><xsl:value-of select="paygate/language/strCCSaveData" disable-output-escaping="yes"/></span>
    <div class="row"></div>
</div>
```

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 -NullWertAuthentisierung-.
- Das führt zu einem Authentisierungsstatus = CAVV, den die zentrale 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 Paygate, 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 Authentisierungsstatus (CAVV). Dieser wird über eine sogenannte "Externe 3DS Authentisierung" bestimmt. Zwei Schritte sind notwendig:
 1. Das externe Händlersystem, welches die erste Zahlung (AccVerify/NullWertAuthentisierung) ausgelöst hat, speichert den Authentisierungsstatus
 2. Nachfolgend kann eine wiederkehrende Zahlung über das Computop Paygate erfolgen. In diesem Fall muss der Händler sowohl das JSON-Objekt threeDSDData in den JSON-Daten als auch die originalen Kartendaten der anfänglichen Authentisierung (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, Direct oder PayNow

- Sie bieten Ihren Kunden die Zahlung per Kreditkarte an, die telefonisch erfasst wird.
- Die Kreditkartendaten werden in einer separaten Callcenter-Anwendung eingegeben, die eine Zahlung über das Computop Paygate mittels PaySSL.aspx, Direct.aspx oder Paynow.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, PayNow.aspx und Direct.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 weitere Anpassungen erforderlich.

Szenario 09 – Batch-Verarbeitung

Die Vorgabe zum korrekten Kennzeichnen von wiederkehrenden Transaktionen seitens der Kartenmarken ist schon eine ältere Anforderung, hierzu hatten wir im Januar 2019 entsprechend informiert. Im Zuge der PSD2-SCA Umsetzung wird diese vollumfänglich zur Pflicht, damit eben wiederkehrende Transaktionen eindeutig gekennzeichnet werden und somit die nachgelagerten Systeme erkennen, warum in diesen Fällen kein 3-D Secure verarbeitet wurde, da in diesen Fällen kein Endkunde aktiv am Payment teilnimmt. Somit ist eine Umsetzung für alle Händler verpflichtend. Nachfolgend finden Sie unsere Beschreibungen / Details dazu, d.h. im ersten Schritt muss die initiale Zahlung aus dem Shop heraus als CredentialOnFile gekennzeichnet werden und darauf basierend die wiederkehrenden Batch-Transaktionen.

***initiale Shop-Transaktion oder AccountVerification:

Auf unserer Anwendungsseite ist das korrekte initiale Flagging (CredentialOnFile) für diverse Anwendungsfälle beschrieben und nachfolgendes Kapitel + Szenario kommt bei Ihnen zum tragen: [3DS 2.0 Händler-Anwendungsfälle & Testen von 3D-Secure 2.0](#)

Kapitel: 2. Anwendungsfälle für Transaktions-Kennzeichnung

Szenario 03 – Kreditkarte Wiederkehrende Zahlung / Anzahlung / Schlusszahlung

***wiederkehrende Batch-Einreichung

Da aus dem initialen Shop Request die schemeReferenceID generiert und zurückgemeldet wurde, muss dieser eine initiale Wert für alle Batch-Nachreichungen verwendet werden + Angabe RTF=M (M=Merchant Initiated Transaction): [Kreditkarten](#)

Kapitel: Batch-Nutzung der Schnittstelle

betroffenes Batch Format / Action:

CC,Sale,<Amount>,<Currency>,<TransID>,<RefNr>,<CCBrand>,<CCNr|PCNr>,<CCExpiry>,<OrderDesc>,<textfield1>,<textfield2>,<RTF>,<cardholder>,<transactionID>,<schemeReferenceID>
CC,Authorize,<Amount>,<Currency>,<TransID>,<RefNr>,<CCBrand>,<CCNr|PCNr>,<CCExpiry>,<OrderDesc>,<textfield1>,<textfield2>,<RTF>,<cardholder>,<transactionID>,<schemeReferenceID>

Im Batch Request gibt es einmal das Feld "transactionID" + zusätzlich "schemeReferenceID" und in einem der beiden Felden können Sie die initiale schemeReferenceID für wiederkehrende Zahlungen anliefern. Wir empfehlen das Feld=schemeReferenceID zu verwenden.

Beispiele Batchversion 1.2 Übergabe der ID im Feld schemeReferenceID:

CC,Sale,<Amount>,<Currency>,<TransID>,<RefNr>,<CCBrand>,<CCNr|PCNr>,<CCExpiry>,<OrderDesc>,<textfield1>,<textfield2>,<RTF>,<cardholder>,<transactionID>,<1234567890>

Die Übergabe des Scheme Identifiers kann im Batchformat im Feld TransactionID / schemeReferenceID erfolgen, d.h. auf die initiale 3DS-2.X Transaktion erhalten Sie die schemeReferenceID / transactionID zurück und dieser Wert muss dann in das Batchformat mit einfließen, damit alle wiederkehrenden Payments korrekt gekennzeichnet sind. Das Feld cardHolder muss vorhanden sein, kann allerdings auch leer übergeben werden. Durch unsere Empfehlung, den Scheme Identifier auch im Feld schemeReferenceID zu übergeben, bleibt das Feld transactionID somit auch leer.

Beispiel:

CC,Sale,999,EUR,987456321,123456789,<VISA>,<0123456789123456>,<052029>,<mein Warenkorb>,,,M,,,<1234567890>

***PKN Mandat - SchemeReferenceID

Sie können versuchen, Payments ohne schemeReferenceID einzureichen, allerdings können wir Ablehnungen dahingehend nicht ausschließen. Sollte es zu einer Ablehnung kommen, empfehlen wir eine separate Transaktion als Card Check (AccVerify=YES) Transaktion über unsere ...PaySSL.aspx auszuführen, d.h. der Kunde erhält einen Payment Link, über welchen dieser eine initiale 3DS-2.X Transaktion prozessiert und darüber wird dann die schemeReferenceID gemeldet und kann in den Prozess der wiederkehrenden Zahlungen mit einfließen.

Szenario 10 – Erweitertes Transaktions-Management (ETM)

- Bei Nutzung des Computop ETMs kümmert sich das Paygate für Sie um die richtige Markierung der Transaktionen.

Test von 3-D Secure 2.x

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 im Computop Paygate eine Testsimulation durchführen. Das ermöglicht Ihnen die Durchführung von 3-D Secure Authentisierungen mit verschiedenen Rückgabewerten.

Gehen Sie für Tests bitte folgendermaßen vor:

- Aktivieren Sie 3-D Secure 2.0 für Ihre Computop MerchantID. Wenn Sie unsicher sind, ob das bereits aktiviert ist, wenden Sie sich bitte an [Computop Helpdesk](#)
- In der verschlüsselten Datenanfrage verwenden Sie den Standardparameter OrderDesc mit dem Wert "Test:0000". Das gibt Ihnen eine entsprechend erfolgreiche Autorisierung nach einer erfolgreichen Authentisierung.
- Führen Sie die 3-D Secure Authentisierung durch
- Verwenden Sie bitte **NUR** die verfügbaren [Testkarten](#) (Ablaufdatum immer in der Zukunft + CVV/CVC kann jeden Wert enthalten)
- Je nach gewünschtem Szenario (z.B. Browser 3-D Secure 2.0 Challenge, reibungslose Browser-Authentisierung usw.) verwenden Sie bitte die entsprechenden [Einmal-Passwörter](#)