Strong Customer Authentication (SCA) for MobilePay Online - version 1.1

This document describes in technical terms how MobilePay intends to meet the SCA requirement that is mandated by the PSD2 directive from January 2021.

Supporting our SCA solutions is mandatory for you as integrators when entering 2021, and therefore we urge you to review the specifications and plan ahead accordingly to have this implemented in ample time before end of year.

SCA for Visa cards

MobilePay will become a Visa Token Requestor and use the Visa Token Service in order to initiate token based transactions for MobilePay Online. This means that MobilePay will get the TAVV cryptogram from the VTS and pass both cryptogram and a token on to the PSP to be used for initiating the card transaction with the appropriate card acquirer. Compliance regarding Delegated Authentication is handled with the card issuers in order to ensure a seamless flow without soft declines/3D Secure step-up.

Technical specifications can be found in Appendix 1 – Visa Token Service

SCA for Mastercard

MobilePay is working with Mastercard in order to implement a solution called *ID Check Express* that will ensure a seamless flow.

Technical specifications can be found in Appendix 2 – Mastercard: ID Check Express

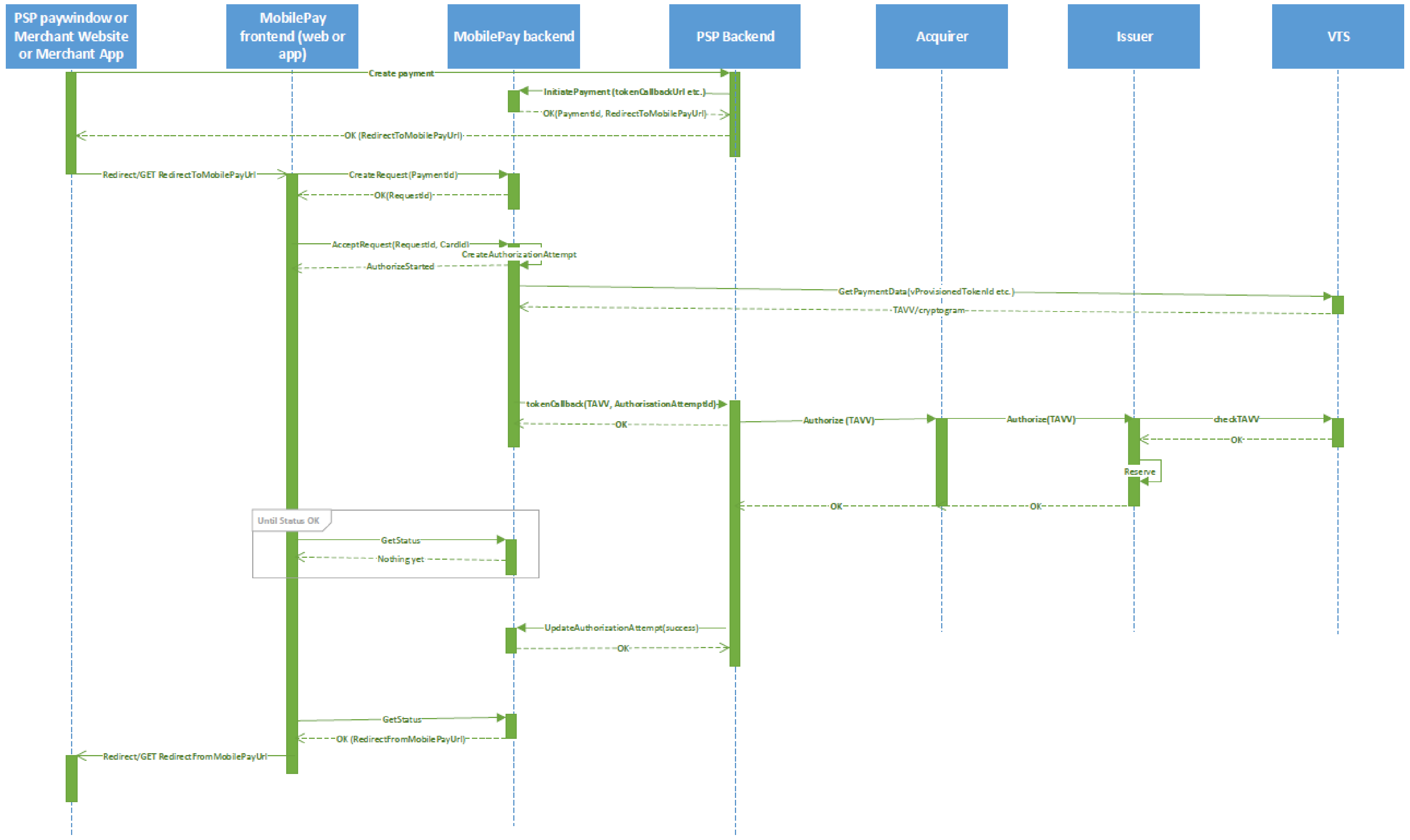
3D Secure fallback

For both Visa and Mastercard a 3D Secure fallback must be implemented in order to ensure a functional user journey. In case a step-up is required by a card issuer, a 3D Secure challenge is enforced. This challenge is presented inside a webview in the MobilePay app as described in the sequence diagram in Appendix 3. Please note that the PSP will host the threeDSecureUrl provided to MobilePay, and ideally do an unseen redirect to the ACS from it. Also, don´t forget to redirect to MobilePay´s 3DSDone page in the webview when the 3DS challenge is completed.

Technical specifications can be found in Appendix 3 – 3D Secure fallback

Appendix 1 – Visa Token Service

Sequence diagram


PSP paywindow or Merchant Website or Merchant App
MobilePay frontend (web or app)
MobilePay backend
PSP Backend
Acquirer
Issuer
VTS
Create payment
InitiatePayment (tokenCallbackUrl etc.)
OK(PaymentId, RedirectToMobilePayUrl)
OK (RedirectToMobilePayUrl)
Redirect/GET RedirectToMobilePayUrl
CreateRequest(PaymentId)
OK(RequestId)
AcceptRequest(RequestId, CardId)
CreateAuthorizationAttempt
AuthorizeStarted
GetPaymentData(vProvisionedTokenId etc.)
TAVV/cryptogram
tokenCallback(TAVV, AuthorisationAttemptId)
OK
Authorize (TAVV)
Authorize(TAVV)
checkTAVV
OK
Reserve
OK
OK
Until Status OK
GetStatus
Nothing yet
UpdateAuthorizationAttempt(success)
OK
GetStatus
OK (RedirectFromMobilePayUrl)
Redirect/GET RedirectFromMobilePayUrl

Initiate payment V2

A version2 of initiatePayment is needed. Instead of simply a list of accepted cardTypes, we will have a list of cardType objects. What was previously the cardType is now the name in the cardType object. Instead of the cardDataCallbackUrl on the payment is now an encryptedPanCallbackUrl on the cardType object. Here is also a tokenCallbackUrl. Both are required for all three Visa card types. MobilePay will prefer to use the tokenCallbackUrl, but if that does not work, we will use the PAN url. This is how the entire body will look:

```
{
  "merchantId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "merchantName": "string",
  "merchantLogoUrl": "string",
  "merchantOrderId": "string",
  "pspReferenceId": "string",
  "amount": 0,
  "currencyCode": "string",
  "allowedCardTypes": [
    {
      "name": "string",
      "encryptedPanCallbackUrl": "string",
      "tokenCallbackUrl": "string"
    }
  ],
  "publicKeyId": 0,
  "redirectFromMobilePayUrl": "string",
  "failedPaymentCallbackUrl": "string",
  "isCheckout": true,
  "addressCallbackUrl": "string",
  "deliveryAddressAllowed": true,
  "deliveryAddressDisallowedReasonCode": 0,
  "deliveryLimitedTo": [
    "string"
  ],
  "validUntil": "2020-09-08T09:21:21.489Z",
  "autoCapture": false,
  "customerLanguageCode": "string"
}
```

paymentToken callback

Whenever possible, we will instead of returning an encrypted PAN to encryptedPANCallbackUrl, make a callback to tokenCallbackUrl like this:

```
{
 "paymentId":"a84781b3-af34-42ae-b296-260cfb6859fe",
 "authorizationAttemptId":"ba12c5d5-8fd1-49cc-bc3f-2cb2ecb888c7",
 "cardType":"VISA-CREDIT",
 "tokenMethod":"VTS",
 "tokenData":{
 "vPaymentDataID":"string",
 "cryptogramInfo":[
   {
    "cryptogram":"string",
    "eci": "string",
    "cryptogramExpirationDate": "string"
   }
 ],
 "paymentInstrument":{
  "last4": "string",
  "paymentType":{
    "cardBrand":"string"
  },
  "paymentAccountReference":"string"
 },
 "tokenInfo":{
  "encTokenInfo":"string",
  "decTokenInfo":"string",
  "last4": "string",
  "expirationDate": {
    "month":"string",
    "year":"string"
  }
 },
 "encryptionMetaData":"string"
 }
}
```

PATCH authorization attempt

As always an authorizationAttempt is created by MobilePay. It must be PATCHed by the PSP with succeded=true/false. No changes to that part of the flow.

Appendix 2 – Mastercard: ID Check Express

Our current card data callback looks like this (you must still support this going forward)

```
{
  'EncryptedCardData': 'fsfnsdjkfbgdft34895u7345',
  'PaymentId': 'a84781b3-af34-42ae-b296-260cfb6859fe',
  'AuthorizationAttemptId': 'ba12c5d5-8fd1-49cc-bc3f-2cb2ecb888c7',
  'PublicKeyId': 263012
}
```

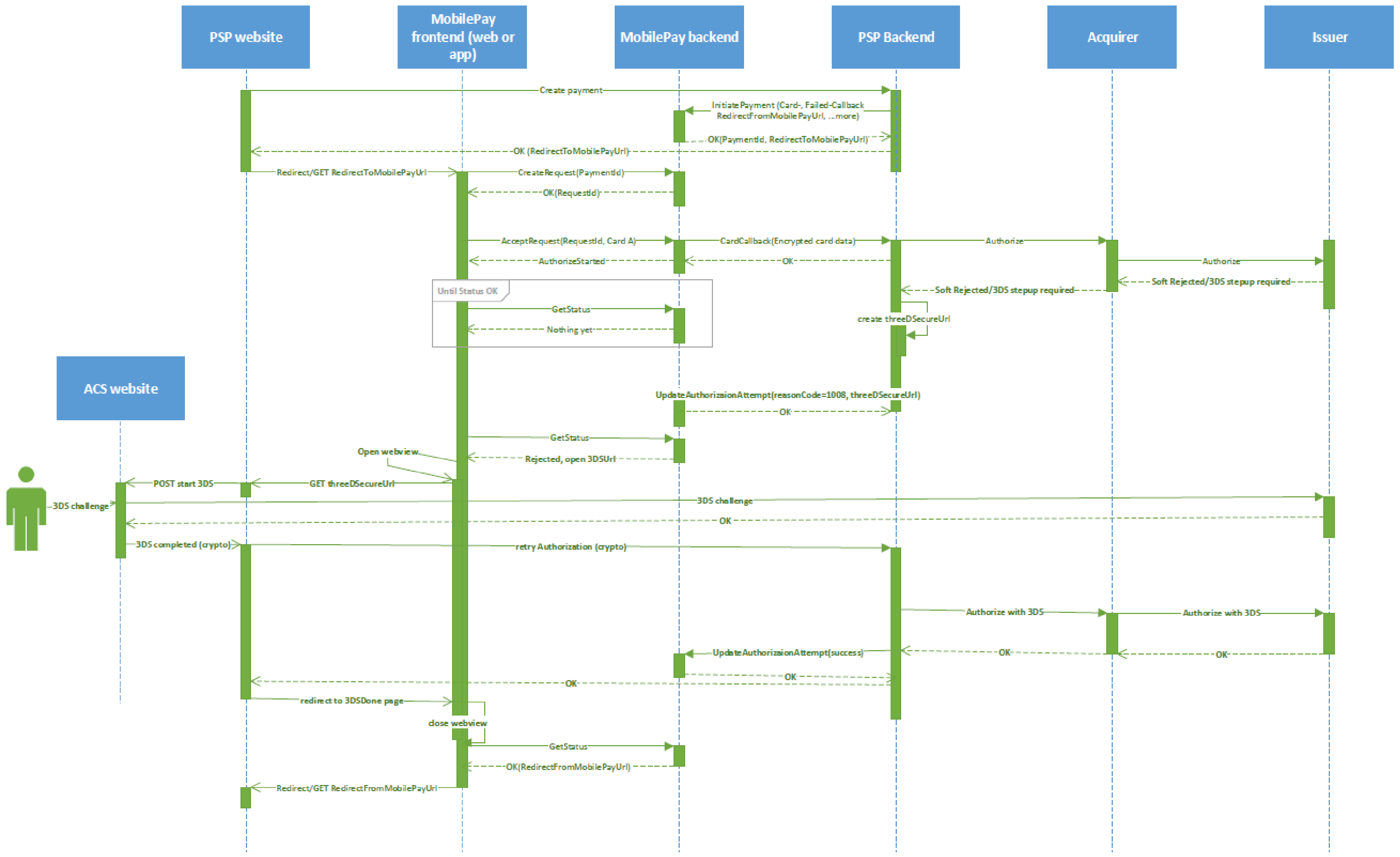
When a successful ID Check Express has been performed it will look like this:

```
{
  "encryptedCardData": "fsfnsdjkfbgdft34895u7345",
  "paymentId": "a84781b3-af34-42ae-b296-260cfb6859fe",
  "authorizationAttemptId": "ba12c5d5-8fd1-49cc-bc3f-2cb2ecb888c7",
  "publicKeyId": 263012,
  "scaMethod": "Mastercard 3DS SPA2 AAV",
  "scaData": {
    "aav": "abc123",
    "dsTransactionId": "abc123"
  }
}
```

Appendix 3 – 3D Secure fallback / soft reject

To handle cases where the card issuer ends up returning a soft reject / 3DS step-up (despite our other SCA solutions), the PSP must support our 3DS fallback solution.

Sequence diagram


PSP website
MobilePay frontend (web or app)
MobilePay backend
PSP Backend
Acquirer
Issuer
Create payment
Initiate Payment (Card-, Failed-Callback RedirectFromMobilePayUrl, ...more)
OK(PaymentId, RedirectToMobilePayUrl)
OK (RedirectToMobilePayUrl)
Redirect/GET RedirectToMobilePayUrl
CreateRequest(PaymentId)
OK(RequestId)
AcceptRequest(RequestId, Card A)
CardCallback(Encrypted card data)
Authorize
Authorize
AuthorizeStarted
OK
Soft Rejected/3DS stepup required
Soft Rejected/3DS stepup required
Until Status OK
GetStatus
Nothing yet
create threeDSecureUrl
ACS website
UpdateAuthorizaionAttempt(reasonCode=1008, threeDSecureUrl)
OK
GetStatus
Open webview
Rejected, open 3DSUrl
POST start 3DS
GET threeDSecureUrl
3DS challenge
3DS challenge
OK
3DS completed (crypto)
retry Authorization (crypto)
Authorize with 3DS
Authorize with 3DS
UpdateAuthorizaionAttempt(success)
OK
OK
OK
OK
redirect to 3DSDone page
close webview
GetStatus
OK(RedirectFromMobilePayUrl)
Redirect/GET RedirectFromMobilePayUrl

Appendix 4 – Document change log

V1.0 -> V1.1

Appendix 1 - The specification for “Initiate payment” has changed.
Appendix 2 - Added “dsTransactionId” under “scaData” in the new card data callback.