Saved card payments after PSD2 and Strong Customer Authentication for webshops

Introduction

On September 14th 2019, Strong Customer Authentication (SCA) requirement related to PSD2 mandates will be fully enforced. In practice, this means that all card transactions must be authenticated, and any authorization done without SCA will be rejected by the card issuer. Currently, several merchants are using the saved card feature provided by Verifone to commit one-click transactions using *process-payment* server interface call. With this method, it is not possible to do strong authentication for the customer, and the authorization is extremely likely to be rejected.

Notably, out-of-scope are Merchant Initiated Transactions (MIT), where the buyer is not present for payment. In general, this applies to things such as recurring payments (eg. Netflix or Spotify subscriptions that are automatically billed), late/no-show fees in hotels where the buyer is not available/present for the transaction. MIT transactions can still be done without needing to go through SCA; but the initial subscription where the card was saved, must have gone through 3D Secure authentication.

For mobile applications (iOS/Android), an SDK is available that will perform 3D Secure v2 authentication with the help of additional server interface call *3ds-lookup*. Specifications for the SDK are provided in a separate document if needed.

For web based solutions, this would mean that *process-payment* can no longer be used for payments without applying for acquirer exemptions or issuer whitelistings. Issue with the exceptions, however, is that not all of them are supported out of the game on September 14th, and card issuers still can override these exemptions, effectively still requiring strong authentication to be done. Same applies for issuer whitelistings, it's unlikely that many, if any, issuers support for the buyer to whitelist webshops on the deadline date.

Card on File payment with authentication for webshops and mobile applications without SDK

For browser based shops, instead of using *process-payment* call as is done today, a payment request should be sent to the Verifone Hosted Payment Page (HPP), similar to how it is done with with the initial card save transaction. Webshop can provide */-t-1-20_saved-payment-method-id* on the HPP request as is done with *process-payment* call today. This way, the buyer does not need to re-enter card details on the payment page, and the payment can proceed directly to 3D Secure if needed.

Additionally, for 3D Secure v2, card schemas require additional data to be provided on the payment. Following information would need to be sent, as if they are missing, HPP may ask the buyer to input them:

Data	API parameter(s)	Notes
Buyer first name	s-f-1-30_buyer-first-name	Already mandatory in the API
Buyer last name	s-f-1-30_buyer-last-name	Already mandatory in the API
Buyer phone number	s-t-1-30_buyer-phone-number	Currently optional in the API
Buyer email address	s-f-1-100_buyer-email-address	Already mandatory
Buyer delivery address	s-t-1-30_delivery-address-line-one, s-t-1-30_delivery-address-line-two, s-t-1-30_delivery-address-line-three, s-t-1-30_delivery-address-city, s-t-1-30_delivery-address-postal-code, i-t-1-3_delivery-address-country-code	Currently optional in the API
Buyer billing address	s-t-1-30_bill-to-address-first-name, s-t-1-30_bill-to-address-last-name, s-t-1-30_bill-to-address-line-one, s-t-1-30_bill-to-address-line-two, s-t-1-30_bill-to-address-line-three, s-t-1-30_bill-to-address-postal-code, s-t-1-30_bill-to-address-city, i-t-1-3_bill-to-address-country-code	New parameters that can be sent on the request. If billing address is not set, delivery address will be used.

If these details are not set, buyer may be required to input them in the HPP, so if these details are already gathered by the webshop, they should be relayed to Verifone in the payment request if not done today.

Example

Pre-PSD2 flow:

1. Webshop displays list of saved cards for buyer
2. Buyer picks a saved card (or had one preselected)

3. Buyer clicks on buy
4. Webshop issues a *process-payment* call to Verifone server interface
5. Verifone authorizes the transaction via the acquirer
6. Webshop receives payment OK response, payment is done

Required PSD2 flow:

1. Webshop displays list of saved cards for buyer
2. Buyer picks a saved card (or had one preselected)
3. Buyer clicks on buy
4. Webshop redirects the buyer to the Verifone hosted payment page, and includes the saved-payment-method-id in the payment request along with other buyer and payment details
5. If required by the issuer, buyer is authenticated using 3D Secure
6. Verifone authorizes the transaction via the acquirer with 3D Secure details
7. Webshop receives OK payment as buyer is redirected back to the shop, payment is done
8. Delayed success callback message is sent to the shop in case buyer doesn't return from payment page. This will be sent for all successful payments.

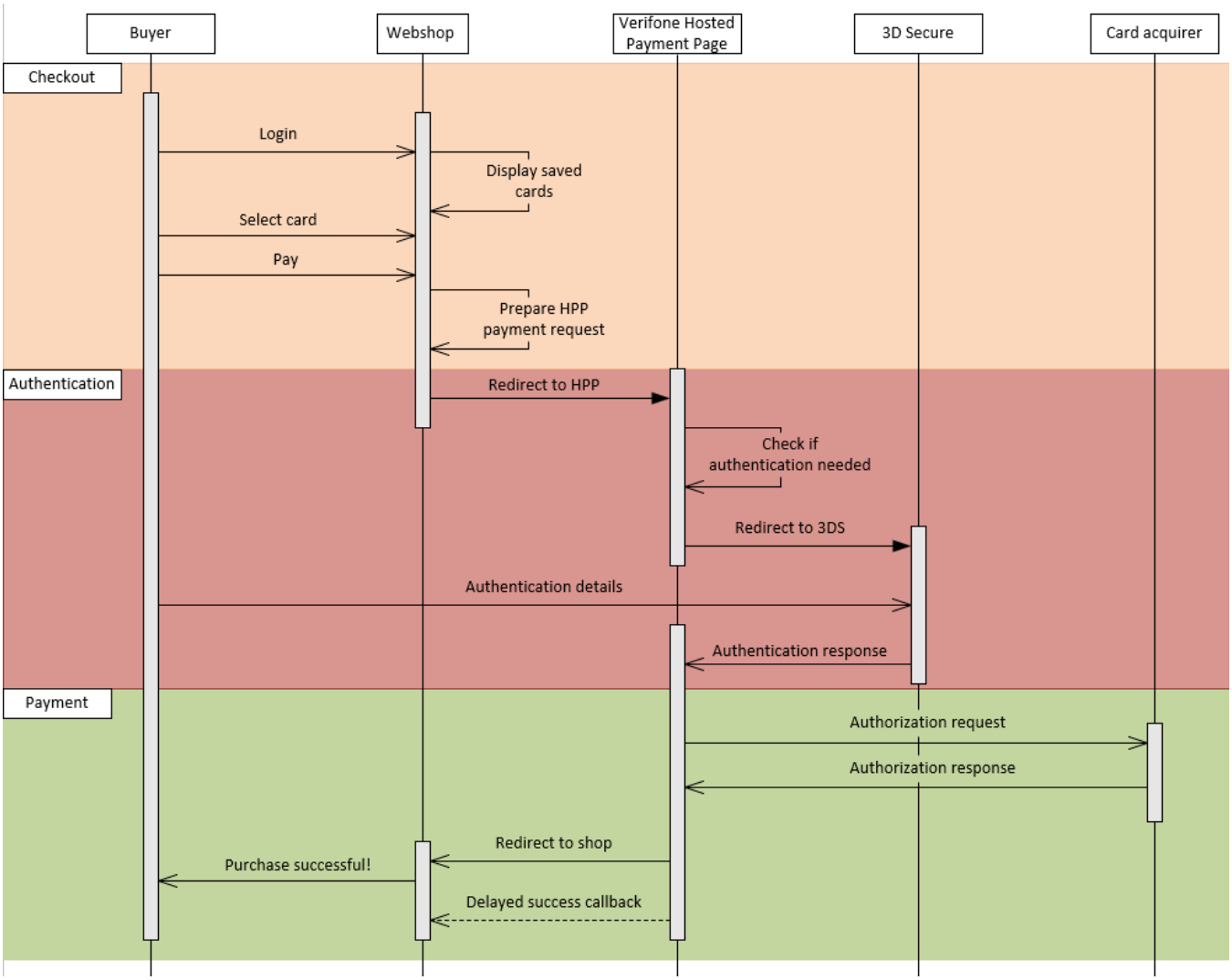


Diagram 1. Transaction flow for saved cards from a browser-based shop

Changes to card saving

While no drastic changes are needed for card saving transaction itself, but as authentication is done when saving the card, for 3DSv2 it is recommended to include the same buyer details as mentioned before for card on file transactions; otherwise the buyer may be required to input the details on the payment page.

Card payments after PSD2 and Strong Customer Authentication for webshops

Introduction

On September 14th 2019, Strong Customer Authentication (SCA) requirement related to PSD2 mandates will be fully enforced. In practice, this means that all card transactions must be authenticated, and any authorization done without SCA will be rejected by the card issuer. Due to this reason, any merchant configuration to skip 3D Secure will be reverted, and authentication must be attempted for all transactions.

Card schemas are also bringing out new version of 3D Secure; 3D Secure v2. This interface allows merchant to provide more detailed information on the buyer and the payment in general to the card issuer, and based on this additional data, issuer may allow buyer to proceed with the payment without being forced to do full 3D Secure authentication, and approving the payment as is. For this reason, merchant should collect and relay as much information on the buyer to Verifone in the payment request as possible.

New parameters for customer authentication in the payment request

Following parameters must be present in the payment request in the future. Some of these are already collected and used in the payment interface, but if the shop currently is collecting for example the phone number and delivery address, but not sending it to Verifone in the payment request, Verifone payment page may request the buyer to input the details again. Thus, it is recommended that the shop sends as much buyer information as possible.

Data	API parameter(s)	Notes
Buyer first name	s-f-1-30_buyer-first-name	Already mandatory in the API
Buyer last name	s-f-1-30_buyer-last-name	Already mandatory in the API
Buyer phone number	s-t-1-30_buyer-phone-number	Currently optional in the API
Buyer email address	s-f-1-100_buyer-email-address	Already mandatory
Buyer delivery address	s-t-1-30_delivery-address-line-one, s-t-1-30_delivery-address-line-two, s-t-1-30_delivery-address-line-three, s-t-1-30_delivery-address-city, s-t-1-30_delivery-address-postal-code, i-t-1-3_delivery-address-country-code	Currently optional in the API
Buyer billing address	s-t-1-30_bill-to-address-first-name, s-t-1-30_bill-to-address-last-name, s-t-1-30_bill-to-address-line-one, s-t-1-30_bill-to-address-line-two, s-t-1-30_bill-to-address-line-three, s-t-1-30_bill-to-address-postal-code, s-t-1-30_bill-to-address-city, i-t-1-3_bill-to-address-country-code	New parameters that can be sent on the request. If billing address is not set, delivery address will be used.

3DSv2 via Cardinal SDK

Table of Contents

Document change history

Date	Version	Changed by	Changes
18.06.2019	1.0	Toni Kankkunen	Initial release

Notice

As of writing 18.6.2019, described functionalities cannot be tested yet by the merchant. Current expectation would be to offer the new calls in Verifone's preview-test environment in August, and features to be in production by September 14th. Exact dates will be provided later.

Initial setup

Developer will need a BinTray username and API key to access the SDK in their IDE which will be supplied by Verifone.

Merchant backend system will need an additional API key to be used with Cardinal Commerce itself; this API key will be used for signing JWT towards the SDK. This will be supplied by Verifone.

In-depth documentation on the Cardinal JWTs can be found from https://cardinaldocs.atlassian.net/wiki/spaces/CC/pages/327884/JWT+Documentation

Documentation on the SDK itself can be found at https://cardinaldocs.atlassian.net/wiki/spaces/CMSDK/overview. Note that SDK documentation doesn't match the solution for Verifone eCommerce 100%, notably Centinel related cmpi_lookup call is done instead towards Verifone's eCommerce interface using the *3ds-lookup call,* which will handle the full card PAN required for 3DS.

This document assumes familiarity with Verifone's eCommerce API. Refer to Hosted Payment Pages Interface Reference Guide and Server to Server Interface Reference Guide for details.

Flow overview

1. SDK setup. Application will need to call Cardinal.Configure to prepare the SDK. Customization can be added at this point, for further information, refer to the Cardinal SDK documentation
2. Card token and BIN. Application will need BIN portion of the saved card (first 6 digits). This can be fetched using the eCommerce server interface call list-saved-payment-methods. This will return both masked PAN (with BIN portion in tact) as well as saved payment method ID. These can be saved at the merchant back end, and is recommended.
3. JWT. Application needs to request the merchant backed for a JWT for each transaction. Application will need to request, or merchant backend needs to generate an unique ID for each transaction, which will be included in the JWT with ReferenceId claim. Application will need to provide 3ds-lookup call this value in order to link up Cardinal SDK and Verifone eCommerce platform.
4. Initialization. Application needs to call Cardinal.Init with the JWT and card BIN. Refer to *Cardinal.init with accountNumber* in Cardinal documentation.
5. Lookup. Application or the backend needs to send server interface call *3ds-lookup* to Verifone. This step differs from the Cardinal documentation, where this call is set to Centinel instead. In response, there can be 6 different outcomes:
    a. lookup-result: *authenticated*. CAVV, ECI, XID and 3DS Transaction ID (not to be mixed up with the normal Verifone transaction-id) are received. Authentication is complete. Proceed to step 6.
    b. lookup-result: *challenged*. *3ds-lookup* replies with pares-status *C.* Included will be s-t-1-2048_acs-url, s-t-1-2048_payload and s-t-1-20_sdk-transaction-id.
        i. Application needs to call SDK with cardinal.cca_continue using ACSUrl, Payload, TransactionId received from the lookup.
        ii. SDK will then take over, and perform Strong Customer Authentication using 3DSv2.
        iii. Once authentication is successfully done, the SDK will call onValidated() with response JWT. This JWT will contain CAVV, ECI and XID.

iv. Proceed to step 6.

c. lookup-result: *3ds-v1-required*. Card issuer has not implemented 3DSv2. The Cardinal SDK does not support 3DSv1 payments, to process the payment, application will need to make a Hosted Payment Page payment request to Verifone to use 3DSv1. This request can contain the saved-payment-method-id to simplify the payment process. Note, that the buyer will be required to input their CVV.

d. lookup-result: *not-enrolled*. Enrollment status is replied as something else than *Y*. The card is either not enrolled, DS or ACS servers were unavailable, or a merchant bypass was applied. Proceed to step 6.

e. lookup-result: *rejected*. pares-status is N or R in the response. Authentication is rejected. Payment may not be completed.

f. lookup-result: *failed*. There was an issue in performing the lookup. See *s-f-1-30_error-message* for error message.

6. Call *process-payment* on Verifone eCommerce. If CAVV, XID, ECI and Transaction ID are available, they must be passed in the process-payment call.

Server interface calls

3ds-lookup request

Request headers

Following address details are mandatory for Cardinal:

Buyer name, billing address (bill-to-address-*), email address, phone-number. Delivery address can be included, but is not required.

Name	Format	Opt	Example value	Title	Ver	Description
s-f-1-50_sdk-referenceid	String with length of 1-50 characters.	No	ABCDEFGHIJKLM1234	SDK ReferenceID	5	ReferenceID used in the SDK JWT with cardinal.init call. Used to connect the Cardinal init call to Verifone lookup call.
s-f-1-36_order-number	String with length of 1-36 characters.	No	123	Order Number	5	Textual order number assigned by shop system. Valid characters are a-z, A-Z, 0-9 and minus sign.
l-t-1-20_saved-payment-method-id	64 bit signed integer value formatted as a string with 1-20 numeric characters	Yes	1234567890	Payment method id	5	ID of the saved payment method.
s-f-1-30_payment-method-code	String with length of 1-30 characters.	No	visa	Payment Method	5	String key identifying the payment method used.
l-f-1-20_order-gross-amount	64 bit signed integer value formatted as a string with 1-20 numeric characters.	No	100	Gross Amount	5	Total amount including taxes and discount with two decimal precision. Example value corresponds to 1 EUR.
i-f-1-3_order-currency-code	String with length of 1-3 numeric characters.	No	978	Currency Code	5	Numeric ISO 4217 currency code.
s-f-1-30_buyer-first-name	String with length of 1-30 characters.	No	John	First Name	5	First name of the buyer.
s-f-1-30_buyer-last-name	String with length of 1-30 characters.	No	Smith	Last Name	5	Last name of the buyer.
s-t-1-30_buyer-phone-number	String with length of 0,1-30 characters.	No	+358 40 234234	Phone Number	5	Phone number of the buyer.
s-f-1-100_buyer-email-address	String with length of 1-100 characters.	No	john.smith @gmail.com	Email Address	5	Email address of the buyer.
s-t-1-255_buyer-external-id	String with length of 1-255 characters.	No	213123123	Buyer External Identifier	5	Unique identifier of the buyer assigned by web shop.
s-t-1-30_bill-to-address-first-name	String with length of 1-30 characters.	No	Card	First name of the payee	5	Cardholders first name
s-t-1-30_bill-to-address-last-name	String with length of 1-30 characters.	No	Holder	Last name of the payee	5	Cardholders last name
s-t-1-30_bill-to-address-line-one	String with length of 1-30 characters.	No	Billingstreet 15	Billing address line one	5	Cardholder billing address, line one
s-t-1-30_bill-to-address-line-two	String with length of 1-30 characters.	Yes	Apartment 5	Billing address line two	5	Cardholder billing address, line two
s-t-1-30_bill-to-address-line-three	String with length of 1-30 characters.	Yes	Room 6	Billing address line three	5	Cardholder billing address, line three

s-t-1-30_bill-to-address-postal-code	String with length of 1-30 characters.	No	Helsinki	Billing address postal code	5	Cardholder billing address, postal code
s-t-1-30_bill-to-address-city	String with length of 1-30 characters.	No	00100	Billing address city	5	Cardholder billing address, city
i-t-1-3_bill-to-address-country-code	String with length of 0 or 1-3 characters.	No	246	Billing address country code	5	Cardholder billing address, country in numeric ISO 3166 country code of the delivery address.
s-t-1-30_delivery-address-line-one	String with length of 0 or 1-30 characters.	Yes	Street 31	Delivery Address Line #1	5	Line one of the delivery address.
s-t-1-30_delivery-address-line-two	String with length of 0 or 1-30 characters.	Yes	Apartment 2	Delivery Address Line #2	5	Line two of the delivery address.
s-t-1-30_delivery-address-line-three	String with length of 0 or 1-30 characters.	Yes	Room 3	Delivery Address Line #3	5	Line three of the delivery address.
s-t-1-30_delivery-address-city	String with length of 0 or 1-30 characters.	Yes	Helsinki	Delivery Address City	5	City of the delivery address.
s-t-1-30_delivery-address-postal-code	String with length of 0 or 1-30 characters.	Yes	00270	Delivery Address Postal Code	5	Postal code of the delivery address.
i-t-1-3_delivery-address-country-code	String with length of 0 or 1-3 characters.	Yes	246	Delivery Address Country Code	5	Numeric ISO 3166 country code of the delivery address.

Response headers

Name	Format	Opt	Example value	Title	Ver	Description
s-t-1-30_lookup-result	String with length of 1-30 characters.	No	challenged	Lookup Result	5	Result of the lookup. Available values: authenticated - authentication was success. s-t-1-40_cavv, s-t-1-2_eci-flag, s-t-1-36_ds-transaction-id and s-t-1-40_xid will be present in the response. challenged - step up to strong customer authentication was required by the issuer. s-t-1-2048_acs-url and s-t-1-2048_payload will be present in the response. 3ds-v1-required - Issuer is not enrolled to 3DSv2. Authentication must be done using 3DS v1 via the Verifone HPP interface. not-enrolled - Card is not enrolled to 3DS rejected - rejected by the issuer. Payment may not continue. failed - Lookup could not be performed for some reason. See *s-f-1-30_error-message* for details.
s-t-1-36_ds-transaction-id	String with length of 1-36 characters	Yes	ABC12345678	DS Transaction ID	5	Unique transaction identifier assigned by the Directory Server (DS) to identify a single transaction. *Present only with lookup-result **authenticated**. Required for process-payment if available.*
s-t-1-2_eci-flag	String with length of 1-2 characters	Yes	02	ECI flag	5	Electronic Commerce Indicator (ECI) from 3DS. **Possible Values:** 02 or 05 - Fully Authenticated Transaction 01 or 06 - Attempted Authentication Transaction 00 or 07 - Non 3-D Secure Transaction Mastercard - 02, 01, 00 VISA - 05, 06, 07 AMEX - 05, 06, 07 JCB - 05, 06, 07 DINERS CLUB - 05, 06, 07 *Present only with lookup-result **authenticated**. Required for process-payment if available.*
s-t-1-40_cavv	String with length of 1-40 characters	Yes	AAAEFEBABABABABABABABABA	CAVV	5	Cardholder Authentication Verification Value (CAVV) / Authentication Verification Value / (AVV) / Universal Cardholder Authentication Field (UCAF) received from 3DS. *Present only with lookup-result **authenticated**. Required for process-payment if available.*

s-t-1-40_xid	String with length of 1-40 characters	Yes	AAAAAAAAAAAAAAAAAAAAA	XID	5	Transaction identifier resulting from authentication processing. *Present only with lookup-result **authenticated**. Required for process-payment if available.*
s-t-1-2048_payload	String with length of 1-2048 characters	Yes	AAAAAAAAAAAAAAAAAAAAA	Payload	5	Payload generated by Centinel for the SDK. *Present only with lookup-result **authenticated** and **challenged**. Must be passed onto the SDK.*
s-t-1-20_sdk-transaction-id	String with length of 1-20 characters.	Yes	ABCDEFGHIJKLM1234	SDK Transaction ID	5	Cardinal server transaction identifier. Used to link up the lookup and Authenticate messages with the SDK flow. *Present only with lookup-result **challenged**. Must be passed onto the SDK.*
s-t-1-2048_acs-url	String with length of 1-2048 characters	Yes	https://acs.3dsecure.com/authenticate	ACS URL	5	The fully qualified URL to redirect the Consumer to complete authentication by the SDK. *Present only with lookup-result **challenged**. Must be passed onto the SDK.*
s-f-1-30_error-message	String with length of 1-30 characters	Yes	ds-unavailable	Error message	5	Raised if there was an error performing the lookup.

process-payment

New request headers

Refer to Verifone Ecommerce Server Interface Reference Guide for details on existing parameters.

Name	Format	Opt	Example value	Title	Ver	Description
s-t-1-36_ds-transaction-id	String with length of 1-36 characters	Yes	ABC12345678	DS Transaction ID	5	Unique transaction identifier assigned by the Directory Server (DS) to identify a single transaction. ***Required** as received from 3ds-lookup or SDK if available*
s-t-1-2_eci-flag	String with length of 1-2 characters	Yes	02	ECI flag	5	Electronic Commerce Indicator (ECI) from 3DS. **Possible Values:** 02 or 05 - Fully Authenticated Transaction 01 or 06 - Attempted Authentication Transaction 00 or 07 - Non 3-D Secure Transaction Mastercard - 02, 01, 00 VISA - 05, 06, 07 AMEX - 05, 06, 07 JCB - 05, 06, 07 DINERS CLUB - 05, 06, 07 ***Required** as received from 3ds-lookup or SDK if available*
s-t-1-40_cavv	String with length of 1-40 characters	Yes	AAAEFEBABABABABABABABABABA	CAVV	5	Cardholder Authentication Verification Value (CAVV) / Authentication Verification Value / (AVV) / Universal Cardholder Authentication Field (UCAF) received from 3DS. ***Required** as received from 3ds-lookup or SDK if available*
s-t-1-40_xid	String with length of 1-40 characters	Yes	AAAAAAAAAAAAAAAAAAAAA	XID	5	Transaction identifier resulting from authentication processing received from 3DS. ***Required** as received from 3ds-lookup or SDK if available*

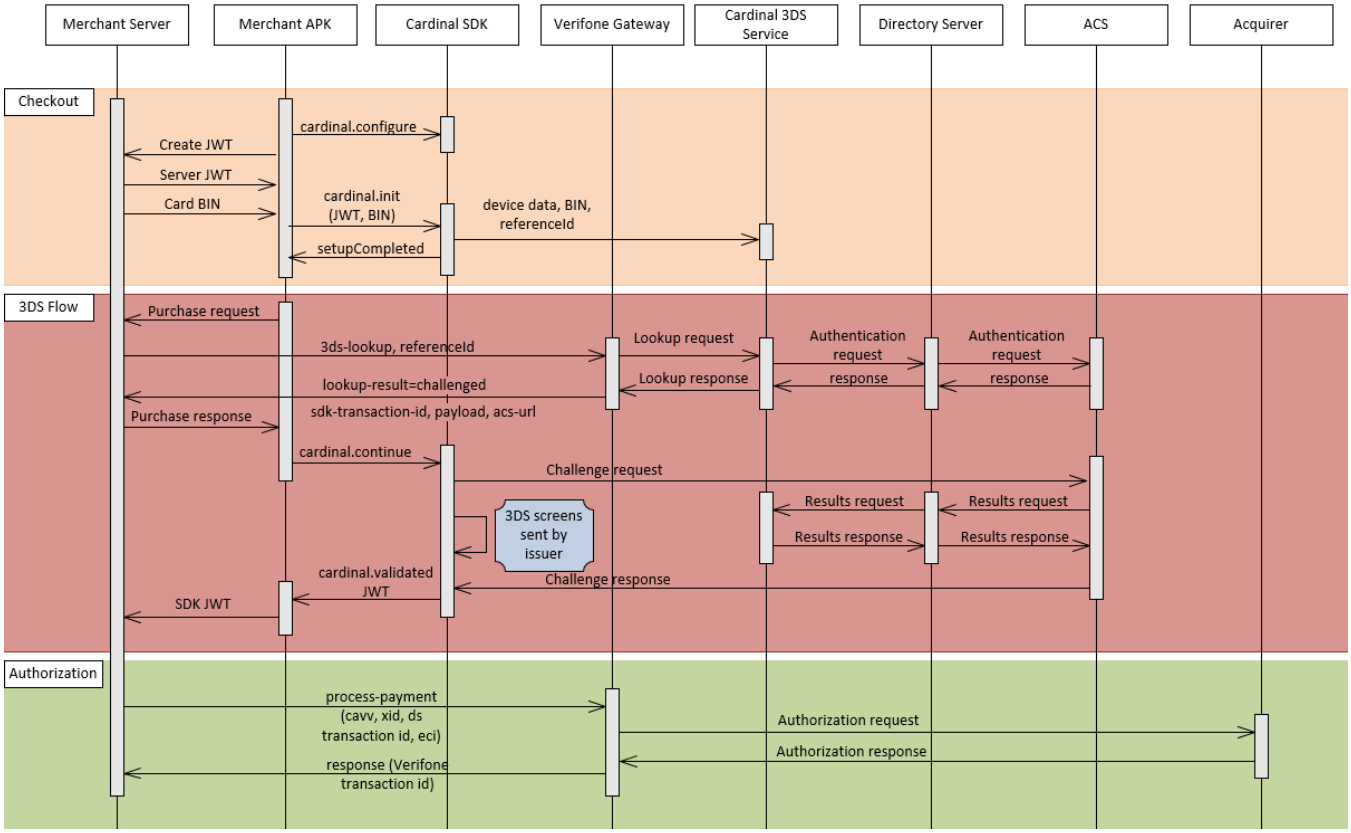
New dynamic feedback parameter for process-payment

Existing s-t-1-1024_dynamic-feedback -parameter in the API can be currently used to request additional information on the transaction. A new field enabling the merchant to receive the authorization response if the authorization has failed has been added; i-t-1-3_failed-auth-response.

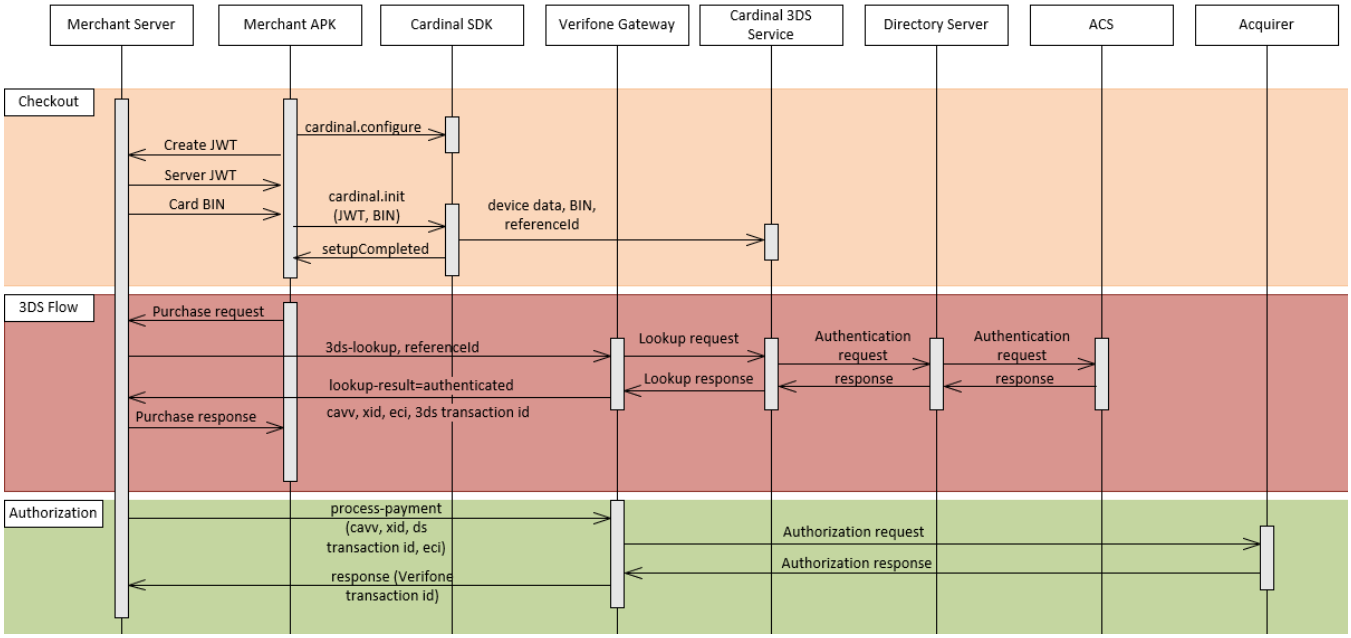
s-t-1-1024_dynamic-feedback	String with length of 1-1024 chars.	Yes	i-t-6-6_card-pan-first6,i-t-4-4_card-pan-last4,i-t-1-3_failed-auth-response	Dynamic feedback	5	List of parameters to be added to response if available. Refer to process-payment call on Server Interface Reference for a full list. Example value would return the listed 3 extra parameters in the response (first 6 digits of card pan, last 4 digits of card pan, and auth response code if the authorization fails)

Failed auth response can be simulated in the Verifone test environment with order gross amounts in the range of 90,00 - 99,99. Gross amount value 9116 would reject the payment with response code 116 (Insufficient funds), and if requested in dynamic feedback parameter, the response code will be returned.

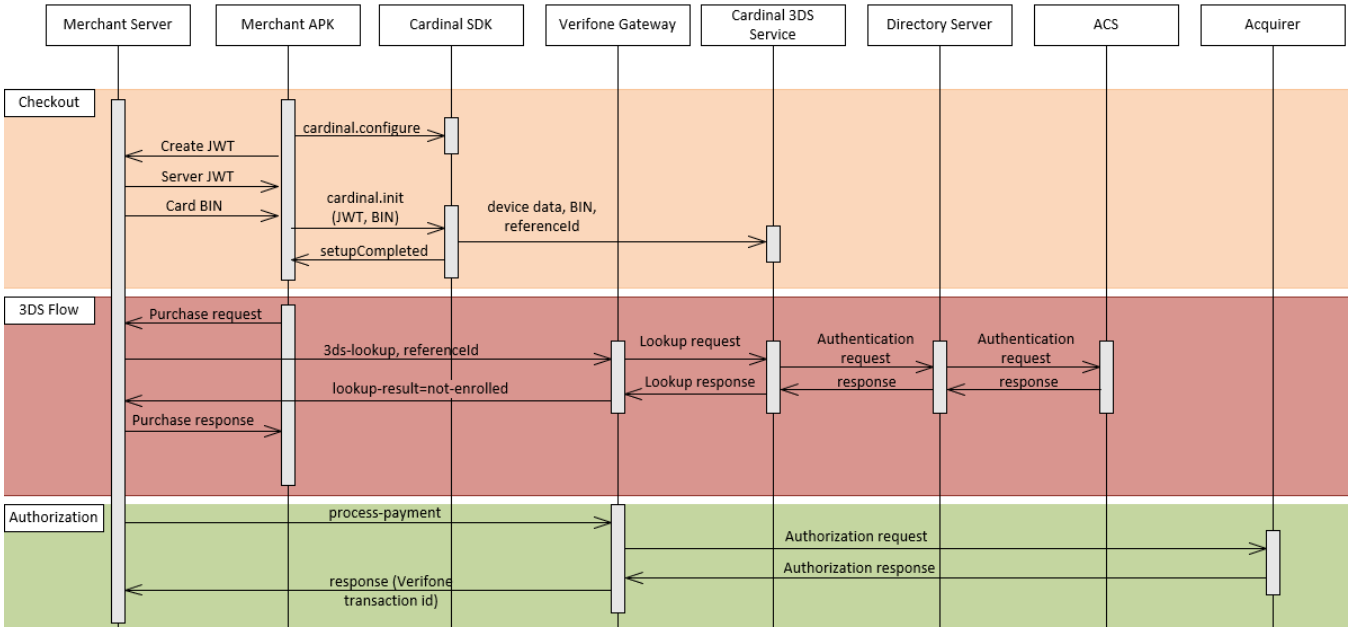
Flow diagrams



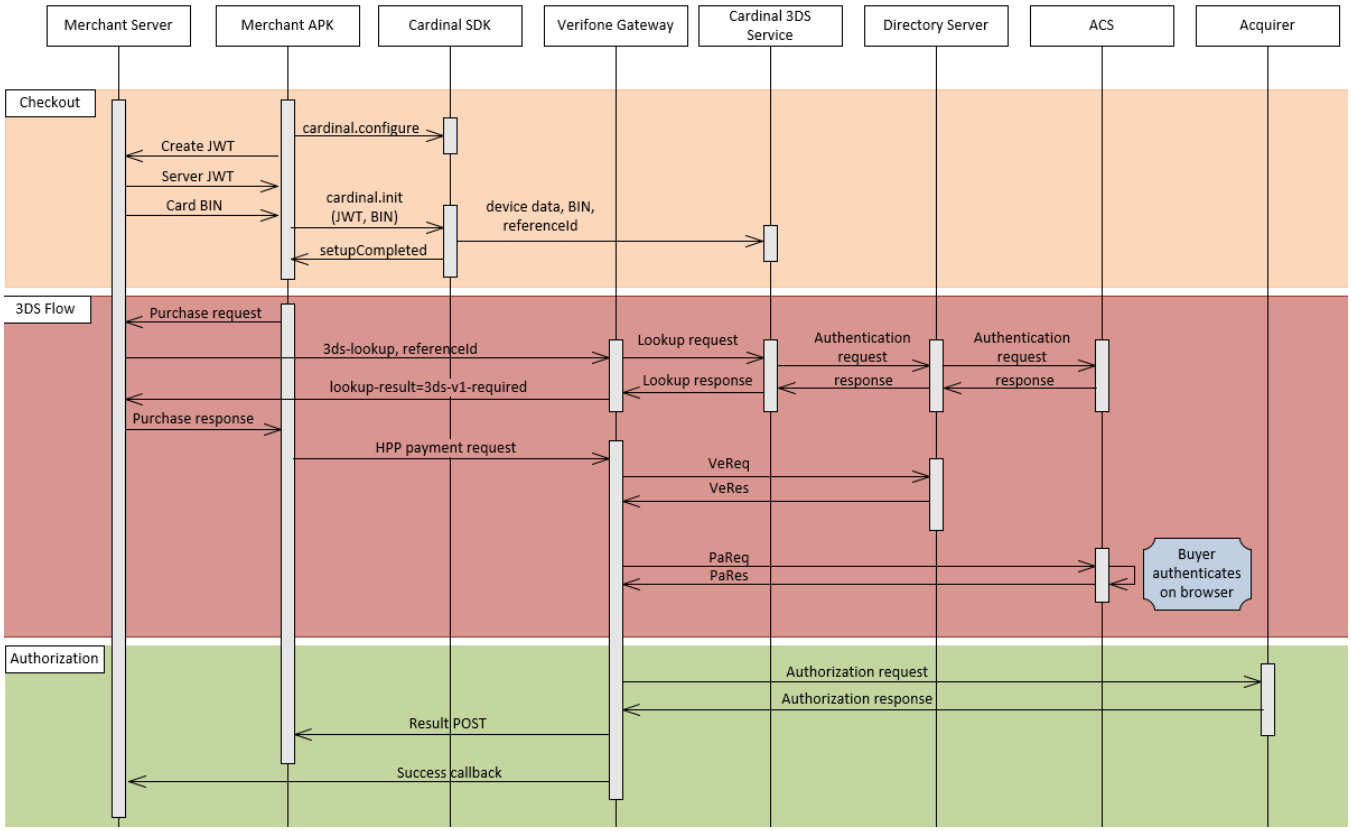
Flow 1. 3DSv2, Challenged by issuer,



Flow 2. 3DSv2; Frictionless flow.



Flow 3. Card not enrolled to 3DS



Flow 4. Issuer not participating in 3DSv2