Technical Integration Guide for PSP

Integration Guide
Version 1.0
As of 17.06.2020

Contents

Document history

Date	Name	Change
17.06.2020	Peter Posse	Original version

Introduction

The BNP Paribas payment platform (also called Payment Platform or Payment Gateway) offers several interfaces for the submission of payment tasks.

This manual describes the programming of the platform and serves to connect PSP system.

The platform will process the data and carries out payment transactions. Therefore the payment gateway will ensure the secure processing of all transactions.

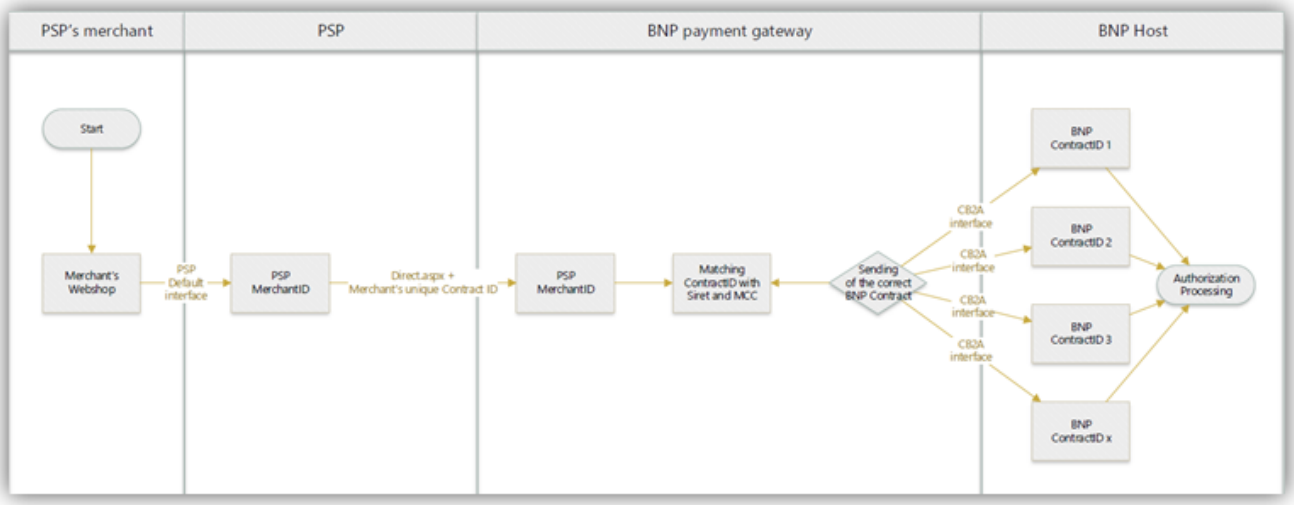


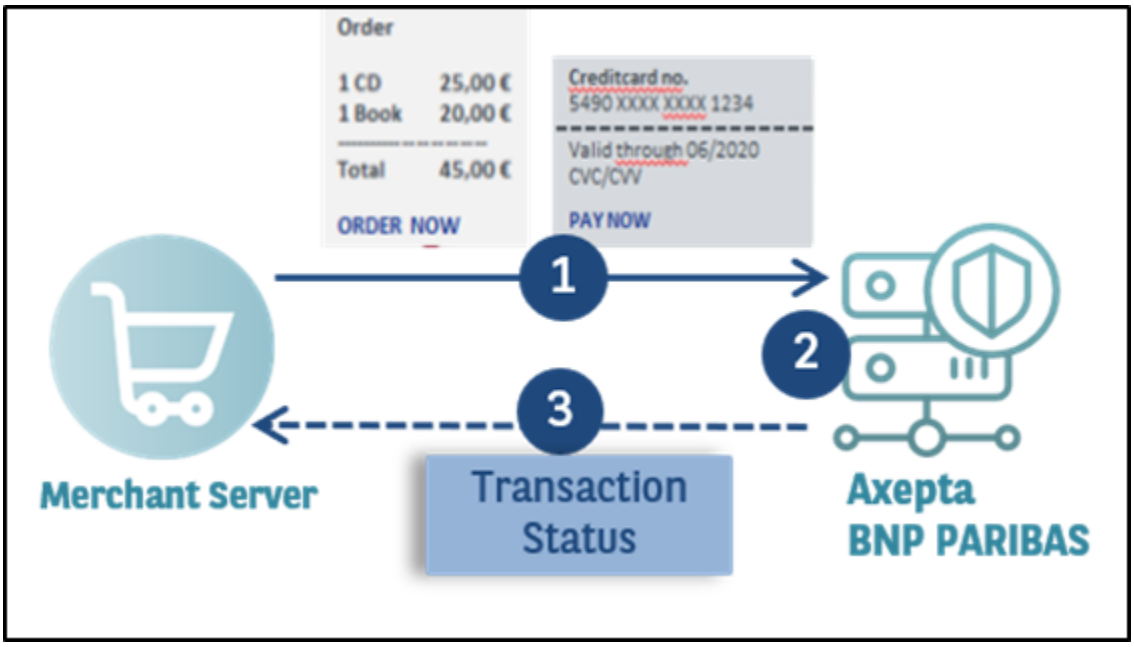
Chart of process flow via Server-to-Server connection

Server-to-Server integration

Server to server (also called machine to machine): your system saves payment details such as card numbers or bank account details and then creates a TLS socket-connection to the Payment platform Server in order to process the payment. Your system controls the communication with Payment platform which automatically process the payments for you.

The PSP is required to pass full PCI DSS certification.

In the case of payments via the Server-to-Server connection, the PSP holds payment details such as card numbers and bank account details. PSP systems create a TLS socket-connection to the Payment platform server in order to carry out a payment transaction.



Notice: When processing payments via a Server-to-Server connection your system must control the communication with the payment platform automatically.

Process of a Server-to-Server payment

The request for a payment starts with the correct composition of the parameters which consist of a key and a value (Name Value Pairs - NVP format) and which are separated by an “equals sign” (=).

```
MerchantID=YourMerchantID
```

All parameters are assembled in a character string and separated by the character &:

```
Amount=100&Currency=EUR&TransID=12345
```

Notice: Since the characters "=" and "&" are used as separating characters, these characters cannot be transmitted as values. All values which you transmit without BlowFish-encryption must be URL-Encoded.

There is only one exemption from this rule: For cards which are registered for Verified/SecureCode/SafeKey/JSecure/ProtectBuy for example the ACSURL is transmitted unencoded.

A correct parameter character string for the payment platform contains three basic parameters: **MerchantID**, **Len** and **Data**. The parameters MerchantID and Len are unencrypted. Only the Data parameter is Blowfish-encrypted:

```
MerchantID=YourMerchantID&Len=67&Data=0A67FE96a65d384350F50FF1
```

The **Data** parameter contains the sensitive payment details such as amount and currency. The encrypted bytes are Hex-encoded and completed to two characters from the left with a zero. Encryption is via Blowfish ECB and is available to you as source-code and components.

The **Len** parameter is very important for encryption because it contains the length of the unencrypted character string in the **Data** parameter. Since the data quantity to be encrypted is increased by a multiple of 8 in the case of the Blowfish encryption, the correct length of the character string must be known for decryption. Otherwise accidental characters emerge at the end of the character string.

The following example show the development of an unencrypted payment request:

```
MerchantID=YourMerchantID&TransID=100000001&Amount=11&Currency=EUR&OrderDesc=My
purchase&CCNr=1111333355557777&CCCVC=123&CCExpiry=200407&CCBrand=VISA
```

Notice: Please note that a value is to be assigned to each parameter. Do not transmit empty parameters, as this can cause the payment to fail.

When this character string is encrypted with Blowfish, it may look like the following:

```
MerchantID=YourMerchantID&Len=140&Data=D622C5FE7414F73539A1852C2CE7AA0BE904A7E2339DCF9363DA6ACDBE5EF98E169FC3092
B1602564DBF2C3C75173A62C484962A247B8A91EA7A544ADCF2A037135421FD0CE092C174A7D1D63517BD45099AC2B682F5E3CD2C942
A6F0E741A833C
```

In order to make payments via a Server-to-Server connection, open a TLS-Socket connection to the payment platform and transfer the generated character string to the following URL:

https://paymentpage.axepta.bnpparibas/direct.aspx

As soon as the TLS socket connection is made, a normal HTTP POST, version 1.1 is carried out. In this case the following fields are specified in the HTTP header:

Field	Value
Host	paymentpage.axepta.bnpparibas
Connection	Close
Content-type	Application/x-www-form-urlencoded
Content-length	Length of character string transferred to the HTTP-Body
Charset	UTF-8

Mandatory information within HTTP-header

The HTTP Body contains the parameter character string. Note that the values must be submitted as URL-encoded parameters. The following listing is an example of a card payment:

```
POST /direct.aspx HTTP/1.1

Host: www.BNPParibas-Payment platform.com

Connection: Close

Content-type: application/x-www-form-urlencoded

Content-Length: 287



MerchantID=YourMerchantID&Len=162&Data=E98D40FFFD622C5FE7414F73539A1852C2CE7C8B09D3E876F52CBECF59EC63E9B8AA0130F
A92F65964E3EEE74DF217E27FA2E194B9968DE9ABAE3B1F44B5485EFE3EF2597C7395BADBAD4340CDFD000DD57129EEFAA0BE904A7E
2339DCF9363DA6ACDBE5EF98E169FC3092B1602564DBF2C3C75173A62C484962A247B8A91EA7A5
```

Notice: Please note that the maximum length of a payment request is limited to 5120 characters. If you require longer strings please contact BNP Paribas Support.

The following listing shows a typical Payment platform response. Payment platform writes the Blowfish-encrypted data into the socket:

```
HTTP/1.0 200 OK

Connection: Close

Content-type: text/plain

Content-Length: 228



Len=125&Data=ECF59EC63E9BEE74DF217E27FA2E194B92597C7395BADBAD4340CDFD000DD57129EEFAA0BE904A7E233ACDBE5EF98E
1692B1602564DBF2C3C75173A62C484962A247B8A91EA7A544
```

The decrypted Payment platform response within the **Data** parameter looks like this:

```
PayID=a234b678e01f34567090e23d567890ce&XID=50f35e768edf34c4e090e23d567890ce&TransID=100000001&Status=AUTHORIZED&Descrip
tion=AUTHORIZED&Code=00000000
```

It is a synchronous communication such that the Socket-connection remains open until the payment platform has supplied the answer. If a request is not answered within 120 seconds, the Payment platform may issue a timeout error message.

Notice: The **URL encoded** parameters are transmitted in key-value pairs (Key1=Value1&Key2=Value2). Please note that new parameters can be added unannounced at any time. Therefore, we recommend the use of the parameter name for the analysis, not the order since this can change at any time. Please do not use case sensitive mechanisms for the spelling of the parameters as this can change at any time.

For more information, please check:

www.w3.org/MarkUp/html-spec/html-spec_8.html#SEC8.2.1

Transaction testing

During the test mode card payments are authorized but there is no cashflow because the Payment platform has not instigated a capture. The exchange will be done between the BNP platform and the PSP.

In the case of successful payments the Payment platform returns the value zero in the **Code** parameter. If a payment fails, the Code parameter is greater than zero, for which there may be many reasons: an incorrect expiry date, an exceeded card limit or even a blocked card are just a few examples. You can find a full list of error codes as an Excel-file in the error codes list.

If you wish to test the different error cases, Payment platform allows you to simulate the desired error codes. To simulate an error, transmit the keyword **Test** in the **OrderDesc** parameter followed by the four-digit detailed error code, for example "Test:0110" to simulate an expired card. Payment platform then returns the four-digit detailed error code with the respective response parameters.

Payment means integration

General information

BNP Paribas's Payment platform processes all major cards and currencies worldwide. Card data is protected against unauthorized access by TLS encryption.

Verified by Visa and MasterCard SecureCode secure your payment claim by password validation if a customer disputes the payment later. American Express SafeKey also uses the 3D-Secure technology, which means that the card holder must confirm their identity with an authentication feature.

Card brands

Card brand, correct spelling for CCBrand
MasterCard
VISA
AMEX
Diners
JCB
Maestro
Cartes Bancaires
DISCOVER
CUP

Definitions

Data formats:

Format	Description
a	alphabetical
as	alphabetical with special characters
n	numeric
an	alphanumeric
ans	alphanumeric with special characters
ns	numeric with special characters
bool	boolean expression (true or false)
3	fixed length with 3 digits/characters
..3	variable length with maximum 3 digits/characters
enum	enumeration of allowed values
dttm	ISODateTime (YYYY-MM-DDThh:mm:ss)

Abbreviations:

Abbreviation	Description
CND	condition
M	mandatory
O	optional
C	conditional

Notice: Please note that the names of parameters can be returned in upper or lower case.

Process of 3D Secure payments

MasterCard SecureCode (UCAF), Verified by Visa (VbV), Diners ProtectBuy, JCB J/Secure and American Express SafeKey are authentication methods which verify the identity of the card holder before making the payment. The name 3D Secure used by technicians describes only the protocol. The correct brand names are Verified by Visa, MasterCard SecureCode, SafeKey, ProtectBuy and J/Secure.

Merchant's PSP benefit from authentication with 3D Secure because the card schemes provide a liability shift: If you are using Verified by Visa, MasterCard SecureCode, Diners ProtectBuy, JCB-Card J/Secure or American Express SafeKey, the burdon of proof and thereby generally the liability is shifted from the merchant to the card issuing bank, should the customer dispute the payment. Irrespective of whether the card holder actually uses the authentication you obtain a very high protection against payment defaults / chargebacks in case the customer states they have not authorized the card payment.

From a technical perspective 3D Secure is not a payment method but an authentication process which precedes the payment: Once the card data has been entered, Payment platform checks the identity of the card holder and does not process the payment until after the authentication.

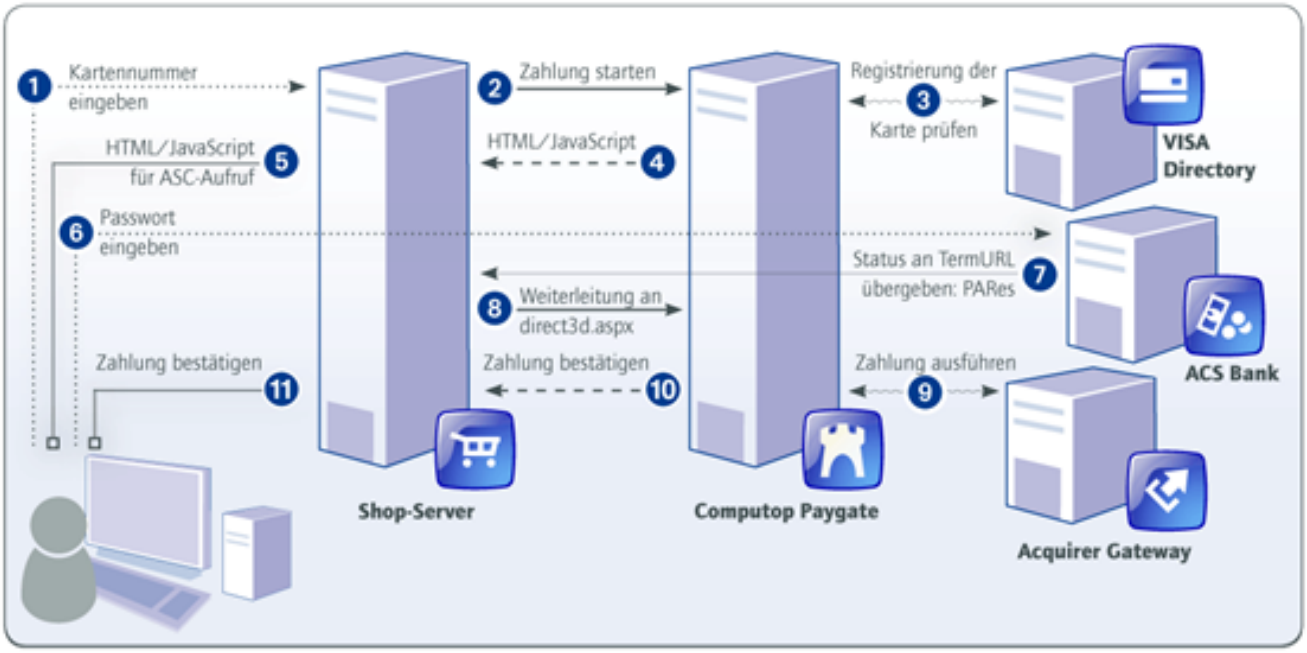
With a Server-to-Server-connection the PSP has to manage the authentication through a separate interface.

Notice: Please make sure to review our EMV 3D Secure Specification to integrate according to the newest standards.

Process of a 3D Secure transaction via Server-to-Server-connection

To carry out authentication, Payment platform connects the card holder to his bank, which confirms the identity. A payment process with Verified by Visa or MasterCard SecureCode, Diners ProtectBuy, JCB-Card J/Secure or American Express SafeKey comprises two steps: authentication and payment.

Following scheme illustrates the processes of a Server-to-Server-payment with 3D Secure.



Communication for card payments with 3D via socket connection

In the next phases there are different cases in which Payment platform responses differ:

- Case 1: card **not registered** for 3D Secure

Payment platform initially contacts Visa or MasterCard, Diners, JCB or American Express Directory Server to determine whether the purchaser's card is registered for Verified or SecureCode or SafeKey.

- Case 2**:** card **registered** for a 3D Secure system

Knowing that the card is registered on the Directory Server, Payment platform returns the following parameters via the socket connection.

Parameter	Format	CND	Description
ACSURL	ans..	C	Only in the case of registered credit cards: URL of the Access Control Server of the card issuer with attached request parameters (not URL-encoded!). It is possible to use ACS sever ampersand and question mark as value within the URL; everything before parameter PAReq is part of ACSURL.
PaReq	ans..	M	Payer Authentication Request, which must be URL-encoded.
MD		M	Merchant Data is an empty value, which must be transferred for compatibility reasons
TermURL	ans..	M	Shop return address. Paygate adds the parameters PayID, TransID and MID as request parameters to the initial TermURL seperated with a question mark.

Response parameters of Socket-connection for the Authentication Request

Notice: Please note in this process that data must sometimes be transferred directly from the bank network. Therefore e.g. the ACSURL parameter is not URL-encoded, although Payment platform uses other URL-encoded data.

These parameters should be included as HIDDEN fields in an HTML page which posts itself to the ACS-URL. The following listing shows one such HTML page, in which the return parameters are embedded:

```
<HTML>
<HEAD>
<META http-equiv=Content-Type content="text/html; charset=unicode">
<A content="MSHTML 6.00.2800.1106" name=GENERATOR>
</HEAD>
<BODY onload="sendpareq.submit();">
<FORM action="[ACSURL]" method="POST" name="sendpareq">
<input type="hidden" name="MD" value="">
<input type="hidden" name="PaReq" value="[PaReq]">
<input type="hidden" name="TermUrl" value="[TermUrl]">
</FORM>
</BODY>
</HTML>
```

Notice: You can also use this code if you only want to verify the identity of the card holder without immediately making a card payment (Authentication Hosting). BNP Paribas Support can configure your checkout so that Payment platform can carry out Verified by Visa or SecureCode without payment.

After the customer has been authenticated with its bank, the bank's Access Control Server (ACS) requests the TermURL in the shop. In the case of this Request the ACS transfers the following parameters via GET (QueryString) to the TermURL of the shop: MID, PayID and TransID (unencrypted). The PARes parameter is transferred unencrypted via POST.

Notice: The PAResponse parameter must be URL encoded but not Blowfish-encrypted since the content can include special characters.

The parameter must be transferred in whole via POST to the following URL:

https://paymentpage.epayment.bnpparibas/direct3d.aspx

Notice: If you forward the PARes and MID of the ACS parameters please use the specified parameter name MerchantID, PAResponse for the direct3d.aspx page.

Call of interface: general parameters

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

Credit card still must be valid at time of capture / refund. Therefore BNP accepts credit cards when the card is at least 1 week valid before expire (e.g.: CC expire: 2020-03 authorizations possible until 2020-03-24, 23:59:59).

To carry out a TLS card payment via a Server-to-Server connection, call the following URL:

https://paymentpage.axepta.bnpparibas/direct.aspx

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

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	Merchant ID. This parameter is to be passed in plain language.
TransID	ans..64	M	TransactionID which should be unique for each payment

RefNr	an..12	OC	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file we cannot add the additional payment data.
			
			Notes:
			
			• BNP requires
			  ○ fixed length of 12 characters, BNP automatically align value to the right and add spaces to the left
			  ○ only characters (A..Z, a..z) and digits (0..9) are allowed, no characters like underscore, minus, ...
			• CAPN: RefNr is mandatory

Amount	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
Amount3D	n..10	OC	Only for 3DSecure: Amount for authentication with Verified, SecureCode and SafeKey if Amount deviates. E.g. Customer confirms flight costs of 120 Euros with 'Verified' but the travel agent captures only the booking fee of 20 Euros: **Amount3D=12 000; Amount=2000**. Amount in the smallest currency unit (e.g. EUR Cent) Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
Currency	a3	M	Currency, three digits DIN / ISO 4217
CCNr	n..19	M	Card number at least 12-digit, numerical without spaces. You can optionally transmit also a pseudo card number (PCN)
CCCVC	n..4	O	Card verification number (CVV): for Visa and MasterCard the last 3 numbers on the signature strip of the card. For American Express the last 4 numbers.
CCExpiry	n6	M	In combination with TOKEN: Expiry date of the card in the format YYYYMM (201706).
CCBrand	a..22	M	Designation of card brand. Please note the spelling! According to table of card brands!
Capture	ans..6	O	Determines the type and time of capture. **AUTO**: capturing immediately after authorisation (default value). **MANUAL**: capturing made by the merchant. **<Number>:** Delay in hours until the capture (whole number; 1 to 696).
OrderDesc	ans..768	M	Description of purchased goods, unit prices etc.
TermURL	ans..256	C	Only for 3DSecure. Shop URL selected by the ACS (Access Control Server) of the cardholder's bank for the transmission of authentication result. The bank transmits the following parameters: PayID, TransID and MerchantID via GET et the parameter PAResponse via POST to the TermURL.
UserAgent	ans..128	C	Only for 3DSecure. User's browser type calling the page. For exemple: IE Mozilla/4. 0 (compatible ; MSIE 6.0 ; Windows NT 5.0 ; NET CLR 1.0.3705)
HTTPAccept	ans..128	C	Only for 3DSecure: MIME types that the merchant's clients accept. E.g. image/gif, image/x-xbitmap, image/jpeg, image /pjpeg, application/vnd. ms-powerpoint, ap-plication/vnd. ms-excel, application/msword, */*
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
ReqID	ans..32	O	To avoid double payments, enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction is submitted again with the same ReqID, the payment platform will not carry out the payment, but will just return the status of the original transaction. Please note that the payment platform must have a finalized transaction status for the first initial action. Submissions with identical ReqID for an open status will be processed regularly.
AccVerify	a3	O	If AccVerify=Yes the card will be checked at the acquirer according to the ac-quirer's interface description. The merchant has to submit only this parameter; the parameter "Amount" is optional. If "Amount" is used we replace the amount according to acquirer's interface description. At payment always Amount=0 is stored. Allowed value: yes
RTF	a1	O	For regular payment (Subscription) : I = Initial payment for the new subscription R = Recurring payment
CONTRACTID	n..8	O	Further reference which can be used to retrieve the combination TerminalID/Contract partner number

General parameters for card payments via Socket connection

The following table gives the parameters with which the Payment platform responds:

Parameter	Format	CND	Description
MID	ans..30	M	MerchantID
PayID	an32	M	ID assigned by the payment platform for the payment, e.g. for referencing in batch files
XID	an32	M	ID for all single transactions (authorisation, capture, refund note) for one payment assigned by the payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or AUTHORIZED (URLSuccess) as well as FAILED (URLFailure)
Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!

Code	n8	M	Error code according to the payment platform Response Codes
PCNr	n16	O	TOKEN (Pseudo Card Number): Random number generated by the payment platform which represents a genuine card number. The TOKEN (PCN) starts with 0 and the last 3 digits correspond to those of the real card number. You can use the PCN like a genuine card number for authorisation, capture and refunds.
CCExpiry	n6	OC	In combination with TOKEN: Expiry date of the card in the format YYYYMM (201706).
CCBrand	a..22	OC	In combination with PCNr: Designation of card brand Please note the spelling! According to table of card brands!
MaskedPan	an..19	OC	Masked card number 6X4
CAVV	ans..40	OC	In the case of 3D Secure with Authentication Hosting (only 3D request without authorisation): Cardholder Authentication Validation Value: Contains the digital signature for authentication with the ACS of the card issuing bank.
ECI	n2	OC	For 3D Secure: ACS E-Commerce indicator: defines the security level of a card payment via different communication paths: MOTO, SSL, Verified by Visa etc.
DDD	a1	C	For 3D Secure Authentication Hosting: Y - fully authenticated (complete authentication done) N - not enrolled (checked, but Issuer does not participate) U - uneledgeble (technical error) A – attempt (card does not participate) B – bypass (bypass, only for Cardinal Commerce)
ACSXID	ans..40	O	Only for cases 2 and 3 (with pop up and without pop up: page 31 and 32 of this guide) with hosting authentication: ID to identify the transaction. The ACSXID is transferred with the authorisation to the acquirer

General parameters for answers of card payments via Socket connection

Card payment management

Capture

Captures are possible via a Server-to-Server connection. To perform captures via a Server-to-Server connection please use the following URL:

https://paymentpage.axepta.bnpparibas/capture.aspx

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

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	Merchant ID, assigned by BNP
PayID	an32	M	ID assigned by Payment platform for the payment
TransID	ans..64	M	TransactionID which should be unique for each payment
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
Amount	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
Currency	a3	M	Currency, three digits DIN / ISO 4217
RefNr	an..12	C	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file we cannot add the additional payment data.
ReqID	ans..32	O	To avoid double payments, enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction is submitted again with the same ReqID, the payment platform will not carry out the payment, but will just return the status of the original transaction. Please note that the payment platform must have a finalized transaction status for the first initial action. Submissions with identical ReqID for an open status will be processed regularly.
Textfeld1	ans..30	O	Card holder information: Name

Textfeld2	ans..30	O	Card holder information: City

Parameters for captures of card payments

The following table describes the Payment platform‘s response parameters:

Parameter	Format	CND	Description
MID	ans..30	M	MerchantID
PayID	an32	M	ID assigned by Payment platform for the payment, e.g. for referencing in batch files
XID	an32	M	ID for all single transactions (authorisation, capture, refund) for one payment assigned by the payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or FAILED
Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!
Code	n8	M	Error code according to the payment platform Response Codes Excel file
RefNr	an..12	C	Merchant's unique reference number, which serves as payout reference in the acquirer EPA file. Please note, without the own shop reference delivery you cannot read out the EPA transaction and regarding the additional settlement file we cannot add the additional payment data.

Response parameters for captures of card payments

Refunds

Refunds are possible via a Server-to-Server connection. The payment platform permits refunds which relate to a capture previously activated by the payment platform and allows merchants to carry out refunds without a reference transaction. This section describes the processing of refunds with reference transactions. If you refer to a capture for a refund, the amount of the refund is limited to the amount of the previous capture.

To carry out a refund with a reference transaction, please use the following URL:

https://paymentpage.axepta.bnpparibas/credit.aspx

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

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	Merchant ID, assigned by BNP
PayID	an32	M	ID assigned by the payment platform for the payment
TransID	ans..64	M	TransactionID which should be unique for each payment
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
Amount	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
Currency	a3	M	Currency, three digits DIN / ISO 4217
OrderDesc	ans..768	O	Merchant's reference number
ReqID	ans..32	O	To avoid double payments, enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction is submitted again with the same ReqID, the payment platform will not carry out the payment, but will just return the status of the original transaction. Please note that the payment platform must have a finalized transaction status for the first initial action. Submissions with identical ReqID for an open status will be processed regularly.
Textfeld1	ans..30	O	Card holder information: Name
Textfeld2	ans..30	O	Card holder information: City

Parameters for refunds of card payments

The following table describes the Payment platform response parameters:

Parameter	Format	CND	Description
MID	ans..30	M	MerchantID
PayID	an32	M	ID assigned by Payment platform for the payment, e.g. for referencing in batch files
XID	an32	M	ID for all single transactions (authorization, capture, refund) for one payment assigned by the payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or FAILED
Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!
Code	n8	M	Error code according to the payment platform Response Codes Excel file

Response parameters for refunds of card payments

Cancellation

A card authorization lowers the customer's authorization line. Payment platform can reverse an authorisation so that it no longer blocks the limit any more. Use the following URL:

https://paymentpage.axepta.bnpparibas/reverse.aspx

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

Notice: Reverse.aspx does not only reverse authorizations, but any last transaction stage. If the last transaction was a capture, Reverse.aspx initiates the reverse, e.g. a refund. Therefore, the utmost caution is urged. Use is at your own risk. We recommend checking the transaction status with Inquire.aspx before using Reverse.aspx.

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	MerchantID
PayID	an32	M	Payment platform ID for the identification of a payment
TransID	ans..64	M	TransactionID which should be unique for each payment
Amount	n..10	M	Amount in the smallest currency unit (e.g. EUR Cent). Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
Currency	a3	M	Currency code, three digits DIN / ISO 4217
MAC	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
ReqID	ans..32	O	To avoid double payments, enter an alphanumeric value which identifies your transaction and may be assigned only once. If the transaction is submitted again with the same ReqID, the payment platform will not carry out the payment, but will just return the status of the original transaction. Please note that the payment platform must have a finalized transaction status for the first initial action. Submissions with identical ReqID for an open status will be processed regularly.

Response parameters for cancellation of card payments

The following table describes the payment platform response parameters:

Paramater	Format	CND	Description
MID	ans..30	MC	MerchantID
PayID	an32	M	ID assigned by the payment platform for the payment, e.g. for referencing in batch files
XID	an32	M	ID for all single transactions (authorisation, capture, refund) for one payment assigned by the payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or FAILED

Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!
Code	n8	M	Error code according to the payment platform Response Codes Excel file

Response parameters for reversals of card payments

Status inquiries

Status inquiries based on PayID

Status inquiries within Payment platform give detailed information about the amounts that are actually authorized, captured or credited. Especially before executing reversals via the interface reverse.aspx it is recommended to check the transaction status with inquire.aspx, because Reverse.aspx reverses not only authorizations but ALWAYS THE LAST TRANSACTION STEP.

Inquiries of transaction status based on PayID are possible via a Server-to-Server connection. In order to inquire about the status of a transaction via a Server-to-Server connection, please use the following URL:

https://paymentpage.axepta.bnpparibas/inquire.aspx

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

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	ID of merchant. Additionally this parameter has to be passed in plain language too.
PayID	an32	M	ID for identifying a transaction given by Payment platform
TransID	ans..64	M	TransactionID which should be unique for each payment

Parameters for status inquiries via socket connections

The following table describes the Payment platform response parameters:

Parameter	Format	CND	Description
MID	ans..30	M	ID of merchant
PayID	an32	M	ID assigned by Payment platform for the payment, e.g. for referencing in batch files.
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by Payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or FAILED
Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!
Code	n8	M	Error code according to Payment platform Response Codes Excel file
AmountAuth	n..10	O	Approved amount
AmountCap	n..10	O	Captured amount
AmountCred	n..10	O	Credited amount
LastStatus	a..50	O	Status of last transaction (Authorisation, Capture or Credit)

Response parameters in the case of status inquiries via socket connections

Notice: The **LastStatus** parameter is not sufficient to determine the status because it supplies the status of only the last transaction. For example LastStatus delivers the status **Failed** if a credit has failed although the previous authorisation and capture were successful. The fields **AmountAuth**, **A mountCap** and **AmountCred** give the precise status.

Status inquiries based on TransID

Inquiries of transaction status based on TransID are possible via a Server-to-Server connection. In order to inquire about the status of a transaction via a Server-to-Server connection, please use the following URL:

https://paymentpage.axepta.bnpparibas/inquire24.aspx

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

The following table describes the encrypted payment request parameters:

Parameter	Format	CND	Description
MerchantID	ans..30	M	ID of merchant. Additionally this parameter has to be passed in plain language too.
TransID	ans..64	M	TransactionID which should be unique for each payment

Parameters for status inquiries via socket connections

The following table describes the Payment platform response parameters:

Parameter	Format	CND	Description
MID	ans..30	M	ID of merchant
PayID	an32	M	ID assigned by Payment platform for the payment, e.g. for referencing in batch files
XID	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by Payment platform
TransID	ans..64	M	Merchant's transaction number
Status	a..50	M	OK or FAILED as status of inquiry and not of the requested transaction
Description	ans..1024	M	Further details in the event that payment is rejected. Please do not use the **Description** but the **Code** parameter for the transaction status analysis!
Code	n8	M	Error code of inquiry not of the requested transaction according to Paygate Response Codes Excel file
AmountAuth	n..10	O	Approved amount
AmountCap	n..10	O	Captured amount
AmountCred	n..10	O	Credited amount
LastStatus	a..50	O	Status of last transaction (Authorisation, Capture or Credit)
LastXCode	n8	O	Error code of last transaction according to Payment platform Response Codes Excel file
LastXAmount	n..10	O	Amount in the smallest currency unit (e.g. EUR Cent)

Response parameters in the case of status inquiries via socket connections

Notice: The **LastStatus** parameter is not sufficient to determine the status because it supplies the status of only the last transaction. For example LastStatus delivers the status **Failed** if a credit has failed although the previous authorisation and capture were successful. The fields **AmountAuth**, **A mountCap** and **AmountCred** give the precise status.