Eximbay Online Payment

Technical
Integration Guide

EXIMBAY

Contact Information

Technical Support	Email : TechnicalSupport@eximbay.com
Operation Support	Email : OnlineSupport@eximbay.com

If you have any suggestion or remark about this document please write an email to us.

Copyright

Contents

Recent revisions

Date	Description
2017.04.11	Initial creation of document (based on v2.3 rev 2)
2021.02.01	Add payment method (Diners, Discover)
2021.02.23	Add payment method (Payco, Virtual Account, SmilePay)
2021-11-16	Add payment method (Mir)
2022-02-23	Updated (up to 2022-01-26 of Eximbay Online Global Payment Integration Guide v2_3_kr – rev 10)
2022-03-23	Add payment method (2C2P)
2022-04-12	Add multi_paymethod parameter
2022-04-14	Table of Contents 4.4 Add linepay and alipay+ for app-to-app settings
2022-11-15	Alipay + add payment method due to v1.8 upgrade
2023-03-29	Add payment method(econtext-Linepay)

1 Overview

This guide is intended for developers who want to integrate Eximbay into their online payment system.

1.1 Communication Method

1.1.1 BasicProcessor.krp

When EndPoint is BasicProcessor.krp, the request data for each task is requested by the URL defined by HTTP POST method through web-based communication, and the response data is returned by the user browser-based method(returnurl) and the web server-based method (statusurl) through the back-end.

⚠ Important

The browser based Front-end(returnurl) URL

URL where a user will be redirected to upon payment completion and payment result will be returned to merchant's page in the browser. User might close their browser any time throughout the payment process. As a result, merchant might not receive any payment result with the URL. Therefore, it is recommended to use this URL only for displaying the payment result to the customer.

⚠ Important

The server based Back-end(statusurl) URL

URL where payment result will be sent from Eximbay server to merchant's server in background. This is a back-end server-to-server operation that javascript, cookie and session could not be used. It is recommended to use this URL to update payment status in merchant system.

Due to Eximbay notification mechanism, same payment result can be sent with the Back-end URL(statusurl) multiple times to the merchant system. Therefore, it is advised to prevent duplication in updating merchant system.

1.1.2 DirectProcessor.krp

When EndPoint is DirectProcessor.krp, it is web-based communication, and request data for each task is requested through the URL defined by HTTP POST method.

And it returns the response value in the HTTP GET format method of text method as a result value without providing UI.

(data format : name1=value1&name2=value2&.....)

1.2 Data Format

Field Name starts with a lowercase letter and it is case sensitive.

Data Type	Description
Integer	Whole number {..., -3, -2, -1, 0, 1, 2, 3, ...}
String	Sequence of letters, numbers, spaces, and special characters

"R" - Required, "C" - Conditional, " " – Optional.

2 Online Payment

Online payment in this document refers to credit card + online Third-party payments and provides an integrated linked document

Online third-party payments means the payment method provided by Eximbay excluding credit cards such as Paypal, Alipay, Wechat, Tenpay, etc.

Warning

Unable to support iframe, please use *displaytype='P'* or *displaytype='R'*.

Credit Card

The authentication, authorization and capture are requested in one transaction at the same time. And the transaction is automatically captured on the next day.

3D Secure(Payer Authentication), DM, and DCC services can be additionally performed according to ***Eximbay*** settings.

Summary of the online Credit Card Payment process

txntype	UI	Request URL	Summary
PAYER_AUTH	O	....../BasicProcessor.krp	Card Issuer Authentication(3D) process (DM/DCC can be optionally processed)
AUTHORIZE_PA	X	....../DirectProcessor.krp	Credit Card Authorization process. EXIMBAY does not capture the authorization until CAPTURE request is received or merchant separately do the capture.
AUTH_REVERSAL	X	....../DirectProcessor.krp	Authorization Reversal process (Only available before capture) If merchant separately do the capture, it cannot be used.
CAPTURE	X	....../DirectProcessor.krp	Authorization capture request. Automatically processed in +1 working day.

As mentioned above, there are 4 types of transactions(txntype), and they can be combined as in the below table.

txntype	UI	Request URL	Combination
AUTHORIZE	O	……/BasicProcessor.krp	PAYER_AUTH & AUTHORIZE_PA
PAYMENT_PA	X	……/DirectProcessor.krp	AUTHORIZE_PA & CAPTURE
PAYMENT	**O**	**……/BasicProcessor.krp**	**PAYER_AUTH & AUTHORIZE_PA & CAPTURE**

Note. 1 –Based on *txntype,* request/response parameters can be different. For more information, you can contact us through the contact details in Contact Information.

Online Third-Party Payments

The authentication, authorization and capture are performed in one transaction from the payment processor.

Summary of the online third-party payment process

txntype	UI	Request URL	Combination
PAYMENT	**O**	**……/BasicProcessor.krp**	**PAYER_AUTH & AUTHORIZE_PA & CAPTURE**

Note. 1 – Online third-party payments include Paypal, Alipay, WeChat, Tenpay etc…

Note. 2 – For third-party payments, all authentication, authorization and capture are performed by a payment processor.

2.1 Performing a Sale & Authorizing a Payment

Using the UI provided by Eximbay, you can perform two approval tasks.

PAYMENT includes ***CAPTURE*** work and automatic billing will be carried out next day.

AUTHORIZE is charged only when purchased directly from a merchant or when ***CAPTURE*** is sent to Eximbay.

2.1.1 Request URL

● Test	https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp
● Production	https://secureapi.eximbay.com/Gateway/BasicProcessor.krp

2.1.2 Request Parameters

Field Name	**Type**	**Length**	**Required**	**Description**
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"**PAYMENT**"
ref	String	30	R	Merchant-generated unique order number or tracking number for each transaction including failed one. (e.g. orderid)
cur	String	3	R	ex. USD, SGD, KRW...(refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
paymethod	String	4		Payment Method Code (Refer Appendix C) This parameter is used when you want to call specific payment method.
multi_paymethod	String	255		Payment Method Code (Refer Appendix C) When designating multiple payment methods, it is required to set them into groups through a separator. (Separator : "-") Ex. P000-P185-P186
shop	String	255		Shop Name(Required when Shop name is different from Merchant name)
buyer	String	64	R	Buyer Name (Required to use real name)
tel	String	32		Buyer's Contact Number
email	String	32	R	Buyer's email address(Buyer will receive email notification)
lang	String	2	R	Payment screen language (Refer to Appendix B)
returnurl	String	255	R	Merchant page that is called when the user exits the payment screen on the payment result

				confirmation screen (Since returnurl operates based on the customer's browser, it may not be called if the browser is forcibly closed.)
statusurl	String	255	R	After payment processing is completed, it is the merchant page called from Backend, and the returnurl and parameter are the same. Since it is not called from the browser, scripts, cookies, and sessions cannot be used. DB operation and payment process must be handled by statusurl, and returnurl may not be called depending on the payment method or when the customer forcibly closes the payment screen. Since statusurl can be called in duplicate, please make sure that your order is not processed in duplicate.
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
charset	String	32		Default charset – UTF-8 (Refer to Appendix C)
fgkey	String	64	R	Refer to “4.1 FGKEY”
partnercode	String	10		Partner Code (assigned by KRP)
item_#_product	String	255	R	Item Name
item_#_quantity	String	10	R	Item Quantity (Positive number not less than 1)
item_#_unitPrice	String	15	R	Item Price (Do not use comma ‘,’. Must be more than 0)
Item_#_link	String	255	R	Item Link (Open market)
surcharge_#_name	String	255		Additional charge item name (e.g. discount(-), shipping fee(+) etc..)
surcharge_#_quantity	String	10		Additional charge item quantity (Must be more than 1)
surcharge_#_unitPrice	String	15		Discount amount per item (Do not use comma ‘,’ and negative amount allowed) (e.g. -1000.50, 9.15)
shipTo_city	String	50	R	Shipping city (e.g. Hanoi, Brisbane, Houston)
shipTo_country	String	2	R	Shipping country. Must be ISO3166 country code (e.g. KR, US..)
shipTo_firstName	String	60	R	First name of the person receiving product
shipTo_lastName	String	60	R	Last name of the person receiving product
shipTo_phoneNumber	String	15	R	Phone number for the shipping address(Including country code)

shipTo_postalCode	String	20	R	Postal code for the shipping address
shipTo_state	String	20	R	State or province of the shipping address. Required only when shipping country is US or CA. (e.g. MA, NY, CA) (Refer to Appendix D)
shipTo_street1	String	100	R	Street of the shipping address (e.g. 123 Main street, 56 Le Loi street)
billTo_city	String	50		Billing city (e.g. Hanoi, Brisbane, Houston)
billTo_country	String	2		Billing country. Must be ISO3166 country code (e.g. KR, US..)
billTo_firstName	String	60		Cardholder first name
billTo_lastName	String	60		Cardholder last name
billTo_phoneNumber	String	15		Cardholder contact number(including country code)
billTo_postalCode	String	20		Postal code for the billing address
billTo_state	String	20		State or province of the billing address. Only used for US or CA. (e.g. MA, NY, CA) (Refer to Appendix D)
billTo_street1	String	100		Street of the billing address(e.g. 123 Main street, 56 Le Loi street)
dfs_deliverer	String	100	R	For duty-free use-shipper (requires duty-free shop)
dfs_delivery	String	400	R	Duty free-delivery (required at duty-free shops)
dfs_delivery_date	String	8	R	For duty-free use-Delivery date (required for duty-free shops) (ex. 20200101)
traveldt	String	6	R	For airline-travel date (airline required) (ex. 200101)
ticketnum	String	14	R	For airline-ticket number (airline required)
passengernm	String	100	R	For airline-passenger name (airline required)
ticketissueagentid	String	50	R	For airlines-Ticket issuing agency ID (required by airline)
leaveplacecd	String	3	R	For airlines-Departure (destination) (required by airline)
flightnm	String	6	R	For airline-flight number (airline required)
ostype	String	1		P : pc version((default)) M : mobile
displaytype	String	1	R	P : popup (default) R : page redirect
siteforeigncur	String	3		This is the customer-selected currency of the merchant site, and can be used when you want to display it on the first screen of the payment screen. - When using paymethod, only 'P000' is allowed -Only DCC-supported currencies of DCC-enabled merchants are supported, and foreign amounts

				are displayed by applying real-time exchange rates searched through DCC Provider in Eximbay. -After entering the final card information, it may be changed to the currency of the card issuing country.
callfromapp	String	1		Classification of client environment Y : When called in Android and IOS APP environment N : When called from a web browser environment Default : N
callfromscheme	String	40		For merchant using APP APP Url Scheme for focusing back to merchant APP ex) exb.upay.sample, eximpay

Note. 1 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document..

Note. 2 – The *ref* parameter is allowed to have duplicate values. You can contact us If you want to prevent from duplicate payments in accordance with *ref*.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters.

Note. 4 – The *partnercode* is used for specific process required for a particular merchant. And the merchant need to consult with us before using this parameter.

Note. 5 – *item_#* - you can specify this parameter up to item numbers. *surcharge_#* - you can specify this parameter up to additional item-specific discounts or charges. # starts from 0.

(*item_#_quantity* * *item_#_unitPrice*) + (*surcharge_#_quantity* * *surCharge_#_unitPrice*) must be equal to *amt*.

Note. 6 – The shipping address *shipTo_** parameters are required in order to prevent from chargebacks. For more information, you can ask detail information from our Operation Support team.

Note. 7 – When **PayPal** payment is used, **"*item_**"** and **"*shipTo_**"** parameters are sent to PayPal and their values are shown on the PayPal page. In case if invalid information is sent for these parameters, PayPal can decline the payment. So it is recommended to provide valid information to these parameters.

2.1.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"**PAYMENT**"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.
cur	String	3	R	e.g. USD, SGD, KRW…(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
email	String	32	R	Buyer Email Address
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2

param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Transaction ID
rescode	String	6	R	Response Code. (Success:'0000', Refer to Chapter 4)
resmsg	String	700	R	Response Message
authcode	String	8	C	Approval Code (Only provided for credit card payment)
resdt	Integer	14	R	YYYYMMDDHHMMSS
cardholder	String	50	C	Cardholder's Name in English
accesscountry	String	2	R	Customer's accessed country. ISO 3166 country codes (e.g. KR, JP, US…)
fgkey	String	64		Refer to "4.1 FGKEY"
cardno1	Integer	4	C	First 4 digits of card number
cardno4	Integer	4	C	Last 4 digits of card number
paymethod	String	4	C	Payment Method Code (Refer to Appendix C)
payto	String	64	C	Merchant Name that receiving payment
shipTo_city	String	50		Shipping city (e.g. Hanoi, Brisbane, Houston)
shipTo_country	String	2		Shipping country. ISO3166 country code (e.g. KR, US..)
shipTo_firstName	String	60		First name of the person receiving product
shipTo_lastName	String	60		Last name of the person receiving product
shipTo_phoneNumber	String	15		Phone number for the shipping address(Including country code)
shipTo_postalCode	String	20		Postal code for the shipping address
shipTo_state	String	20		State or province of the shipping address. Available only if shipping country is US or CA. (e.g. MA, NY, CA)
shipTo_street1	String	100		Street of the shipping address (e.g. 123 Main street, 56 Le Loi street)
status	String	20	C	(Cash payment)"**Registered**" or "**Sale**" Registered means an order registration, only when "Sale" status is returned, you need to consider that the order is completed. "Sale" means that payment is completed. And this status is returned with statusurl. For other payment statuses, refer to 3.2 Cash Payment Notice.
paymentURL	String	Variable	C	(Cash payment) When "Japanese Convenience Store/Online Banking" payment is used, this payment instruction URL is sent to customer upon they finish their order registration.
dm_decision	String	6	C	Returns when DM service is used. It contains one of following values : ACCEPT, ERROR, REJECT, and REVIEW. ACCEPT: Transaction approval after DM Process risk

				review. REJECT: Transaction failure after DM process risk review REVIEW: DM Process Approval can be processed after risk review, but it is necessary to confirm the risk of transaction. ERROR: DM service failed.
dm_reject	String	Variable	C	Returns CyberSource DM REJECT Rule Codes. The codes are separated by '|'.
dm_review	String	Variable	C	Returns CyberSource DM REVIEW Rule Codes. The codes are separated by '|'.
basecur	String	3	C	Transaction Currency (Refer to Note.6) (Usually it is same with cur. When Reverse-DCC is used, either KRW or USD currency can be returned. It can also be used for the capture data if the merchant is direct acquirer contracted.)
baseamt	String	15	C	Transaction Amount (Refer to Note.6) (Usually it is same with amt. When Reverse-DCC is used, either KRW or USD amount can be returned. It can also be used for the capture data if the merchant is direct acquirer contracted.)
baserate	String	14	C	Transaction Currency Exchange Rate (Refer to Note.6) (Exchange rate from cur to basecur. If cur and basecur are same, it is 1)
foreigncur	String	3	C	Customer's billing currency (Refer to Note.6) (When DCC is used, customer's selected currency is returned)
foreignamt	String	15	C	Customer's billing amount (Refer to Note.6) (When DCC is used, customer's accepted amount is returned)
foreignrate	String	14	C	Customer's billing currency exchange rate (Refer to Note.6) (Exchange rate from cur to foreigncur. If cur and foreigncur are same, it is 1)
dccrate	String	14	C	DCC Exchange Rate (Refer to Note.6) (Exchange rate from basecur to foreigncur. If basecur and foreigncur are same, it is 1) When DCC service is used by direct acquirer contract merchants, this exchange rate is used for a partial refund batch process.
tid	String	10	C	VAN TID (Refer to Note.6)
cardno	String	30	C	Card number. (Refer to "4.3 AES 256 encryption") (Refer to Note.6)
expirydt	String	30	C	Expiration date. (Refer to "4.3 AES 256 encryption") (Refer to Note.6)
memberno	Strnig	15	C	Acquirer merchant number (Refer to Note.6)

Note. 1 –*ver* ~ *param3* parameters are returned same as they are sent in the request.

Note. 2 –The response is sent to *returnurl* and *statusurl* which are set in the request.

Note. 3 – *cardholder* can be different from the name printed on the card.

Note. 4 – if *statusurl* is not received properly, it is sent repeatedly 3 times, the repeat rule can be different depends on the configuration.

Note. 5 – In case **DM** is used, *dm_** response parameters are returned.

Note. 6 – In case of Direct Merchant, *basecur* ~ *memberno* parameters are returned.

2.2 Payer Authentication

Payer Authentication refers to the card issuer's identity verification, and DM and **DCC may be added** depending on Eximbay settings. After success authentication processing, you can proceed with the approval processing through ***PAYMENT_PA*** or ***AUTHORIZE_PA***.

2.2.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/BasicProcessor.krp

2.2.2 Request Parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"**PAYER_AUTH**"
(The following is the same as the **2.1.2 request parameter**)				

2.2.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"**PAYER_AUTH**"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.
cur	String	3	R	e.g. USD, SGD, KRW…(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
email	String	50	R	Buyer Email Address
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
payerauthid	String	24	C	PAYER_AUTH Transaction id

				After that, use PAYMENT_PA or AUTHORIZE_PA.
rescode	String	6	R	Response Code. (Success:'**0000**')
resmsg	String	700		Response Message
fgkey	String	64	R	Refer to "4.1 FGKEY"
dm_decision	String	6		Returns when CyberSource DM is used. It contains one of following values : ACCEPT, ERROR, REJECT, and REVIEW.
dm_reject	String	Variable		The REJECT Rule Code value is returned. Rule code is divided into'|', and information on rule code is provided separately.
dm_review	String	Variable		The REVIEW Rule Code value is returned. Rule code is divided into'|', and information on rule code is provided separately.
cavv	String	40		Certification result cavv
xid	String	40		Certification result xid
eci	String	28		Certification result eci

Note. 1 –ver ~ param3 parameters are returned same as they are sent in the request.

Note. 2 –The response is sent to returnurl and statusurl which are set in the request.

Note. 3 – Returns cavv, xid, eci as the result of Payer Authentication.

Note. 4 – If the statusurl is not transmitted normally, it is retransmitted 3 times, and the retransmission rule may be different depending on the settings.

Note. 5 – When using DM, the dm_* parameter values are transmitted.

2.3 Performing a Sale and Authorizing a Payment with Payer Authentication

First, perform the Payer Authentication process separately. It is then used to process PAYMENT (**PAYMENT_PA**) or AUTHORIZE (**AUTHORIZE_PA**) using the non-UI method.

AUTHORIZE is charged only when purchased directly from a merchant or when **CAPTURE** is sent to Eximbay.

Note

Authorization transactions are capped after approximately 7 to 14 days depending on the card issuer.

Important

In the case of direct merchants, it is possible to transmit billing purchases directly from the merchant through VAN using the tid ~ memberno response parameter, and when using DCC service, tid and memberno are transmitted differently depending on whether the customer selects the local currency Therefore, you must purchase the bill using the return value.

2.3.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/DirectProcessor.krp

2.3.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	**PAYMENT_PA" or "AUTHORIZE_PA"**
ref	String	30	R	Merchant-generated unique order number or tracking number for each transaction
cur	String	3	R	e.g. USD, SGD, KRW…(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0) (Should be the same as the request parameter value of **PAYER_AUTH**)
payerauthid	String	24	R	PAYER_AUTH Transaction id (**PAYER_AUTH** response parameter value)
lang	String	2	R	Payment screen language (Refer to Appendix B)
returnurl	String	255		Page to be moved, after processing
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "4.1 FGKEY"
charset	String	32		Default charset – UTF-8

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

2.3.3 Response parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	**"PAYMENT_PA" or "AUTHORIZE_PA"**

(The following is the same as the **2.1.3 response parameter**)

2.4 Capturing an Authorization

Use to request a claim for an **AUTHORIZE** or **AUTHORIZE_PA** transaction.

2.4.1 Request URL

● Test	https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp
● Production	https://secureapi.eximbay.com/Gateway/DirectProcessor.krp

2.4.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	**"CAPTURE"**
ref	String	30	R	Original approved transaction ref
cur	String	3	R	Originally approved transaction currency
amt	String	15	R	Originally approved transaction amount (e.g. 1000.50, 9.15)
transid	String	24	R	Transaction ID of the original approved transaction
lang	String	2	R	Payment screen language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "4.1 FGKEY"
charset	String	32		Default charset – UTF-8

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – Capturing an Authorization does not support capturing partial amounts.

2.4.3 Response parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"**CAPTURE**"
ref	String	30	R	Merchant-generated unique order number or tracking number. Same value returns.

cur	String	3	R	e.g. USD, SGD, KRW...(Refer to Appendix A.)
amt	String	15	R	Order total amount. (e.g. 1000.50, 9.15) (Do not use comma ‘,’. Must be more than 0)
lang	String	2	R	Payment screen language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Transaction id
rescode	String	6	R	Response Code. (Success:’0000’, Refer to 4.2)
resmsg	String	700	R	Response Message
resdt	Integer	14	R	YYYYMMDDHHMMSS
allowedpvoid	String	1		Y: Partial cancellation is possible, N: Partial cancellation is not possible (default)
fgkey	String	64		Refer to “4.1 FGKEY”

2.5 Performing a Sale (Token Creation)

Token Creation is a service for online third-party payments of cards issued overseas. First payment approval is the same as Eximbay's general payment approval method, and this task is processed through the UI provided by Eximbay.

The **tokenID** is issued when payment approval is successful, and the ID is mapped to the member information of the merchant site and used for **Token Payment (Rebill).**

A separate review is required when using token billing, so please consult with Eximbay before proceeding.

Please note that services that are not allowed in advance may be subject to future restrictions.

.

Note

Performing a Sale (Token Creation) only supports txntype=PAYMENT,

Duplicate registration is allowed for already registered cards.

2.5.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/BasicProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/BasicProcessor.krp

2.5.2 Request Parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"**PAYMENT**"
tokenBilling	String	1	R	Whether to use TokenBilling. (**Y**: use, **N**: do not use)
(The following is the same as the **2.1.2 request parameter**)				

2.5.3 Response Parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"**PAYMENT**"
tokenID	String	20	R	Token ID to be used for rebill (issued only when approval is successful)
tokenBilling	String	1	R	Whether to use TokenBilling. (**Y**: use, **N**: do not use)
(The following is the same as the **2.1.3 response parameter**)				

Note. 1 –TokenBilling allows duplicate registration for already registered cards.

2.6 Performing a Sale (Token Payment)

Token Payment is an API that allows you to conveniently pay without entering additional card information through the tokenID issued through **Token Creation**.

2.6.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/DirectProcessor.krp

2.6.2 Request Parameters

Field Name	Type	Length	Required	Description
txntype	String	30	R	"**REBILL**"
tokenID	String	20	R	TokenID received in response when performing a sale (Token Creation)
(The following is the same as the **2.1.2 request parameter**)				

2.6.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	Web-Payment Integration Guide ver
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	**"REBILL"**
ref	String	30	R	Merchant-generated unique order number or tracking number for each transaction including failed one. (e.g. orderid)
cur	String	3	R	ex. USD, SGD, KRW
amt	String	15	R	Original transaction amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
shop	String	15		Shop Name(Required when Shop name is different from Merchant name)
buyer	String	64	R	Buyer Name (Required to use real name)
tel	String	32		Buyer's Contact Number
email	String	32	R	Buyer's email address(Buyer will receive email

				notification)
product	String	255	R	Product name of the ordered product
transid	String	24	R	Transaction id
rescode	String	6	R	Response Code. (Success:'0000')
resmsg	String	700	R	Response Message
authcode	String	8		Approval Code
cardco	String	32	R	Card type
resdt	Integer	14	R	YYYYMMDDHHMMSS (Reception date, Completion date)
cardholder	String	50		The English name of the cardholder entered by the payer
fgkey	String	64	R	Refer to 4.1 Fgkey
accesscountry	String	2		ISO 3166 payer access country (e.g. KR, JP, US…)
tokenID	String	20	R	Token ID issued at first payment
(The following is the same as the **2.1.3 response parameter**)				

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – When the card information is changed (expiration of the validity period, etc.), a new registration is made through "Performing a Sale (Token Creation)".

2.7 statusurl

Merchant needs to send Eximbay an acknowledgement message once they have received and updated the transaction result via statusurl successfully. In order to do that the following line message need to be sent back via the same connection or printed out on the same page:

- For success : *rescode*=0000&*resmsg*=Success
- For fail : *rescode*=(failure code)&*resmsg*=(failure message)

Important

If the acknowledgement message is not sent back to us, statusurl can be called multiple times. You can recognize duplicated calls if same *transid* is received. It's recommended that even if you receive duplicate call, you need to send back us the acknowledgement message.

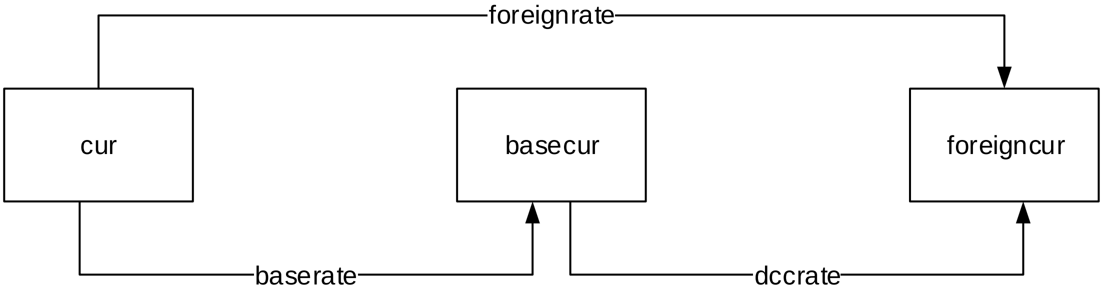
2.8 Currency Information

The below table shows how the cur, basecur, and foreigncur parameters of Online Payment(Refer to 2.1.3 response parameters) are returned.

Transaction Category	**cur**	**basecur**	**foreigncur**
Normal Transaction	KRW	KRW	KRW
DCC Transaction	KRW	KRW	Customer's billing currency
MCP Transaction	TWD	KRW	TWD
Alipay	KRW	USD	USD
PayPal	KRW	USD	USD
UnionPay	KRW	KRW	KRW

Note. 1 – basecur is the merchant's settlement currency and foreigncur is the customer's billing currency.
For normal transactions, the final charged currency of a customer can be different.

Note. 2 – In case of UnionPay, the currencies are returned as KRW-CNY-CNY in the test server. (UnionPay only supports CNY for testing)



2.9 Cash Payment Notice

This is used for **Japanese Convenience Store** payments(Refer to Appendix C (*paymethod*)).

Deposit Notice

Cash payment returns the payment result in 2 steps.

Step1. **Order Registration** : returnurl, statusurl

Step2. **Deposit Notice** : statusurl

Note

The statusurl for notifying the order registration in step1 can be set up to not send If the merchant doesn't want to receive it.

The statusurl of Step1 and Step2 can be distinguished by "*status*" parameter.

Step1. **Order Registration** : Registered

Step2. **Deposit Notice** : Sale

In addition, the parameter "*paymentURL*" is returned in the response of Order Registration in Step1 for **Japanese Convenience Store payment**. Customers use this URL to complete their order. Merchant may also send this URL to their customers.

Note

paymentURL

When a customer completes Order Registration process, Eximbay provides *paymentURL* by displaying the URL on the order registration completion page and by sending an email to the customer.

Order Registration Cancellation Notice

This interface is used to notify the merchant when Order Registration is cancelled before it is paid by a customer or if the given-limited time expires before the customer pays the order.

Note

You need to contact us if you need this interface.

Order Registration Cancellation is notified by statusurl. You can refer to the parameter "status".

Order Registration Cancellation Notice : Void

3 Common Management Interfaces

3.1 Crediting a Payment

This interface is used to refund transactions of **PAYMENT** or **PAYMENT_PA**, **AUTHORIZE_PA**. It usually takes 3-4 business days for acquiring bank to transfer and return money to the customer.

3.1.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/DirectProcessor.krp

3.1.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	**"REFUND"**
refundtype	String	1	R	**"F" : Fully, "P" : Partial**
ref	String	30	R	Original transaction's order number
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount. (e.g. 1000.50, 9.15) (Do not use comma ','. Must be more than 0)
refundamt	String	15	R	Amount to be refunded It cannot exceed the original transaction amount. If it is empty and refundtype is set to F, the full amount of the original transaction is refunded.
balance	String	10, 2		Available balance = Transaction amount – Sum total refunded amounts (based on cur) Used to verify refundable balance in Eximbay for exactness. Only used when the parameter is provided.
transid	String	24	R	Original Transaction ID
refundid	String	30	R	Unique refund request id generated from merchant. Each request should have a unique value.
reason	String	255	R	Reason of refunding the transaction
lang	String	2	R	Payment screen Language (Refer to Appendix B)
param1	String	255		Merchant-defined Parameter 1

param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "4.1 FGKEY" (For fgkey creation, refundid need to be added)
charset	String	32		Default UTF-8

Note. 1 – No particular UI is provided. The refund result is returned to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – Conditions for partial refund:

- if *refundtype* is not set with value **P**, full amount will be refunded.
- You can request partial refund any number of times if the total sum of the partial request amounts are not more than the original transaction amount.

3.1.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	**"REFUND"**
ref	String	30	R	Original transaction's order number
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount. (e.g. 1000.50, 9.15)
refundamt	String	15	R	Refunded amount
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
transid	String	24	R	Original Transaction ID
refundid	String	30	R	Unique refund request id generated from merchant. Each request should have a unique value.
rescode	String	6	R	Response Code. (Success : 0000, Refer to Chapter 4)
resmsg	String	700	R	Response Message
authcode	String	8		Approval Code (No value for online third-party payments)
resdt	Integer	14	R	YYYYMMDDHHMMSS (Response Date Time, Complete Date Time)
refundtransid	String	24		Refund transaction ID
fgkey	String	64	R	Refer to "4.1 FGKEY"
baseamt	String	15		Transaction Amount

			(Usually it is same with amt. When Reverse-DCC is used, either KRW or USD amount can be returned. It can also be used for the capture data if the merchant is direct acquirer contracted.)
baserate	String	14	Transaction Currency Exchange Rate (Exchange rate from cur to basecur. If cur and basecur are same, it is 1)
foreigncur	String	3	Customer's billing currency (When DCC is used, customer's selected currency is returned)
foreignamt	String	15	Customer's billing amount (When DCC is used, customer's accepted amount is returned)
foreignrate	String	14	Customer's billing currency exchange rate (Exchange rate from cur to foreigncur. If cur and foreigncur are same, it is 1)
dccrate	String	14	DCC Exchange Rate (Exchange rate from basecur to foreigncur. If basecur and foreigncur are same, it is 1)
balance	String	15	Available balance = Transaction amount – Refund amount (based on currency *cur*) Used to verify refundable balance for exactness.

Note. 1 – *ver ~ transid* parameters are returned same as they are sent in the request.

Note. 2 – *foreigncur ~ rateid* parameters are returned when the original transaction is DCC transaction.

3.2 Querying a Single Transaction

This interface is used to query a transaction detail and verify its result through API call from Merchant's Admin site. The transaction result can also be checked and verified by logging in Eximbay Admin site.

3.2.1 Request URL

- Test https://secureapi.test.eximbay.com/Gateway/DirectProcessor.krp
- Production https://secureapi.eximbay.com/Gateway/DirectProcessor.krp

3.2.2 Request Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	MerchantID provided by Eximbay
txntype	String	30	R	**"QUERY"**
keyfield	String	10	R	"**REF**" or "**TRANSID**"(default) For case of REF, the very last transaction(can be either payment or refund if they have same REF) is returned. For case of TRANSID, only its transaction is returned(TRANSID is unique for every transactions).
ref	String	30	R	Original transaction ref
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount (e.g. 1000.50, 9.15) Note : do not use comma ','
transid	String	24	C	Original transaction ID
lang	String	2	R	Payment screen Language (Refer to Appendix B)
returnurl	String	255		The page to be redirected
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
fgkey	String	64	R	Refer to "4.1 FGKEY"
charset	String	32		Default UTF-8

Note. 1 – No particular UI is provided. The query result is forwarded to returnurl if it is specified. If not specified, the response parameters are printed as name=value pairs.

Note. 2 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated.

Note. 3 – Do not send any other parameters except the ones specified in this request parameters

Note. 4 – if the parameter *keyfield* is not defined, "TRANSID" is set as default. And the parameter *transid* is required.

Note. 5 – The refund transaction cannot be queried by this interface, the refund status can be checked with the response parameter *balance* by querying its original transaction. If balance is 0, you can consider the transaction has been refunded fully.

3.2.3 Response Parameters

Field Name	Type	Length	Required	Description
ver	Integer	3	R	"**230**"
mid	String	10	R	Merchant ID provided by Eximbay
txntype	String	30	R	"**QUERY**"
ref	String	30	R	Original transaction ref
cur	String	3	R	Original transaction currency
amt	String	15	R	Original transaction amount (e.g. 1000.50, 9.15)
param1	String	255		Merchant-defined Parameter 1
param2	String	255		Merchant-defined Parameter 2
param3	String	255		Merchant-defined Parameter 3
status	String	10	R	Payment Completed : SALE (Payment confirmed) Authorization : AUTH (Payment not confirmed, confirmed after Capture) Order Registration : REGISTERED (Payment not confirmed, confirmed after Deposit Notice) No Transaction : NONE
transid	String	24	R	Original transaction ID
rescode	String	6	R	Response Code. (Success:'0000', Refer to Chapter 4)
resmsg	String	700	R	Response Message
authcode	String	8		Approval Code (Only provided for credit card payment)
resdt	Integer	14	R	YYYYMMDDHHMMSS (Transaction Date)
fgkey	String	64	R	Refer to "4.1 FGKEY"
cardno1	Integer	4		First 4 digit of card number
cardno4	Integer	4		Last 4 digit of card number
balance	String	15		Balance(Refundable Amount) = Transaction Amount – Sum Total Refunded Amount Refer to the next table for possible values.
paymethod	String	4		Payment Method Code (Refer to Appendix C)

Note. 1 – Following values are returned for *balance* depending on the status

Transaction category	status	balance	Description
AUTHORIZE AUTHORIZE_PA	AUTH	Same with amt	

AUTH_REVERSAL	AUTH	0	Reversal transaction
Order Registration (Japanese payment)	REGISTERED	Same with amt	
Order Cancelation (Japanese payment)	REGISTERED	0	Cancelled transaction
PAYMENT CAPTURE PAYMENT_PA Order Completion (Japanese payment)	SALE	Transaction Amount – Sum total refunded amounts	Available Refundable Amount

4 References

4.1 FGKEY

FGKEY is used to authenticate and validate parameters transferred between the merchant and Eximbay.

4.1.1 How it works

The SHA-256 hashing is a specific way of encrypting information to make it unreadable. You can generate your *fgkey* value by encrypting a string composed of specific parameters and send it in your request. If the fgkey value that you generated matches with the value we generated, your request is considered as valid, otherwise it is rejected. In the same way you can verify against the fgkey field provided in Eximbay response.

4.1.2 Setting up the SHA-256 hashing

A secret key is required to generate the fgkey and it is assigned to you by Exmbay. You must keep it confidential.

4.1.3 Hash via the merchant

You should use the following steps to generate your fgkey value:

A. You need to sort all request/response parameters in alphabetical order

B. Put the secretkey and A's data connecting together with "?" symbol in between

C. Encrypt B's result with SHA256 Hashing function

Example:

- secretkey

  F6DCE41DA82064F478B934663FD2A07E

- Eximbay link buffer

  -Request

  A = **sort**("ver=230&mid=1234567890&ref=A1234&cur=USD&amt=1.0......")

  -Response

  A = **sort**("ver=230&mid=1234567890&ref=A1234&cur=USD&amt=1.0&rescode=0000......")

- SHA-256 encryption

  B = "F6DCE41DA82064F478B934663FD2A07E?**A**"

  C = SHA256(B.getBytes("UTF-8"))

Note. 1 – You need to set character set as UTF-8 when converting to byte data for SHA256 encryption.

- Set the generated hash value to fgkey parameter

  fgkey=**C**

4.2 Response Code

When you receive the response parameter *rescode* with value 0000, the transaction is successful, otherwise it is a failure. For the failure reason, you may refer to the response parameter *resmsg*.

We cannot provide all the failure codes because each acquiring bank has different codes. But we may provide Eximbay system failure codes if merchants require.

4.3 AES 256 Encryption

The confidential parameters like cardno(credit card number) and expirydt(expiration date) are returned in the response as encrypted by AES 256 encryption algorithm. If you want to decrypt the parameters, you can do as following:

A. Encryption Algorithm : AES256

B. Operation mode : CBC

C. Padding : PKCS5Padding

D. IV Key : Eximbay_AES_256_

E. Round Key : Eximbay allotment (Note: Round Key is provided by Eximbay)

4.4 About app-to-app

Eximbay is based on the web. In the case of merchants who use EXIMBAY domestic payment in the merchant app, additional work related to calling the payment company's app is required.

4.4.1 IOS

1. Registration of URL Scheme

It is necessary to register an external URL scheme(LSApplicationQueriesSchemes) as follows for running a 3rd party app.

```
<key>LSApplicationQueriesSchemes</key>
<array>
```

```
            <string>line</string>       <!--LINEPAY-->
            <string>alipays</string>  <!—Alipay+-->
</array>
```

2. Setting for App Transport Security

ATS(App Transport Security) needs to be set up as follows to allow security restrictions for HTTP requests entering the web view.

```
<key>NSAppTransportSecurity</key>
<dict>
   <key>NSAllowsArbitraryLoadsInWebContent</key>
   <true/>
   <key>NSAllowsArbitraryLoads</key>
   <true/>
</dict>
```

4.4.2 ANDROID

Additional work related to calling the payment company's app is required on WebView, shouldOverrideUrlLoading.

```
@Override
public boolean shouldOverrideUrlLoading(WebView view, String url) {

            if (!URLUtil.isNetworkUrl(url) && !URLUtil.isJavaScriptUrl(url)) {
                        final Uri uri;
                        Intent intent = null;

                        try {
                                    uri = Uri.parse(url);
                                    intent = Intent.parseUri(url, Intent.URI_INTENT_SCHEME);

                        } catch (Exception e) {
                                    return false;
                        }

                        if ("intent".equals(uri.getScheme())) {
                                    try {
                                                Log.d("LOG", "intent startActivity");
                                                startActivity(intent);
                                                return true;
                                    } catch (ActivityNotFoundException e) {
```

```
                final String packageName = intent.getPackage();
                Log.d("LOG", "ActivityNotFoundException packageName :" +
packageName);
                if (!TextUtils.isEmpty(packageName)) {
                    startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=" + packageName)));
                    return true;
                  }
            }
        } else if ("alipays".equals(uri.getScheme())) { //Alipay+
            try {
                Log.d("LOG", "alipays startActivity");
                startActivity(new Intent(Intent.ACTION_VIEW, uri));
                return true;
            } catch (ActivityNotFoundException e) {
                Log.d("LOG", "ActivityNotFoundException alipays");
                startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=com.eg.android.AlipayGphone")));
                return true;

            } catch (Exception e) {
                return false;
            }
        } else if ("line".equals(uri.getScheme())) { //LINEPAY
            try {
                Log.d("LOG", "line startActivity");
                startActivity(new Intent(Intent.ACTION_VIEW, uri));
                return true;
            } catch (ActivityNotFoundException e) {
                Log.d("LOG", "ActivityNotFoundException line");
                startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=jp.naver.line.android”)));
                return true;
            } catch (Exception e) {
                return false;
            }
        } else {
            try {
                Log.d("LOG", "else startActivity");
                startActivity(new Intent(Intent.ACTION_VIEW, uri));
                return true;
            } catch (ActivityNotFoundException e) {
                final String packageName = intent.getPackage();
                Log.d("LOG",    "4091_else    startActivity    ActivityNotFoundException
packageName :" + packageName);
                if (!TextUtils.isEmpty(packageName)) {
                startActivity(new                              Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=" + packageName)));
                return true;
                }
            } catch (Exception e) {
                return false;
            }
        }
    }
    return false;
}
```

4.4.3 Additional settings by payment method

1. Alipay+

- Add Parameters on Payment Request Action

callfromapp = Y
callfromscheme = url scheme value of merchant app ex) exb.upay.sample

- Additional actions related to webview settings.

```
//Enable javascript
webSettings.setJavaScriptEnabled(true);
//Enable scaling
webSettings.setSupportZoom(true);
//Enable scaling controls (buttons)
webSettings.setBuiltInZoomControls(true);
//2 cache mode for WebView (web and WAP). Load no cache here.
webSettings.setCacheMode(WebSettings.LOAD_NO_CACHE);
//Allow JavaScript to open new windows (false by default).
webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
//Allow JavaScript to load the local cache.
webSettings.setDomStorageEnabled(true);
//WAP cache size (No need to manually set)
//webSettings.setAppCacheMaxSize(1024 * 1024 * 8);
//WAP cache path
String absolutePath = getApplicationContext().getCacheDir().getAbsolutePath();
///WAP cache size
webSettings.setAppCachePath(absolutePath);
```

```
//Whether allow WebView to access files (true by default)
webSettings.setAllowFileAccess(true);
//Enable to save WAP cache
webSettings.setAppCacheEnabled(true);
//When using overview mode, if the the page width exceeds WebView dispaly, scale the page to adapt to the WebView (false by default)
webSettings.setLoadWithOverviewMode(true);
// support for the viewport HTML meta tag
webSettings.setUseWideViewPort(true);
```

Appendix A Supported Currencies

A.1 Request Currencies (cur)

Currency Code	Currency ID(ISO4217)	Currency Name	Minor Unit
KRW	410	Korea Won	0
USD	840	US Dollar	2
EUR	978	Euro	2
GBP	826	Pounds Sterling	2
JPY	392	Japan Yen	0
THB	764	Thailand Baht	2
SGD	702	Singapore Dollar	2
RUB	643	Russian Ruble	2
HKD	344	Hong Kong Dollars	2
CAD	124	Canadian Dollars	2
AUD	036	Australian Dollars	2

A.2 DCC Currencies (foreigncur)

Currency Code	Currency ID(ISO4217)	Currency Name	Minor Unit
USD	840	US Dollar	2
SGD	702	Singapore Dollar	2
EUR	978	Euro	2
GBP	826	Pounds Sterling	2
JPY	392	Japan Yen	0
CNY	156	China Yuan Renminbi	2
THB	764	Thailand Baht	2
RUB	643	Russian Ruble	2
HKD	344	Hong Kong Dollars	2
CAD	124	Canadian Dollars	2
AUD	036	Australian Dollars	2
TWD	901	New Taiwan Dollar	2

MYR	458	Malaysian Ringgit	2
VND	704	Vetnam Dong	0
PHP	608	Philippine Peso	2
MNT	496	Tugrik	2
NZD	554	New Zealand Dollar	2
AED	784	UAE Dirham	2
MOP	446	Pataca	2
BRL	986	Brazilian Real	2
KZT	398	Tenge	2
NOK	578	Norwegian Krone	2
SAR	682	Saudi Riyal	2
TRY	949	Turkey Lirasi	2

Appendix B Supported Languages

Code	Language
KR	Korean
EN	English
CN	Chinese (Simplified Chinese Characters)
JP	Japanese
RU	Russian
TH	Thai
TW	Chinese (Traditional Chinese Characters)
VN	Vietnam

Appendix C Payment Methods (paymethod)

paymethod	Payment Method Name
P000	CreditCard
P101	VISA
P102	MasterCard
P103	AMEX
P104	JCB
P105	CUP (UnionPay 2D)
P106	Diners
P107	Discover
P108	Mir
P001	PayPal
P002	CUP (UnionPay)
P003	Alipay Plus
P174	Alipay Plus(ALIPAY_CN)
P141	WeChat (PC)
P142	WeChat (Mobile)
P143	WeChat (OA)
P006	Japanese Convenience Store, Internet Banking payment
P007	Razer Merchant Services
P171	Razer Merchant Services (Malaysia)
P172	Razer Merchant Services (Vietnam)
P173	Razer Merchant Services (Thailand)
P011	Yandex
PG01	2C2P
P185	grabPay(SGD)
P186	linePay
P189	grabPay(MYR)
P190	grabPay(PHP)

P175	Alipay Plus(TRUEMONEY)
P176	Alipay Plus(DANA)
P177	Alipay Plus(ALIPAY_HK)
P178	Alipay Plus(TNG)
P179	Alipay Plus(GCASH)
P194	Econtext-Linepay

Note. 1 P175 ~ P179 is not available when requesting payment and is only set to the response value of the detailed wallet selected by the customer when responding. (Integrated Payment Conversion)

Appendix D State, Province and Territory Codes

D.1 For the United States and Canada

United States Postal Service(USPS) Abbreviations

State	Code	State	Code
Alabama	AL	Montana	MT
Alaska	AK	Nebraska	NE
American Samoa	AS	Nevada	NV
Arizona	AZ	New Hampshire	NH
Arkansas	AR	New Jersey	NJ
California	CA	New Mexico	NM
Colorado	CO	New York	NY
Connecticut	CT	North Carolina	NC
Delaware	DE	North Dakota	ND
District of Columbia	DC	Northern Mariana Islands	MP
Federated States of Micronesia	FM	Ohio	OH
Florida	FL	Oklahoma	OK
Georgia	GA	Oregon	OR
Guam	GU	Palau	PW
Hawaii	HI	Pennsylvania	PA
Idaho	ID	Puerto Rico	PR
Illinois	IL	Rhode Island	RI
Indiana	IN	South Carolina	SC
Iowa	IA	South Dakota	SD
Kansas	KS	Tennessee	TN
Kentucky	KY	Texas	TX
Louisiana	LA	Utah	UT
Maine	ME	Vermont	VT
Marshall Islands	MH	Virgin Islands	VI
Maryland	MD	Virginia	VA
Massachusetts	MA	Washington	WA
Michigan	MI	West Virginia	WV
Minnesota	MN	Wisconsin	WI
Mississippi	MS	Wyoming	WY
Missouri	MO		

U.S Military Address Format

Line	Description	Examples
Address	Street address	1775 John Paul Jones BLVD

	Post office box number Postal service center number Unit and/or box number	PO Box 405 4PSC 467 Box 291 Unit 30001 Unit 62001 Box 426
City	APO (Army or Air Force Post Office) or FPO (Fleet Post Office for the Navy, Marine Corps, or Coast Guard	FPO AE 09501-4665 FPO AA 34093-2329 FPO AP 96349-1100
State	AE if customer is in Africa, Canada, Europe, or the Middle East AP if customer is in Pacific AA if customer is elsewhere in the Americas	
ZIP	Postal code of the customer's location	
Country	Country of the customer's location or US if customer is not currently assigned to another country (FPO)	

Canadian Province or Territory Abbreviations

Province or Territory(English)	Description
Alberta	AB
British Columbia	BC
Manitoba	MB
New Brunswick	NB
Newfoundland and Labrador	NL
Northwest Territories	NT
Nova Scotia	NS
Nunavut	NU
Ontario	ON
Prince Edward Island	PE
Quebec	QC
Saskatchewan	SK
Yukon	YT

D.2 For other countries (PayPal only)

For PayPal payment method, the country and the state code of shipping address should be set as following.

(1) shipTo_country

Countries stated in the below link can only be sent as a request parameter.

https://developer.paypal.com/docs/api/reference/country-codes/

(2) shipTo_state

1. For countries(**Argentina, Brazil, Canada, China, Indonesia, India, Japan, Mexico, Thailand, USA**) the ***shipTo_state*** parameter is required, and you can find and use the state codes in the below link.
   https://developer.paypal.com/docs/api/reference/state-codes/

2. If any state code is not available for above listed countries, then you may need to develop your UI to **directly input the state code from a customer**.
3. For other than above countries, please use ***shipTo_city*** 's value into ***shipTo_state*** parameter.

For the reference, if you need to distinguish *shipTo_state* parameter's value for each different payment methods, it is advised to use along with *paymethod* request parameter.

Appendix E PayPal Airline Parameters

For airlines using PayPal, you will need to send the additional request parameters below.

E.1 Transaction Type

If txntype is *PAYMENT*, *AUTHORIZE*, or *CAPTURE* at the merchant using PayPal, it is used in addition to the request parameter.

E.2 Request Parameters

Field Name	Type	Length	Required	Description
paypal_ticket_number	String	16	R	Ticket number/id as issued by the airline.
paypal_ticket_issue_date	String	10	R	Date the airline ticket was issued. (YYYY-MM-DD)
paypal_ticket_issuing_carrier_code	String	2	R	Carrier code of the ticket issuer.
paypal_legs_#_flight_number	String	5	R	The flight number of the current leg.
paypal_legs_#_carrier_code	String	2	R	The IATA two-letter accounting code that identifies the carrier.
paypal_legs_#_service_class	String	1	R	The service class to which the airline ticket applies.
paypal_legs_#_departure_date	String	10	R	The date of departure.(YYYY-MM-DD)
paypal_legs_#_departure_time	String	5	R	The local time when the flight departs from the airport. (HH-MM)
paypal_legs_#_departure_airport	String	4	R	The airport code, as defined by IATA.
paypal_legs_#_arrival_airport	String	4	R	The airport code, as defined by IATA.
paypal_legs_#_stopover_code	String	1	R	0 : 스탑오버 가능, 1 : 스탑오버 불가 The one-letter code that indicates whether the passenger is entitled to make a stopover. 0 : stopover possible 1 : stopover impposible
paypal_legs_#_fare_basis_code	String	15	R	The code used by airline to identify a fare type and enable airline staff and travel agents to find the rules for this ticket.
paypal_passenger_name	String	300	R	탑승자 명 The name of the passenger.

Note. 1 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # starts from 0 and is increased by 1 up to the number of additional parameters. (e.g. *paypal_legs_0_flight_number*, *paypal_legs_1_flight_number*, ……)

Note 3 – "*paypal_legs_#*" is based on #, refers to the route information of the aircraft, and the value of # increases when there is a stopover.

Appendix F PayPal ISPP Parameters

What is PayPal ISPP? As a seller protection policy for digital products, the seller can receive compensation for all damages from PayPal in the event of a risk transaction. PayPal ISPP service can only be provided to merchants who have passed Eximbay & PayPal's screening, so please make sure to check the preliminary screening request and results before applying for it. If you are a merchant who uses the service, you must send all additional request parameters below. If it is impossible to send even one of the request parameters, please contact Eximbay in advance.

F.1 Transaction Type

If txntype is **PAYMENT**, **AUTHORIZE** at the merchant using Paypal, adds to the request parameter to use.

F.2 Request Parameters

Field Name	Type	Length	Required	Description
sender_account_id	String	128	R	Unique identifier of the buyer account on the partner/merchant platform
sender_first_name	String	80	R	First name registered with the buyer's partner/merchant account
sender_last_name	String	80	R	Last name registered with the buyer's partner/merchant account
sender_email	String	128	R	Email address registered with the buyer's partner/merchant account
sender_phone	String	30	R	Phone number(national notation) registered with the buyer's partner/merchant account
sender_country_code	String	15	R	Country code registered with the buyer's partner/merchant account
sender_create_date	date		R	Date of creation of the buyer's account on the partner/merchant platform (YYYY-MM-DD HH:MM:SS)

Appendix G Yandex Airline Parameters

For airlines using Yandex, you will need to send the additional request parameters below.

G.1 Transaction Type

If txntype is PAYMENT, AUTHORIZE, or CAPTURE at the merchant using Yandex, it is used in addition to the request parameter.

G.2 Request Parameters

Field Name	Type	Length	Required	Description
yandex_booking_reference	String	20	R	Booking reference number, required for creating a payment.
yandex_ticket_number	String	16		Unique ticket number, required for capturing the payment.
yandex_passengers_#_first_name	String	15	R	Passenger's first name.
yandex_passengers_#_last_name	String	15	R	Passenger's last name.
yandex_legs_#_departure_airport	String	3	R	Code of the departure airport code according to IATA, for example, LED.
yandex_legs_#_destination_airport	String	3	R	Code of the arrival airport code according to IATA, for example, LED.
yandex_legs_#_departure_date	String	10	R	Departure date in the YYYY-MM-DD ISO 8601:2004 format.

Note. 1 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # is an integer greater than to 0, incremented by an additional parameter. (eg. yandex_passengers_0_first_name)

Note. 3 – "yandex_passengers_#_first_name", "yandex_passengers_#_last_name" is a pair of parameters for one passenger information based on #, and supports up to 4 people.

Note. 4 – "yandex_legs_#_departure_airport", "yandex_legs_#_destination_airport" and "yandex_legs_#_departure_date" refer to the route information of an aircraft based on #, and the value of # increases when there is a waypoint.

Appendix H DecisionManger Airline parameters

For airlines using DM service, you will need to send the the additional request parameters below when you request credit card payment.

H.1 Transaction Type

If txntype is PAYMENT, AUTHORIZE, or CAPTURE at the merchant using DM sevice, it is used in addition to the request parameter.

H.2 Request Parameters

Field Name	Type	Length	Required	Description
item_#_passengerFirstName	String	60	R	Passenger's first name.
item_#_passengerLastName	String	60	R	Passenger's last name.
item_#_passengerID	String	32	R	ID of the passenger to whom the ticket was issued. For example, you can use this field for the frequent flyer number.
item_#_passengerStatus	String	32	R	Your company's passenger classification, such as with a frequent flyer program. In this case, you might use values such as standard, gold, or platinum.
item_#_passengerType	String	32	R	Passenger classification associated with the price of the ticket. You can use one of the following values: ADT: Adult CNN: Child INF: Infant YTH: Youth STU: Student SCR: Senior Citizen MIL: Military
item_#_passengerEmail	String	255	R	Passenger's email address, including the full domain name, such as jdoe@example.com.
item_#_passengerPhone	String	15	R	Passenger's phone number. If the order is from outside the U.S., recommends that you include the country code.
decisionManager_travelData_completeRoute	String	255	R	Concatenation of individual travel

				legs in the format ORIG1-DEST1[:ORIG2-DEST2...:ORIGn-DESTn], for example : SFO-JFK:JFK-LHR:LHR-CDG. For a complete list of airport codes, see IATA's City Code Directory.
decisionManager_travelData_departureDateTime	String	25	R	Departure date and time of the first leg of the trip. Use one of the following formats: yyyy-MM-dd HH:mm z yyyy-MM-dd hh:mm a z yyyy-MM-dd hh:mma z HH = hour in 24-hour format hh = hour in 12-hour format a = am or pm (case insensitive) z = time zone of the departing flight, for example: If the airline is based in city A, but the flight departs from city B, z is the time zone of city B at the time of departure. Important For travel information, use GMT instead of UTC, or use the local time zone. Examples 2011-03-20 11:30 PM PDT 2011-03-20 11:30pm GMT 2011-03-20 11:30pm GMT-05:00 Eastern Standard Time: GMT-05:00 or EST
decisionManager_travelData_journeyType	String	32	R	Type of trave, ex) one way, round trip
decisionManager_travelData_leg_#_destination	String	3	R	Airport code for the destination of the leg of the trip designated by the pound (#) symbol in the field name. This code is usually three digits long, for example: SFO = San Francisco. Do not use the colon (:) or the dash (-). For a complete list of airport codes, see IATA's City Code Directory.
decisionManager_travelData_leg_#_origin	String	3	R	Airport code for the origin of the leg of the trip designated by the pound (#) symbol in the field name. This code is usually three digits long, for example: SFO = San Francisco. Do not use the colon (:) or the dash (-). For a complete list of airport codes, see IATA's City Code Directory.
decisionManager_third_party_booking	String	3	R	The buyer is on the boarding list

				ex) Yes, No
decisionManager_corporate_booking	String	3	R	The buyer is 3rd party corporate. ex) Yes, No
decisionManager_hours_until_departure	String	5	R	Hours between purchasing date and departure date ex) Purchasing date - 2015.04.21 15:00 Date of departure - 2015.04.21 15:00 Value is given as 24.
decisionManager_age_of_profile	String	5	R	The date purchaser's account was created. (if non-member or account was created on the day of = 0)
decisionManager_number_of_passengers	String	3	R	Number of passengers
decisionManager_fare_class	String	5	R	Flight ticket type Ex) Promo, Flexi, Eco
decisionManager_baggage_purchased	String	3	R	Baggage availability ex) Yes, No
decisionManager_payer_pessenger	String	3	R	-The name of cardholder and the name of passengers are equal ex) Yes, No
decisionManager_agent_code	String	8	R	Travel agency code (IATA code)
decisionManager_agent_name	String	25	R	Name of travel agency

Note. 1 – The parameters are case sensitive. It is recommended to send parameters with upper/lower case letters as they are stated in this document.

Note. 2 – # is a zero-based integer, *passengerFirstName ~ PassengerPhone* is repeated as many as the number of passengers.

Note. 3 – *The destination and the origin* of *decisionManager_travelData_leg* are repeated as many as the number of airports.