VeriTrans4G Development Guide (MDK Module Method)

Ver. 1.0.0 beta (April 2017~)

Table of contents

Chapter 1 Getting Started

1-1 Contents of This Guide

This guide is intended for developers integrating their website with VeriTrans4G MDK(Merchant Development Kit Version2.x) offered by VeriTrans Inc.It contains information about samples, an explanation about libraries and message formats etc. which can be used as reference during integration.

1.1.1 About MDK (Merchant Development Kit)

MDK is a generic term of the module integrated into the online store website. We provide MDK for multiple languages such as Java, PHP, etc. Please download the MDK that is compatible with your server environment from the VeriTrans download site. As per the contract, you can start accepting payments by integrating and customizing the MDK to the online store website. Multiple payment services can be used simultaneously.

✓ **About Operating environment, please refer to the supplement "MDK installation guide".**

1-2 Copyright and Contact Details

[Contact Details] Technical Support, VeriTrans Inc. E-mail: tech-support@veritrans.jp

1-3 Revision History

2017/04: Ver1.0.0 Release

Chapter 2 Service Overview

2-1 Available Payment Method

Available payment methods are as follows:

Table 2-1-1 List of Payment Services

Payment Service Name	Description
Credit Card	This payment service enables making payments with credit cards issued by various brands and card companies.
3-D Secure	This is a 3-D Secure authentication service. This service can be used as a standalone service or along with credit card payments.
Convenience store	This payment service enables making payments in convenience stores such as 7-Eleven, LAWSON, FamilyMart etc.
E-money	This payment service enables making payments with E-Money (Suica, Edy, WAON, nanaco) on Online shop.
Bank	This payment service enables making payments through ATM's and Internet banking of financial institutes.
UnionPay Online Payments (UPOP)	This payment service enables making payments using UnionPay cards.
PayPal	This payment service enables making payments through PayPal.
Eikyu Fumetsu point	This payment service enambles making payments using Eikyu Fumetsu wallet (point) along with credit card payment.
Alipay	This payment service enables making payments through Alipay account. Online as well as physical retail store (through barcode) payment is also possible.
Carrier	This service enables making payments along with the communication charges of the telecommunication carriers. Payments can be done through 4 telecommunication carriers (DOCOMO Keitai Barai, au Kantan Kessai, SoftBank Matomete Shiharai (B), SoftBank Matomete Shiharai (A), S! Matomete Shiharai, Flets Matomete Shiharai).
Shopping credit	This service enables making payment through shopping credit. It facilitates making payments through Orico web credit.
Rakuten Pay	This service enables making payments using Rakuten ID. It facilitates making payments with Rakuten Super Point.
RECRUIT Kantan Shiharai	This service enables making payments using Recruit ID. It facilitates making payments with Recruit points or coupons.
LINE Pay	This service enables making payments using LINE user ID. It facilitates making payments with credit cards registered with LINE Pay or by using balances from LINE Pay accounts. Payment at physical retail store (through barcode) payment is also possible
MasterPass	This service enables making payments with credit cards registered with MasterPass digital wallet.

✓ **For the System Flow of Each Payment, Please Refer to the "Development Guide Supplement Outline System Flow Chart".**

2-2 About Subscription Service

By using the member management function, it is possible to manage by relating members (consumers) of online sites managed by merchant etc., information of the credit card used at the time of payment, and transaction result etc. By combining this "member management" and "continuous billing" function, routine payment can be made for the payment of monthly dues of service or payment for subscription.

The general term for these services is called "Subscription service".

By integrating subscription service, you can use the following functions:

- Member management (ID management)

  Register members' (consumer) ID, credit card information and the subscription schedule.

  It is possible to change (update), delete the registered information.

- Payment with member ID

  Payment request can be made with member ID which in turn will be processed with credit card information associated to that member ID.

  For payment other than a card, payment result and member ID can be associated by specifying the member ID.

- Continuous billing

  Payment is executed regularly for the payment of subscription fee and monthly fee of services.

- Credit card account updater

  Validity of the credit card number and expiry date is confirmed and new information is updated.

✓ **For details, please refer to supplement "Subscription service usage guide".**

2-3 About Fraud Detection Service

We support fraud countermeasures by stores by offering a "completely integrated" service having integrated fraud detection function for conventional credit card payments. According to the needs of the store, you can select the fraud detection tool from offered multiple options.

- Credit card payment request to the credit card company and fraud detection by the fraud detection engine run in parallel and the credit card authorization result (success / failure) and the fraud detection result (acceprt / challenge / deny) are returned together to the store system.
- The result from fraud detection tool can be confirmed on the transaction information page of the admin tool (MAP).

✓ **Separate application is required for using fraud detection service. For details, refer to the supplement "Fraud detection service usage guide" and contact our sales representative.**

Chapter 3 Integration of MDK

3-1 Procedure for integrating MDK

This chapter explains the procedure for integrating MDK.

No.	Operation	Operational details	Remarks
1	Getting MDK	Download the MDK, which is compatible with the merchant's integration environment and its corresponding document from the VeriTrans download site. https://www.veritrans.co.jp/trial/login/	
2	Getting and generating external resources	Refer to MDK integration environment information and download/get the software and libraries required for integration	Please refer to "MDK installation guide".
3	Configuring external resources and setting up the environment	Configure and set up the downloaded software and libraries.	
4	MDK Installation	Unzip and install the downloaded file. Get the required authentication information from the Admin portal (MAP) and set required information in the MDK configuration file	
5	System test (test environment)	Conduct VeriTrans loop-back testing. Set dummy mode to "ON", verify MDK operation and verify if communication with VeriTrans is possible. VeriTrans can issue a dedicated account for testing.	Please refer to "Integration Test Guide".
6	System test (live environment)	Conduct a communication test with each payment center. Set dummy mode to "OFF" and verify if payment at live run is possible. * Please note that a transaction fee will be charged for testing on live environment.	

* For details about MDK's supported environment for individual programming language, **please refer to** our company's home page or **the "MDK Installation Guide".**

* We may not be able to support you in some cases, for example, if you integrate with an environment that is not compatible with our pre-verified environment (such as OS).

Also, please note that as a rule, we do not provide support for the following cases:

- Operational verification on your own customized OS (kernel)
- Inquiries about your own customized programs
- Server environment configuration that is not relevant with MDK.

3-2 MDK Commands

Table 3-2-1 MDK commands that can be used

Command name		Authorize	ReAuthorize	Capture	Cancel	Refund	Others	Search
Payment Service name	Credit card	○	○	○	○			○
		Authorize	Re-order	Capture	Cancel			Search
	3-D Secure	○	○	*Capture, cancellation of card payment will be performed.				○
		3-D Secure	Re-order					Search
	Convenience store	○		△	○			○
		Authorize		(Payment notification)	Cancel			Search
	E- money	○	○	△	○	○	**Remove**	○
		Authorize	Re-payment (Only nanaco)	(Payment Notification)	Cancel	Refund	Card information removal (only nanako)	Search
	Bank	○		△				○
		Authorize		(Payment Notification)				Search
	UnionPay Online Payments (UPOP)	○		○	○	○		○
		Authorize		Capture	Cancel	Refund		Search
	PayPal	○		○	○	○		○
		Authorize		Capture	Cancel	Refund		Search
	Eikyu Fumetsu point	○		○	○			○
		Page request		Payment request	Cancel			Search
	Alipay	○				○		○
		Authorize with Capture				Refund/ Refund request		Search
	Carrier	○		○	○		**Terminate**	○
		Authorize or Subscription request		Capture	Cancel		Subscription Termination	Search
	Shopping credit	○		△				○
		Page display		(Application review result notification)				Search
	Rakuten Pay	○		○	○			○
		Authorize		Capture	Cancel			Search
	RECRUIT Kantan Shiharai	○		○	○		**ExpandAuth**	○
		Authorize		Capture	Cancel		Authorization period extension	Search
	LINE Pay	○		○	○			○
		Authorize		Capture	Cancel			Search

Command name		Authorize	ReAuthorize	Capture	Cancel	Refund	Others	Search
	MasterPass	○		○	○		**Login**	○
		Authorize		Capture	Cancel		Authentication request	Search
Description		This command will request authorize or capture.	This command will request re-authorization, or payment authorization, based on past transaction information.	This command will request capture confirmation of authorized transactions.	This command performs cancellation for authorization, capture.	This command will request refund for transaction already captured.	Please refer to “interface details” of each service.	This command will perform Transaction Information search.

✓ **Usage condition for command may differ for each service; hence please refer to the "Interface details" of each service.**

◆ Authorize command

Normally, the first command to be executed for payment request is Authorize. Incase Authorize fails, it is impossible to complete the payment, so inform the consumers that payment has failed and resend the payment request.

- In MasterPass payment, the first command to be executed is Login.

◆ Capture command

When authorization is successful with authorize command, it is necessary to capture the funds by capture command. If capture is not executed, the consumer will not be charged. However, this is not applicable in the following cases:

- If you specify "with capture" option at the time of requesting Authorization, capture is executed at the time when Authorize command succeeds. (Credit card payment etc.)
- In Table 3-2-1, for the payment services (convenience store, electronic money etc) where Δ is marked shown in Capture command column, funds will be treated as Captured at the timing when the consumer deposits money and payment service operator confirms the payment. In this case, notification will be sent from the payment server to the merchant system, indicating confirmation of capture.
- In cases like subscription service of carrier payment etc., where capture is executed automatically by our company or payment service operator, there is no need to send capture request.

◆ Cancel / Refund command

In case of payment cancel or payment refund, for services that use both Cancel and Refund command, refer specifications of each service as it is necessary to use Cancel and Refund command differently.

In the service which uses only the cancel command, there is no need to be aware whether it is cancel or refund as depending on the status of the transaction at the time of the command request, the payment service operator or payment server will execute the required command.

◆ ReAuthorize command

In credit card payments (which includes 3-D Secure), it is possible to get authorization (or Authorization with capture) using the old credit card transaction information. At the time of request, specify transaction ID (original transaction ID) of the old transaction as a parameter.

Reauthorization is possible as long as order information is stored at VeriTrans. About the storage period, 400 days is taken as a standard from the day on which the transaction status was last changed.

Possible reauthorization periods are shown below.

Transaction status	Possible reauthorization period * Storage period of transaction history
Capture	Capture execution day + 400 days
Cancel (return)	Cancel (return) execution day + 400 days
Authorize (Authorization expiration)	Authorization date+ Authorization expiry period (60 days) + 400 days

Since the transaction history is deleted once it exceeds the storage period, as the original transaction ID is in reauthorization, you should specify the transaction ID which was used in the most recent reauthorization

- Reauthorize command in nanaco payment differs from reauthorization of credit card payment because it is a recovery process for transactions whose processing is unfinished.

◆ Search command

It enables you to search the status of the transaction.

(Important)

For search, various conditions can be specified for each payment service, but make sure you specify "transaction ID" without fail in conditions when you execute search command. In case search is performed without including transaction ID in conditions, response delays or unexpected results may be returned depending on the result data size. If you want to perform search with multiple conditions and without including transaction ID, please consult with technical support beforehand.

Chapter 4 Interface Details

4-1 Payment Interface Details

4.1.1 Interface Details of Each Service

For interface details, refer to the supplement document for each payment service.

A list of file names is shown in below table.

Table 4-1-1 Interface details file names list

Payment Service Name		Interface Details
Credit Card	Card	VeriTrans4G_Development_Guide_IF02_Card_XXX.pdf
	3-D Secure	VeriTrans4G_Development_Guide_IF03_MPI_XXX.pdf
	MDK Token*1	VeriTrans4G-MDKToken_Development_Guide_XXX.docx
Convenience store		VeriTrans4G_Development_Guide_IF04_CVS_XXX.pdf
Electronic money		VeriTrans4G_Development_Guide_IF05_EM_XXX.pdf
Bank		VeriTrans4G_Development_Guide_IF06_Bank_XXX.pdf
UnionPay Online Payment (UPOP)		VeriTrans4G_Development_Guide_IF07_UPOP_XXX.pdf
PayPal		VeriTrans4G_Development_Guide_IF08_Paypal_XXX.pdf
Eikyu Fumetsu point		VeriTrans4G_Development_Guide_IF09_Saison_XXX.pdf
Alipay		VeriTrans4G_Development_Guide_IF10_Alipay_XXX.pdf
Carrier		VeriTrans4G_Development_Guide_IF11_Carrier_XXX.pdf
Shopping Credit (Orico)		VeriTrans4G_Development_Guide_IF12_Oricosc_XXX.pdf
Rakuten pay		VeriTrans4G_Development_Guide_ex_rakuten_XXX.pdf
RECRUIT Kantan Shiharai		VeriTrans4G_Development_Guide_ex_recruit_XXX.pdf
LINEPay		VeriTrans4G_Development_Guide_ex_line_XXX.pdf
MasterPass		VeriTrans4G_Development_Guide_ex_master_XXX.pdf
Search function		VeriTrans4G_Development_Guide_IF13_Search_XXX.pdf
Subscription service	User guide	VeriTrans4G_PayNowID_Overview_XXX.pdf
	I/F details	VeriTrans4G_Development_Guide_IF01_PayNowID_XXX.pdf
	I/F details (Member management)	VeriTrans4G_Development_Guide_IF01-2_PayNowID (Member Management Function).xlsx
	Operation guide	VeriTrans4G_PayNowID_Operation_Guide_XXX.pdf

Note) “XXX” in the file name indicates version number.

*1 Conforming to "Implementation plan for strengthening security measures in credit card transaction" released by "Japan Consumer Credit Association", Ministry of Economy, Trade and Industry, it is necessary to use tokenized card information (MDK Token) when requesting payment to the payment server, in order to comply with no electronic storage or non-transmission of cardholder data.

4.1.2 Common Parameters

Request message				
Field name	**Parameter name**	**Formatting/ Limitation**	**Description**	**Setting**
memo1	Transaction memo 1	String up to 100 characters	This specifies notes related to the transaction.	**Optional**
freeKey	Key information	Single byte alphanumeric characters up to 256 digits.	This sets optional key information being used at the merchant system.	**Optional**

It is a parameter that can be set for all payment services. The following usage is assumed for each one.

■About memo1 (Transaction memo 1)

This is the memo information related to the transaction.

memo1 (transaction memo1) is not masked and input value is output to log file as it is. For this reason, we request you not to put sensitive information such as card number, personal information and confidential information in memo1 (transaction memo1).

■About freeKey (Key)

This is the key related to the transaction. You can set the key information for linking our transaction. You can use this field to link the transaction information managed by us with the ID separately managed by you. (Example: Transaction ID of your system linked to transaction information on our side.)

reeKey (key information) is not masked and input value is output to log file as it is. For this reason, we request you not to put sensitive information such as card number, personal information and confidential information in freeKey (key information).

■About overwrite of memo1 (Transaction memo 1), freeKey (Key)

The value of transaction memo 1, key is overwritten with the value specified in the latest transaction.

Only the value of transaction memo 1 and key specified in the latest transaction is saved and the value of transaction memo 1 and key specified in the previous transaction is not saved.

Example 1: When specified during authorization/ payment request (Authorize), with capture as well.

⇒ Value of transaction memo 1, key specified at the time of capture will be stored, whereas value set at the time of authorization/ payment request (Authorize) is not stored.

Example 2: When specified at the time of capture and not specified at authorization/ payment request (Authorize)

⇒ Value of transaction memo 1, key specified at the time of capture will be stored.

Example 3: When specified at the time of authorization/ payment request (Authorize) and not specified at Capture.

⇒ Value of transaction memo 1, key specified at the time of capture / payment application (Authorize) will be stored.

Example 4: When specified at the time of capture, and after that cancellation is executed

⇒ Value of transaction memo 1, key specified at the time of cancellation is stored where as value set at the time of

capture, is not stored.

4-2 Payment Result Notification

In the service where time lag occurs from authorization to payment completion, like convenience store payments or bank payments, after confirming the result, the payment result notification is sent from the payment server to the merchant system.

4.2.1 Message Specification

Specification of payment result notification is as follows.

Protocol		HTTP1.0 or HTTP1.1
Method		POST
Request header	**content-hmac**	HMAC value for tempering check Set value: h= {Algorithm name}; s={CCID}; v= {HMAC value} • ”HmacSHA256” is set as algorithm name. • To CCID, “Merchant CCID” is set. • Key at the time of calculating HMAC is, “Merchant secret key”.
	content-length	BODY length
	Content-Type	application/x-www-form-urlencoded
Port number		443 or 80
Character code		UTF-8
MessageBody		Format is as follows. Key 1= [Value 1]&Key 2=[Value 2]&Key 3=[Value 3]&… (Example) Incoming payment notification message of convenience store payment. numberOfNotify=2&pushTime=20170330172650&pushId=00000237& orderId0000=ORDER001&cvsType0000=sej&amount0000=100&receiptNo0000=0000123& receiptDate0000=20170330172505&rcvAmount0000=100&dummy0000=0& orderId0001=ORDER002&cvsType0001=sej&amount0001=100&receiptNo0001=0000123& receiptDate0001=20170330172505&rcvAmount0001=100&dummy0001=0 ✓ Please refer to "Interface details" of each service for details of message components.

* Specifications not specified in this document shall conform to RFC 1945 and RFC 2616.

■ About the result of result notification reception processing

When the merchant's system returns the HTTP status code "200" in response to the result notification from the payment server, the payment server treats that as the merchant system has successfully received the notification. When a code other than the HTTP status code "200" is returned, it treats it as that the notification has failed and keeps on sending notifications at regular intervals

- **Notification will be resent up to a maximum of 8 hours at intervals of 1 to 10 minutes.**
- **If it fails for the specified number of times, notification will be stopped. In order to check the status of transactions that could not be received, please use the management tool (MAP) or Search function.**

■ About pushId (Identification ID)

The pushId (identification ID) is unique within one payment service, but when using multiple payment services, the same ID may be set across services, so please be careful not to treat it as a unique key.

4.2.2 Result Notification List

Payment Service Name	Authorize	Capture and Payment	Cancel	Refund	Others
Credit card	-	-	-	-	-
3-D Secure	◎Payment result notification	-	-	-	-
Convenience store	-	○Payment received notification	-	-	-
E-Money	-	○Payment received notification	-	○Refund notification	-
Bank	-	○Payment received notification	-	-	-
UnionPay Online Payment(UPOP)	◎Payment result notification	◎Payment result notification	◎Payment result notification	◎Payment result notification	-
PayPal	-	○Payment received notification	-	○Refund notification	-
Eikyu Fumetsu point	-	-	-	-	-
Alipay payment					
Online	◎Payment result notification	-	-	◎Authorization result notification ◎Payment result notification	-
Bar code (Store scan type)	-	-	-	-	-
Bar code (Consumer scan type)	◎Payment result notification	-	-	-	-
Carrier	○Authorization completion notification	-	-	-	○ Subscription completion notification ◎ Subscription process result notification ○ Subscription deletion notification
Shopping Credit	-	-	-	-	◎Verification result notification
Rakuten pay	○Authorization completion notification	◎Capture result notification	◎Cancellation result notification	-	-
RECRUIT Kantan Shiharai	○Authorization completion notification	-	-	-	-
LINEPay					
Method via browser	○Authorization completion notification	-	-	-	-

Payment Service Name	Authorize	Capture and Payment	Cancel	Refund	Others
Inter-server correspondence method	◎Authorization completion notification	-	-	-	-
MasterPass	-	-	-	-	-
Subscription service	-	△Billing result notification	-	-	△Reversal result notification

*○ in the table will be sent if the result is successful.

*◎ in the table will be sent regardless of the result (success / failure). However, in the service via the consumers browser, if the consumer leaves halfway and the payment fails, the notification will not be sent.

*△ in the table will be sent if the result is unsuccessful.

4.2.3 Interface Details

For the interface details of the result notification message, refer to the separate document for each payment service (Table 4-1-1).

Chapter 5 Testing

5-1 Notes for testing

For the details of the test specification, please refer to the "Integration test guide".

Given below are the important points at the time of test implementation.

- When conducting a test transaction, please make sure to set the dummy mode of the MDK setting file to "1".

```
#Dummy mode (specified only during test)
DUMMY_REQUEST  = 1
```

- To conduct a test transaction, it is necessary to follow certain conditions (perdecided card number, perdecided payment amount, etc.).

  For the setting condition at the time of the test, please refer to "Integration test guide".

- Please be aware of the following parameters during transaction test.
  - ✓ Transaction ID (orderId): Please set it arbitrarily. It is necessary to set it to unique so that it does not repeat. In addition, you cannot use the transaction ID used in the test transaction again during a live transaction.
  - ✓ Amount: Please set valid amount data. (You cannot use "-" (minus) or a numeric value with a decimal point)
  - ✓ Post processing for live mode transactions which were originally executed in dummy mode and post processing for dummy mode transactions which were originally executed in production mode always result in an error.

5-2 Verification of Test Transaction Results on MAP

The merchant can use Merchant administration portal (MAP) to search for a specific transaction or to check the transaction status.

For more details, please refer to the “Usage Guide”.

＊About MAP

It is an abbreviation of Merchant Administration Portal, and is a web-type management tool that provides all types of information and functions for management and operation of payment.

URL: https://pay.veritrans.co.jp/maps/