VeriTrans4G Interface Details

~ 3-D Secure ~

Ver. 1.0.1 beta (April 2017~)

Table of Contents

Chapter 1　About this Document

1-1　Contents of this Guide

This guide is intended for developers integrating their website with VeriTrans4G using Merchant Development Kit (MDK) offered by VeriTrans Inc. It also describes the interface details used in VeriTrans4G 3-D Secure.

For more information about VeriTrans4G, please refer to the “VeriTrans4G Development Guide”.

Please refer to the separately provided interface details for each payment service.

`

1-2　Copyright and Contact Details

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

1-3　Revision History

2017/02 : Released Ver1.0.0

* The following are the updates from “VeriTrans3G interface details ~ 3-D Secure ~” Ver 2.0.3.
  Added “token” in “2.2.1 Request Message: MpiAuthorizeRequestDto”

2017/04 : Released Ver1.0.1

Corrected description of “token” in “2.2.1 Request Message: MpiAuthorizeRequestDto” “2.3.1 Request Message: MpiReAuthorizeRequestDto” and added description related to card information storage in description of "cardNumber", "cardExpire" and "securityCode".

Chapter 2　Interface Details ~ 3-D Secure ~

This chapter provides an explanation about the message (Dto) used in each payment. The fields given in the following table can be used by the merchants.

Though each message (Dto) may have fields that are not given in the following table, merchants cannot use such fields.

2-1 Common

■ The contents of the “Settings” column are as follows.

Request Message … Mandatory field: ○　Optional field: Δ　Settings disabled: X　Other conditions: *, *n (Conditions are given in the description column or outside the column.)

Response Message … Always returned: ○　Returned only when processing is successful: Δ　Not returned: X　Other conditions: *

■ About orderId (order ID)

Merchant should assign a (unique) number randomly. The number needs to be assigned for each order. Assign a (unique) ID that is different from other order IDs. The order ID must be unique across other payment services as well.

The order ID used in the test order cannot be re-used in the live order.

In the order ID, “-“ (hyphen), and “_” (underscore) can also be used besides the single byte alphanumeric characters.

■ About tampering check using vAuthInfo and authParams

We provide a tampering check functionality where the parameter received by the system from the consumer browser can be confirmed for tampering when the consumer browser is redirected from the payment server to the Online shop.

This tampering check is performed using “authParams” and “vAuthInfo” parameters that are returned while redirecting.

When the hash value calculated by the merchant system matches with the “vAuthInfo” received through the parameter at the time of redirecting, it is presumed that the parameters are not tampered.

Although this tampering check is not mandatory, invalid redirect messages may be received from the third party with the bad intent. Hence, **we strongly recommend that you perform a tampering check**. For details of the implementation method, please refer to the sample program offered by VeriTrans.

■ About capture and cancellation process

For the credit card payment along with 3-D Secure, capture is performed through the Capture process of the credit card payment. Further, it can be canceled or refunded through the

Cancel process.

For Capture and Cancel process, please refer to the “Interface Details ~ Credit Card Payment ~”.

Supplement: Subscription Service

The following document contains the interface details for using the subscription service.

“Interface Details ~ Subscription Service ~”

When using the subscription service, send payment request after setting the fields of request message given in this document and fields for subscription service.

Please note that the fields for the subscription service also get added in the response message.

2-2 3-D Secure

2.2.1 Request Message: MpiAuthorizeRequestDto

Request Message: MpiAuthorizeRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceOptionType	Payment service option type	Please refer to the description in the right column	"mpi-none" : 3-D Secure only "mpi-complete" : Complete authorization "mpi-company" : Normal authorization (card company bears risk) "mpi-merchant" : Normal authorization (card company and merchant bear risk)	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Please refer to the “About orderId (order ID)” described in “2-1 Common”	○
amount	Payment amount	Single byte numbers; 8 digits or less	1 to 99,999,999.	○
token	Token	Single byte alphanumeric characters and symbols; 36 characters	This is a token value used to identify credit card information issued by the token server. Please refer to the "MDK Token Development Guide" for more details.	Δ
cardNumber	Card number	Single byte numbers; 16 digits or less	(Important) Please do not use this parameter to comply with no electronic storage or transmission of any cardholder data. Numbers only, or the number can be specified with hyphen (it should be 19 digits or less when hyphen is included).	Δ
cardExpire	Card expiry	Single byte alphanumeric characters; 5 characters	(Important) Please do not use this parameter to comply with no electronic storage or transmission of any cardholder data. MM/YY (Month + "/" + Year) format (For example: "08/18").	Δ

Request Message: MpiAuthorizeRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
jpo	Payment type	Single byte alphanumeric characters; 83 characters or less Please refer to the description in the right column	“10” (lump-sum payment) “21” (lump-sum bonus) “61Cxx” (payment in installments, specify number of installments in xx.) "80" (revolving payment) ∗ If not specified, default is "10" (lump-sum payment). ∗ Available payment types are different for facilitator contract and aggregator contract. For details, please refer to the “Credit Card Payment Specifying Payment Type Information” of “Interface Details ~ Credit Card Payment ~”.	Δ
withCapture	Capture flag	Please refer to the description in the right column	“true”: Authorization and Capture “false”: Authorization only ∗ If not specified, "false" is set as a default value.	Δ
securityCode	Security code	Single byte numbers; 3 or 4 digits	(Important) Please do not use this parameter to comply with no electronic storage or transmission of any cardholder data. Security code	Δ
redirectionUri	Redirection URI	Single byte alphanumeric characters; 1024 bytes or less	Specifies the URI where verification result is expected If not specified, the already registered URI is used. **(Note)** **In case of payment service option type (serviceOptionType) as 3-D Secure only (“mpi-none”), SSL is mandatory. Always specify URI starting with “https: //”.**	Δ
httpUserAgent	HTTP user agent	No limitations	Browser information of consumer	○
httpAccept	HTTP accept	No limitations	Browser information of consumer	○
pushUrl	Push URL	Single byte characters that can be used in URL; 256 characters or less	Specifies result notification URL If not specified, already registered URL is used.	Δ
browserDeviceCategory	Terminal category	Please refer to the description in the right column	“0”: PC and smart phone "1": Feature phone ∗ If not specified, “0” is set as a default value.	Δ

2.2.2 Response Message: MpiAuthorizeResponseDto

Response Message: MpiAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceType	Payment service type	Single byte alphanumeric characters; 10 characters or less	Payment service type sent by request message.	○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	“success”: Normal termination “failure”: Abnormal termination	○
vResultCode	Detailed result code	String; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the “Result Code List”.	○
merrMsg	Error message	String	Process result in Japanese or English	○
marchTxn	Message ID	String; 100 characters or less	ID assigned by payment server per payment process message (including internal process). Multiple IDs are assigned to a single Order ID.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID randomly assigned and sent by the merchant at the time of payment request.	○
custTxn	ID assigned per order	String; 100 characters or less	ID (uniquely) assigned by payment server to link the order (Order ID).	○
txnVersion	MDK version	Single byte alphanumeric characters; 5 characters	Message version It is not generally used except in the case of a problem.	○
mpiTransactiontype	3-D Secure transaction type	String; 6 characters or less		Δ
reqCardNumber	Request card number	String; 16 characters or less	Value set in the request message Only first 6 characters and last 2 characters are displayed; remaining characters are masked with "*" (asterisk). (For example "411111********11")	Δ
reqCardExpire	Request card expiry	String; 5 characters or less	Value set in the request message All characters are masked with "*" (asterisk).(For example "*****")	Δ
reqAmount	Request order amount	String; 12 characters or less	Value set in the request message	Δ
reqJpoInformation	Request payment category information	String; 83 characters or less	Value set in the request message	Δ
reqWithCapture	Request capture flag	String; 5 characters or less	Value set in the request message	Δ

Response Message: MpiAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
reqSecurityCode	Request Security code	String; 4 characters or less	Value set in the request message All characters are masked with “0” (zero)	Δ
reqRedirectionUri	Request redirection URI	String; 1024 characters or less	Value set in the request message	Δ
reqHttpUserAgent	Request HTTP user agent	No limitations	Value set in the request message	Δ
reqHttpAccept	Request HTTP accept	No limitations	Value set in the request message	Δ
resResponseContents	Response contents	No limitations	Response given to the consumer by the merchant when the 3-D Secure is successful.	Δ
resCorporationId	Acquiring card company ID	String; 2 characters or less	Code of the card company with whom the merchant has signed a merchant contract. It is the code of Acquiring card company rather than credit card issuing card company. Please refer to the “Credit Card Payment - Acquirer List” of “Interface Details ~ Credit Card Payment ~”	Δ
resBrandId	Credit card brand ID	String; 2 characters or less	Following values are set "35": JCB "4": VISA "5": MASTER "34": AMEX "37": AMEX * "34" and "37" can be specified only if you have a contract with AMEX Safekey.	Δ
res3dMessageVersion	3-D Secure message version	String;10 characters or less	Message version of 3-D Secure protocol	Δ
authRequestDatetime	3-D Secure request date and time	String; 28 characters or less	Received time of 3-D Secure request EEE MMM dd HH:mm:ss JST yyyy format (For example: "Tue Mar 07 13:17:40 JST 2017")	Δ
authResponseDatetime	3-D Secure response date and time	String; 28 characters or less	Response time of 3-D Secure EEE MMM dd HH:mm:ss JST yyyy format (For example: "Tue Mar 07 13:17:40 JST 2017")	Δ

2-3 3-D Secure Re-order

Note) There is no need to use the credit card payment Re-Order function if you are using the membership management function of the subscription service.

The function similar to 3-D Secure re-order is provided in the subscription service.

It is possible to perform re-order by specifying member ID and card ID in request message (MpiAuthorizeRequestDto).

This eliminates the need for the merchant to manage the original order ID used in the re-order function.

2.3.1 Request Message: MpiReAuthorizeRequestDto

Request Message: MpiReAuthorizeRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceOptionType	Payment service option type	Please refer to the description in the right column	“mpi-none”: 3-D Secure only "mpi-complete": Complete authorization "mpi-company": Normal authorization (card company bears risk) "mpi-merchant": Normal authorization (card company and merchant bear risk)	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Please refer to the “About orderId (order ID)” described in “2-1 Common”	○
originalOrderId	Original order ID	Single byte alphanumeric characters; 100 characters or less	Order ID of past order for which re-order is to be performed	○
amount	Payment amount	Single byte numbers; 8 digits or less	1 to 99,999,999.	○
cardNumber	Card number	Single byte numbers;16 digits or less	(Important) Please do not use this parameter to comply with no electronic storage or transmission of any cardholder data. Numbers only, or the number can be specified with hyphen (it should be 19 digits or less when hyphen is included).	Δ
cardExpire	Card expiry	Single byte alphanumeric characters; 5 characters	(Important) Please do not use this parameter to comply with no electronic storage or transmission of any cardholder data. MM/YY (Month+ "/" + Year) format (For example: "08/18")	Δ

Request Message: MpiReAuthorizeRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
jpo	Payment category	Single byte alphanumeric characters; 83 characters or less Please refer to the description in the right column	"10" (lump-sum payment) "21" (Lump-sum payment) "61Cxx" (payment in installments; specify number of installments in xx.) "80" (revolving payment) ∗ If not specified, "10" (lump-sum payment) is set as a default value. ∗ Available payment types are different for facilitator contract and aggregator contract. For details, please refer to the “Credit Card Payment - Specifying Payment Type Information” of “Interface Details ~ Credit Card Payment ~”.	Δ
withCapture	Capture flag	Please refer to the description in the right column	"true": Authorization and Capture "false": Authorization only ∗ If not specified, "false" is set as a default value.	Δ
securityCode	Security code	Single byte numbers; 3 or 4 digits	Security code	Δ
redirectionUri	Redirection URI	Single byte alphanumeric characters;1024 bytes or less	Specifies the URI where verification result is expected. If not specified, already registered URI is used. **(Note)** **In case of payment service option type (serviceOptionType) as 3-D Secure only (“mpi-none”), SSL is mandatory. Always specify URI starting with “https: //”.**	Δ
httpUserAgent	HTTP user agent	No limitations	Browser information of consumer	○
httpAccept	HTTP accept	No limitations	Browser information of consumer	○
pushUrl	Push URL	Single byte alphanumeric characters that can be used in URL; 256 characters or less	Specifies result notification URL If not specified, already registered URL is used.	Δ
browserDeviceCategory	Device category	Please refer to the description in the right column	"0": PC and Smart phone "1": Feature phone ∗ If not specified, “0” is set as a default value.	Δ

2.3.2 Response Message: MpiReAuthorizeResponseDto

Response Message: MpiReAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceType	Payment service type	Single byte alphanumeric characters; 10 characters or less	Payment service type sent by request message.	○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	“success”: Normal termination “failure”: Abnormal termination	○
vResultCode	Detailed result code	String; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the “Result Code List”.	○
merrMsg	Error message	String	Process result in Japanese or English	○
marchTxn	Message ID	String;100 characters or less	ID assigned by payment server per payment process message (including internal process). Multiple IDs are assigned to a single order ID.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID assigned randomly and sent by the merchant at the time of payment request.	○
custTxn	ID assigned per order	String;100 characters or less	ID (uniquely) assigned by payment server to link the order (order ID).	○
txnVersion	MDK version	Single byte alphanumeric characters; 5 characters	Message version It is not generally used except in the case of a problem.	○
mpiTransactiontype	3-D Secure transaction type	String; 6 characters or less		Δ
reqCardNumber	Request card number	String;16 characters or less	Value set in the request message	Δ
reqCardExpire	Request card expiry	String; 5 characters or less	Value set in the request message	Δ
reqAmount	Request order amount	String;12 characters or less	Value set in the request message	Δ
reqJpoInformation	Request payment category information	String; 83 characters or less	Value set in the request message	Δ
reqWithCapture	Request concurrent capture	String; 5 characters or less	Value set in the request message	Δ
reqSecurityCode	Security code	String; 4 characters or less	Value set in the request message	Δ

Response Message: MpiReAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
reqRedirectionUri	Request redirection URI	String; 1024 characters or less	Value set in the request message	Δ
reqHttpUserAgent	Request HTTP user agent	No limitations	Value set in the request message	Δ
reqHttpAccept	Request HTTP accept	No limitations	Value set in the request message	Δ
resResponseContents	Response contents	No limitations	Response given to the consumer by the merchant when the 3-D Secure is successful.	Δ
resCorporationId	Acquiring card company ID	String; 2 characters or less	Code of the card company with whom the merchant has signed a merchant contract. It is the code of Acquiring card company rather than credit card issuing card company. Please refer to the “Credit Card Payment - Acquirer List” of “Interface Details ~ Credit Card Payment ~”	Δ
resBrandId	Credit card brand ID	String; 2 characters or less	Following values are set "35": JCB "4": VISA "5": MASTER "34": AMEX "37": AMEX * "34" and "37" can be specified only if you have a contract with AMEX Safekey.	Δ
res3dMessageVersion	3-D Secure message version	String; 10 characters or less	Message version of 3-D Secure protocol	Δ
authRequestDatetime	3-D Secure request date and time	String; 28 characters or less	Time of receiving the 3-D Secure request EEE MMM dd HH:mm:ss JST yyyy format (For example "Tue Mar 07 13:17:40 JST 2017")	Δ
authResponseDatetime	3-D Secure response date and time	String; 28 characters or less	Time of 3-D Secure response EEE MMM dd HH:mm:ss JST yyyy format (For example "Tue Mar 07 13:17:40 JST 2017")	Δ

2-4 3-D Secure Result

When the ‘Additional parameters in the payment result’ feature is enabled, parameters other than “request ID” and “order ID” mentioned below are also returned at the time of redirect. In other words when the ‘Additional parameters in payment result’ feature is disabled, only “request ID” and “order ID” are shared as the parameters at the time of redirect.

To retain the compatibility when ‘Additional parameters in payment result' feature is not used, first character will be upper case for “RequestID” and “OrderID” which are the field names of “Request ID” and “Order ID”. In case of other field names, all first characters will be in lower case.

In future, the number of parameters that are returned during redirect may get added. Therefore, carry out the implementation in such a way that there will be no impact to the system even if parameters not mentioned below are returned.

2.4.1 Redirect Contents: Contents that are Sent (POST) to the Merchant from Payment Server via Consumer Browser

＊Following are common redirect parameters at the time of 3-D Secure and 3-D Secure re-order.

Redirect Contents: Contents that are sent (POST) to the merchant from payment server via consumer browser				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
RequestId	Request ID	Single byte alphanumeric characters and symbols; 128 characters or less	Key use to identify the result of 3-D Secure.	○
OrderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID assigned randomly and sent by the merchant at the time of payment request	○
Note) The following fields are sent when the ‘Additional parameters in payment result’ feature is enabled.				
reqAmount	Request order amount	String;12 characters or less	Value set in the request message	○
reqCardNumber	Request card number	String; 16 characters or less	Value set in the request message Only first 6 characters and last 2 characters are displayed and others are masked with "*" (asterisk). (For example: “411111********11”)	○
reqCurrencyUnit	Request currency unit	String; 3 characters or less	Value set in the request message	○
mpiMstatus	3-D Secure result code	Single byte alphanumeric characters; 32 characters or less	Process result status of 3-D Secure. “success”: Normal termination “failure”: Abnormal termination	○
vResultCode	Detailed result code	String; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the “Result Code List”.	○

Redirect Contents: Contents that are sent (POST) to the merchant from payment server via consumer browser				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
vAuthInfo	Hash value for manipulation check	String Please refer to the description in the right column	The following strings are concatenated and hash value is calculated by SHA-256. • Merchant CCID • Concatenated string of parameter value (concatenated as per the sequence indicated by authParams) • Password When concatenating parameter values, only the parameter values are concatenated without including the parameter name and delimiter. UTF-8 is used as the character encoding while encoding the concatenated character string into the binary.	○
authParams	Hash value calculation parameter sequence	String Please refer to the description in the right column	The Value indicating the concatenating sequence of the parameter is used to calculate the hash value of vAuthInfo. The Comma delimiter string used between the parameter names is Base64 encoded. On decoding, the the Comma delimiter string can recover. For example) "mpiMstatus,vResultCode,OrderId" "OrderId,mpiMstatus,vResultCode" (As the sequence is not fixed, it is necessary to perform the process dynamically on receiving the request)	○
Note) The following fields are sent when the 'Additional parameters in payment result' feature is enabled and serviceOptionType is (mpi-complete / mpi-company / mpi-merchant).				
cardMstatus	Card result code	Single byte alphanumeric characters; 32 characters or less	Process result status of card payment "success": Normal termination "failure": Abnormal termination "pending": Pending ∗ Blank when 3-D Secure result code is failure	Δ
cardTransactionType	Card transaction type	String; 6 characters or less	Detail status of card payment order "a": authorization "ax": authorization (expired) "ap": authorization (pending) "ac": authorization and capture "acp": authorization and capture (pending)	Δ
centerRequestDate	Center request date and time	String; 14 characters or less	Date and time of sending payment request to card payment center YYYYMMDDhhmmss format	Δ

Redirect Contents: Contents that are sent (POST) to the merchant from payment server via consumer browser				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
connectedCenterId	Card processing payment network	String; 5 characters or less	Name of card payment network between payment server and card company For example: ‘jcn‘	Δ
acquirerCode	Acquirer code	String; 2 characters or less	Code of the card company to which payment request message was first delivered. For the Acquirer List, please refer to the “Credit Card Payment Acquirer List” of “Interface Details ~ Credit Card Payment~”.	Δ
authCode	Approval code	Single byte alphanumeric characters; space; 7 characters or less	Approval code issued by card company	Δ
Note) The following fields are sent when ‘Additional parameters in payment result’ feature is enabled and serviceOptionType is (mpi-none).				
dddMessageVersion	3-D Secure message version	Single byte alphanumeric characters; 10 characters or less	Message Version Number (For example “1.0.2”)	Δ
dddTransactionId	3-D Secure transaction ID	Single byte alphanumeric characters; "+","-","="; 28 characters or 0 character	Value obtained after encoding the value of Transaction Identifier(XID) with Base64	Δ
dddTransactionStatus	3-D Secure transaction status	Single byte alphabetic character; 1 character Please refer to the description in the right column	3-D Secure transaction status "Y": 3-D Secure successful "N": 3-D Secure failed (due to issuer or card holder) "U": 3-D Secure failed (reason other than mentioned above) "A": Attempt (3-D Secure authentication attempted successfully) "": No value (blank)	Δ
dddCavvAlgorithm	3-D Secure CAVV Algorithm	Alphanumeric character; 1 character; Please refer to the description in the right column	3-D Secure CAVV algorithm "0": HMAC "1": CVV "2": CVV with ATN "3": SPA Algorithm "": No value (blank) * Other than the above values may be specified depending on the card company specifications.	Δ
dddCavv	3-D Secure CAVV	Single byte alphanumeric characters; 28 characters or 0 digits	3-D Secure CAVV	Δ

Redirect Contents: Contents that are sent (POST) to the merchant from payment server via consumer browser				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
dddEci	3-D Secure ECI	Single byte numbers; 2 digits Please refer to the description in the right column	3-D Secure ECI "01": Attempt (Master Card) "02": Authentication successful (Master Card) "05": Authentication successful (VISA, JCB) "06": Attempt (VISA, JCB) or Not enrolled for 3-D Secure (Master Card, VISA, JCB) "07": Authentication execution not possible (Master Card, VISA, JCB) "": No value (blank)	Δ

✧ About the tampering check

Although the consumer browser is redirected from the payment server to the Online shop, we recommend that you check the tampering of the POST parameters received by the system. This tampering check is performed using “authParams” and “vAuthInfo” parameters that are returned while redirecting.

Please refer to the aforementioned “About tampering check using vAuthInfo and authParams”.

∗ Although this tampering check is not mandatory, invalid redirect messages may be received from the third party with the bad intent. Hence**, we strongly recommend that you perform the tampering check**.

∗ For details of the implementation method, please refer to the sample program offered by VeriTrans

2-5 Result Notification (3-D Secure)

The payment server notifies the result of 3-D Secure and credit card payment to the merchant. It is notified after 3-D Secure result verification (or after credit card payment execution if applicable), irrespective of successful or unsuccessful payment. For the specifications common across the services related to result notification, please refer to the “Development Guide”.

Item no.	Field Name	Item Name	Format and Limitations	Description
1	numberOfNotify	Number of records in notification	Single byte numbers; 4 digits or less	Single notification can contain a maximum of 1,000 records 1001 onwards records are notified next time
2	pushTime	Transmission time	Single byte numbers; 14 digits	Date and time when the notification is sent from payment server YYYYMMDDhhmmss format
3	pushId	Identification ID	Single byte numbers; 8 digits	(Unique) ID assigned while performing push process Note) ID used in other payment services may be duplicated.
Below mentioned item numbers (4~9) are repeated for the notification records. Further, 4 digits serial number (0000~0999) is assigned after the field name.				
4	orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Order ID
5	vResultCode	Detailed result code	Single byte alphanumeric characters; 16 characters	Code that represents the process result in detail. For the details of process result code, please refer to the “Result Code List”.
6	txnType	Transaction type	Single byte alphanumeric characters; 32 characters or less	"Verify": 3-D Secure result
7	mpiMstatus	3-D Secure result code	Single byte alphanumeric characters; 8 characters or less	Process result status of 3-D Secure "success ": Normal termination "failure": Abnormal termination
8	cardMstatus	Card result code	Single byte alphanumeric characters; 8 characters or less	"success ": Normal termination "failure": Abnormal termination “pending”: Pending ∗ Blank in case of mpi-none (3-D Secure unit service) and when 3-D Secure result code is failure
9	dummy	Dummy payment flag	Single byte numbers; 1 digit	In case of dummy order, “1” and in case of live order, “0”.

(Note)

✧ When using result notification function of 3-D Secure, please specify result notification URL of 3-D Secure by changing various settings of MAP.

✧ As pushId (identification ID) may be duplicated with the ID used in other payment services, please do not process as a unique key.

2-6 Appendix

2.6.1 3-D Secure Transaction Type

This section provides the process status of corresponding orders.

The following table explains the types available in normal payments. (Types not mentioned in this table also exist.)

Type Name	Setting Value	Remarks
3-D Secure (approved)	auth	This indicates that 3-D Secure is successfully approved.
3-D Secure (approved, conditional success)	authc	This indicates that 3-D Secure is successfully approved with some conditions.
3-D Secure (verification)	vd	This indicates that the verification of 3-D Secure has succeeded.
3-D Secure (Verification, conditional success)	vdc	This indicates that 3-D Secure is successfully verified with some conditions.
3-D Secure (skip)	vds	This indicates that 3-D Secure is skipped.

Chapter 3 Other - Supplementary Items

3-1 Payment Result Decision

In 3-D Secure, card payment is done when the 3-D Secure succeeds (includes attempt) excluding the case when payment service option type (serviceOptionType) is “mpi-none”.

In merchant system, make sure to ship the items and offer services only after confirming if the card payment is successful.

Examples of result code when the card payment is successful.

vResultCode : G012A001

mpiMstatus : success

cardMstatus : **success**

Example when 3-D Secure is successful and card payment fails.

vResultCode : G012AG33

mpiMstatus : success

cardMstatus : **failure**

3-2 Order Status Inconsistency between Payment Server and Merchant Site

There is a possibility of order status inconsistency between VeriTrans3G and merchant site as merchant is unaware of successful payment because consumer browser is not redirected back to merchant's payment completion page or failed to redirect to merchant site even after successful 3D-Secure and card payment. This problem could arise in the following scenarios:

- Staying on 3-D Secure page (password input page) of card Company for a longer period could cause session time out (order expiry etc.) in merchant site. Due to which, the payment result is not reflected at merchant site.
- Operations such as closing the browser after the successful completion of 3-D Secure.

Carry out either of following on merchant site to deal with this problem.

(1) Receive payment result notification (PUSH) from payment server (Please refer to the “3-4 Result Notification (Push)”)

(2) Search the incomplete order with Search command and check if the payment is complete.

When using Search command, always refer to the “3-1 Payment Result Payment Result” and confirm if “Card payment is successful”.

3-3 Additional Parameters in Payment Result

‘Additional parameters in payment result’ is the feature that shares not only the existing[*1] “request ID” and “order ID” but also the detailed information of payment result as the parameters which are redirected (POST) to the completion page of merchant from payment server via consumer browser.

Enabling this feature makes it possible to obtain the success and failure of payment without using Search command, which were earlier used to be obtained by executing Search command in merchant system.

∗ 1: In the existing 3-D Secure, only “Request Id” and “Order ID” were shared as the redirect parameters. In order to obtain success and failure of payment, it was necessary to execute Search command after performing 3-D Secure (Authorize) or 3-D Secure re-order (ReAuthorize) command.

To use this feature it is necessary to do the settings at the VeriTrans side because of its compatibility with existing 3-D Secure interface.

✓ **If the merchant ID issued before releasing this function, settings of additional parameters were off. Only "Request ID" and "Order ID" parameters are sent (POST). In this case, please contact us as it is necessary to change the settings.**

(Note)

In case of payment service option type (serviceOptionType) as 3-D Secure only (“mpi-none”), SSL (https: //) is mandatory for the URI of result page of the merchant system where result is to be navigated.

Please note that no error occurs even after specifying HTTP URI as the testing in dummy mode is also taken into consideration.

3-4 Result Notification (Push)

This feature sends (Push) 3-D Secure result and credit card payment result to merchant from payment server.
In 3-D Secure, the credit card payment is performed at the time when the verification (Verify) request is sent to the payment server by redirecting via consumer browser, after the password authentication in ACS. However, if the consumer leaves the page in between while returning to merchant site through the redirection after executing credit card payment, payment is shown as incomplete on the merchant site even though the card payment is successful in the payment server. Such inconsistency issues may occur.
On receiving the result notification, it's possible to understand whether the payment has been made on the merchant site even if the consumer has left in between.
Result notification is sent (PUSH) even if the result of the credit card payment has failed.

Settings for this feature can be done through "Change various settings" of MAP (Merchant Administrative Portal). For details, please refer to the usage guide of MAP.
If you have MDK that does not support this feature, you may need to replace the MDK to specify "PUSH URL (pushurl) in request message of dummy order.
For the handling of result notification, please refer to the "4G_Development Guide".

3-5 Feature Phone Support

Consumers using the feature phone can use 3-D Secure. (* 1).
For implementing 3-D Secure in feature phone, a value is set in "Device category (browserDeviceCategory)" of request message that indicates the access from feature phone.

In payment server, device category is not automatically identified from User-Agent value.
The merchant can set the "Device category (browserDeviceCategory)" in request message by taking into consideration the device (whether it is for PC or for feature phone) used by the consumer.

In feature phone, JavaScript may not be supported or is disabled depending upon the device.(There are many cases where JavaScript is not supported or is disabled.)
Therefore, whether its accessing from a PC browser where navigation is automatic, or in the case of accessing from a feature phone, it is necessary to click the button exclusively to navigate to the next page.
For the redirect timing in the consumer browser, please refer to the description related to 3-D Secure in "Supplement of Development Guide Overview of System Flow Diagram".

(* 1) There are also card companies which do not support 3-D Secure on feature phones. Therefore, for some cards, 3-D Secure can be performed on a PC browser but not on feature phone.