Korean Cyber Services Using Alternative Payment Services

Simple Order API

Developer Guide

Restricted Rights Legends

For Government or defense agencies: Use, duplication, or disclosure by the Government or defense agencies is subject to restrictions as set forth the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and in similar clauses in the FAR and NASA FAR Supplement.

For civilian agencies: Use, reproduction, or disclosure is subject to restrictions set forth in subparagraphs (a) through (d) of the Commercial Computer Software Restricted Rights clause at 52.227-19 and the limitations set forth in Cybersource Corporation's standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States.

Trademarks

Authorize.Net, eCheck.Net, and The Power of Payment are registered trademarks of Cybersource Corporation. Cybersource and Cybersource Decision Manager are trademarks and/or service marks of Cybersource Corporation. Visa, Visa International, Cybersource, the Visa logo, the Cybersource logo, and 3-D Secure are the registered trademarks of Visa International in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Version: 23.01

Contents

About This Guide

This section provides you with information about the Korean Cyber Payments Using Alternative Payent Services Simple Order API guide.

Audience and Purpose

This guide is written for merchants who want to offer Korean Cyber Payment (KCP) services to customers and describes tasks a merchant must complete in order to make a payment, request the status of a payment, or refund a payment. It is intended to help the merchant provide a seamless customer payment experience.

Conventions

The following special statements are used in this document:

Important: An *Important* statement contains information essential to successfully completing a task or learning a concept.

Warning: A *Warning* contains information or instructions, which, if not heeded, can result in a security risk, irreversible loss of data, or significant cost in time or revenue or both.

Related Documentation

Refer to these sites for technical documentation:

- Technical Documentation Portal: https://docs.cybersource.com/en/index.html
- Support Center: https://www.cybersource.com/en-us/support/technical-documentation.html

Customer Support

For support information about any service, visit the Support Center:

http://www.cybersource.com/support

Recent Revisions to This Document

23.01

Test Triggers

Added testing information for Check Status and Refunds. See Check Status Testing Information (on page 30) and Refund Testing Information (on page 38).

22.01

New Alternative Payment Services

Added KakaoPay, Naver Pay, and Payco as payment types. See Payment Types (on page 8).

Added **otherTax_greenTaxAmount** and **ap_appUrl** optional fields. See Optional Fields for a Sale Using the Simple Order API (on page 16).

19.01

Multibyte Character Strings

Added information about multibyte character strings. See Working with Multibyte Character Strings (on page 41).

18.01

Initiate Payment Service

Updated the “Initiate Sale Service” section. See Korean Cyber Services Workflow (on page 9).

Updated the “Merchant URL” section. See Merchant URL Parameters (on page 11).

Updated the “Initiate Sale” reply examples with encrypted string. See Simple Order API Example: Initiate Sale Request (on page 18).

15.01

Supported Browsers and Mobile Devices

Updated Introduction regarding supported browsers and mobile devices. See Introduction to Korea Cyber Payment Services (on page 8).

14.01

Related Documents

Updated the “Related Documents” section.

Added the **apCheckStatusReply_processorTransactionID** API reply field. See Simple Order API Field Descriptions (on page 39).

Introduction to Korea Cyber Payment Services

Korea Cyber Payment (KCP) is a South Korean-based payment services company. KCP provides Internet-based payment services to its customers.

All KCP payments are processed in South Korean Won (KRW).

Contact KCP for a list of supported desktop browsers and mobile devices.

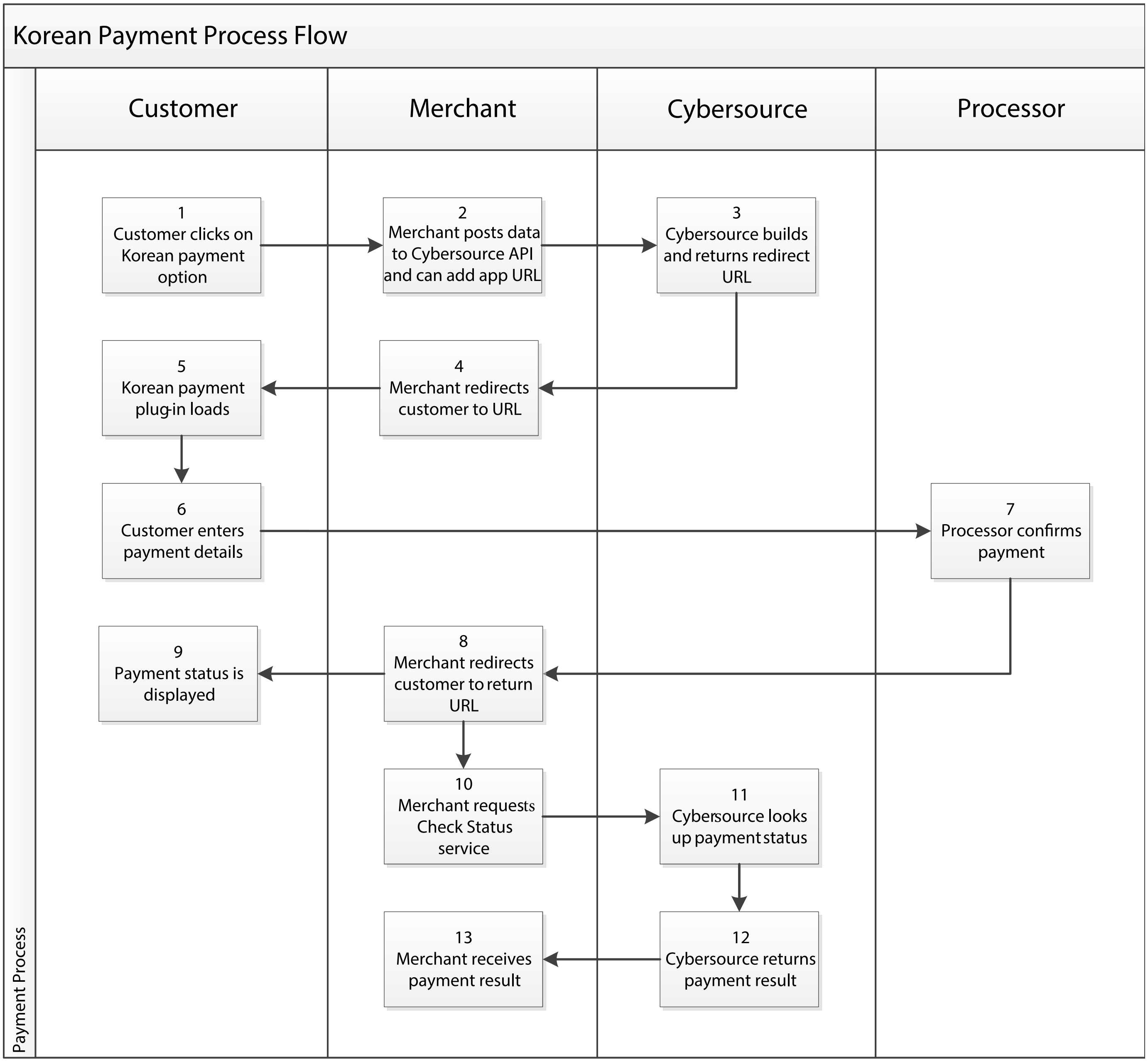
Payment Types

KCP accepts these payment types.

Payment Types

Payment Type	Description
ACC	Credit card payment.
KKP	KakaoPay using alternative payment service.
MBP	Mobile billing payment.
NVP	Naver Pay using alternative payment service.
PYC	Payco using alternative payment service.
RBT	Real-time bank transfer.

Korean Cyber Services Workflow


Korean Payment Process Flow
Payment Process
Customer
Merchant
Cybersource
Processor
1 Customer clicks on Korean payment option
2 Merchant posts data to Cybersource API and can add app URL
3 Cybersource builds and returns redirect URL
4 Merchant redirects customer to URL
5 Korean payment plug-in loads
6 Customer enters payment details
7 Processor confirms payment
8 Merchant redirects customer to return URL
9 Payment status is displayed
10 Merchant requests Check Status service
11 Cybersource looks up payment status
12 Cybersource returns payment result
13 Merchant receives payment result

1. The customer chooses the Korean payment type on your website's checkout or payment page. The alternative payment methods, KakaoPay, Naver Pay, and Payco, will display here as payment options if available.

2. Using the Cybersource API, you POST your merchant details and order information from the initiate sale service request to Cybersource. The merchant may include the `ap_appUrl` optional field to wake up the app during the payment process.

3. Cybersource creates a signed and encrypted merchant URL. See Merchant URL Parameters (on page 11) to redirect the customer to the KCP website. This signed and encrypted merchant URL is sent back to you.

4. The customer is redirected to the URL that Cybersource sent you in Step 3 (on page 9). This URL directs the customer to the KCP web site.

5. The customer clicks **Submit** and the Korean payment plug-in is launched. The plug-in is an executable file that is downloaded to the customer's device when the customer is browsing from a Windows operating system.

6. The customer enters payment details and confirms the payment.

7. The processor confirms the payment and sends verification to Cybersource.

    a. If using KakaoPay, Naver Pay, or Payco, the app URL provided in the sale request will now wake up the merchant app once payment processing is complete. See Step 2 (on page 9).

8. Cybersource stores the payment status. You can retrieve the payment status by requesting the check status service. See Check Status (on page 23).

9. The customer's browser is redirected to the return URL that is specified during merchant boarding. The return URL contains the payment status. Cybersource recommends that this status is used only for browser control to display a user-friendly message to the customer. You should ship goods based on the payment status retrieved from the check status service request.

10. You request the check status service from Cybersource. Cybersource recommends that this service be requested immediately after Step 9 (on page 10) and every one-hour until the payment status is `COMPLETED`, `DECLINED`, or `CANCELLED`.

11. Cybersource verifies the payment status with the processor.

12. Cybersource sends you the payment result.

13. You receive the payment result. If the Check Status result is:

    - `PENDING`—payment is still in progress. Do not ship the goods.
    - `COMPLETED`—payment was successful. Ship the goods.
    - `DECLINE`—payment was rejected by the processor. Do not ship the goods.
    - `CANCELLED`—payment was cancelled by the customer. Do not ship the goods.

Merchant URL Parameters

After you initiate a sale, the **apInitiateReply_merchantURL** response field will contain the merchant URL that will redirect the customer to the KCP website.

Important: Cybersource generates and sends the URL with encrypted data to KCP. Merchants do not use the fields mentioned in this section and should only use this information as a reference to better understand the sale response.

The merchant URL consists of these parameters.

Merchant URL Parameters

Parameter	Description
KCP website	Test: `https://certpay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp` Live: `https://pay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp`
Encrypted Data	All the customer's payment information signed and encrypted in a string of data.

Encrypted Data

The merchant URL consists of an encrypted string of characters representing the payment details.

The encrypted data consists of these parameters.

Merchant URL Encrypted Data Parameters

Parameter	Description
AppUrl	App URL used by KCP to wake up the merchant app after payment processing. **Format:** `{a-z}://`
buyr_mail	Customer email.
buyr_name	Customer name.
buyr_tel	Customer telephone number.

Merchant URL Encrypted Data Parameters (continued)

Parameter	Description
comm_green_deposit_mny	Refundable deposit amount. **Conditions:** • If this field is present and merchant ID enabled, payment will be processed. • If this field is present and merchant ID disabled, payment will not be processed. • If this field is not present and merchant ID enabled, payment will be processed. • If this field is not present and merchant ID disabled, payment will be processed.
currency	Three-digit numeric currency code used for the order. KRW: `410`
eng_flag	English-language indicator for the KCP plug-in.
escw_used	Provides the escrow option to a customer. **Possible values:** • `Y`: Yes • `N`: No ! **Important:** You must have the appropriate contract with KCP to offer escrow. Escrow is supported only for bank transfers made using a desktop PC. It is not supported for mobile devices. Escrow can be used only for transactions with an amount greater than 50,000 KRW.
good_mny	Payment amount.
good_name	Product purchased by the customer.
ordr_idxx	Unique merchant-generated transaction ID. If it is not included, Cybersource will generate this transaction ID.

Merchant URL Encrypted Data Parameters (continued)

Parameter	Description
pay_method	Payment method. **Possible values:** • `100000000000`—Credit card. • `100000000000`—KakaoPay, Naver Pay, and Payco. • `010000000000`—Bank transfer. • `000010000000`—Mobile billing.
quotaopt	Installment payment frequency for credit card payments only. **Possible values:** `00` to `12`. **Default value:** `12`.
ret_url	Merchant-defined URL for displaying the payment results to the customer. You can also specify this URL during merchant boarding.
signature	RSA signature and SHA-256 data hashing.
site_cd	KCP merchant ID.
site_logo	KCP user interface, which can be customized with a merchant logo. The image size must not exceed 150 (w) x 50 (h) pixels. **Format:** JPG or GIF.
site_name	Merchant brand name.
skin_indx	Payment UI skin number. Each number displays a different color of the KCP plug-in UI.

Web Browser Restrictions

Some web browsers restrict browser URL lengths. Cybersource recommends using the following code when performing the redirect using client-side JavaScript to avoid issues with URL lengths:

```
<SCRIPT LANGUAGE='JavaScript'>
    var win=window.open('','_blank');
    win.location='" . $apInitiateReply->merchantURL . "';
</SCRIPT>
```

Initiate Sale

When the customer chooses the Korean payment type during checkout, you can initiate a sale by sending merchant details and order information to Cybersource in a sale service request.

Fields Specific to this Use Case

These API fields are required specifically for this use case.

apInitiateService_run

Set value to `true`.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

purchaseTotals_currency

Set value to `KRW`.

Endpoint

Set the `apInitiateService_run` field to `true`.

Send the request to `https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`.

Merchant App

If the customer is using a merchant app, the **ap_appUrl** optional field enables you to set a redirect URL to take the customer back to the merchant app once they complete the payment on the KCP website.

Required Fields for a Sale Using the Simple Order API

Use these fields to initiate a sale request.

apInitiateService_escrowAgreement

apInitiateService_languageInterface

apInitiateService_returnURL

apInitiateService_run

Set value to `true`.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

merchantID

merchantReferenceCode

purchaseTotals_currency

Set value to `KRW`.

purchaseTotals_grandTotalAmount

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Optional Fields for a Sale Using the Simple Order API

Use these optional fields to initiate a sale request.

apInitiateService_productDescription

apInitiateService_productName

apInitiateService_reconciliationID

ap_appUrl

billTo_city

billTo_country

billTo_email

billTo_firstName

billTo_lastName

billTo_phoneNumber

billTo_postalCode

billTo_state

billTo_street1

billTo_street2

installment_frequency

Supported only for credit card payments when the amount of the transaction is greater then 50,000 KRW. Cards issued outside of Korea are not supported. Depending on the agreement with KCP and the issuers, there could be a service charge. KCP has an agreement with all issuers to process installment payments.

otherTax_greenTaxAmount

purchaseTotals_taxAmount

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Simple Order API Example: Initiate Sale Request

Use this example as a reference for a successful sale request and response.

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206">
<merchantID>kcp_merchant2</merchantID>
<merchantReferenceCode>TC55255-1</merchantReferenceCode>
<clientLibrary>Java XML</clientLibrary>
<clientLibraryVersion>5.0.2</clientLibraryVersion>
<clientEnvironment>Linux/3.10.0-1160.42.2.el7.x86_64/Oracle
 Corporation/1.8.0_301</clientEnvironment>
<billTo>
    <firstName>John</firstName>
    <lastName>SMITH</lastName>
    <street1>201 S. Division St.&lt;/</street1>
    <street2>Unit 111</street2>
    <city>Ann Arbor</city>
    <state>MI</state>
    <postalCode>98005</postalCode>
    <country>KR</country>
    <phoneNumber>999-999-9999</phoneNumber>
    <email>null@cybersource.com</email>
</billTo>
<purchaseTotals>
    <currency>KRW</currency>
    <grandTotalAmount>100</grandTotalAmount>
</purchaseTotals>
<installment>
    <frequency>0</frequency>
</installment>
<otherTax>
    <greenTaxAmount>10</greenTaxAmount>
</otherTax>
<apPaymentType>KKP</apPaymentType>
<apInitiateService run="true">
    <productName>Television</productName>
    <productDescription>Television</productDescription>
    <reconciliationID>1234512345678800888</reconciliationID>
    <escrowAgreement>Y</escrowAgreement>
    <languageInterface>Korean</languageInterface>
</apInitiateService>
<ap>
    <appUrl>gmailLite://abc</appUrl>
</ap>
</requestMessage>
```

Response

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.206">
<c:merchantReferenceCode>TC55255-1</c:merchantReferenceCode>
<c:requestID>6669371940396961403016</c:requestID>
<c:decision>ACCEPT</c:decision>
<c:reasonCode>100</c:reasonCode>
<c:requestToken>AxjnrwSTaYwg6/GulJCIAF8mxZM2jViyZtGrZu4cMGDhw4UafAybOyzQfIR+Qybf/Y
uLFOEnpNpjCDr8a6UkIgAAuQ1n</c:requestToken>
<c:apInitiateReply>
    <c:reasonCode>100</c:reasonCode>

 <c:merchantURL>https://
testpay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp?AppUrl=http%3A%2F
%2FgmailLite://
abc&amp;buyr_mail=null%40cybersource.com&amp;buyr_name=John+SMITH&amp;buyr_tel=999
9999999&amp;currency=410&amp;eng_flag=N&amp;escw_used=N&amp;good_mny=100&amp;good_
name=Television&amp;noti_url=https%3A%2F
%2Fwww.cybersource.com%2Fnotify%2Fkcp&amp;ordr_idxx=1234512345678800888&amp;pay_me
thod=100000000000&amp;quotaopt=0&amp;ret_url=http%3A%2F
%2Fkcp.cybersource.com%2Fkcp_return.jsp&amp;signature=n%2Fj7WnkAIBVcwA2mwMOhBHgEGM
hnbcYk8lQtt%2F7Sx5GnlD%2FImuhZAU0K8KR8MmNR%0D
%0AwzTF9mziNDbpu6XB2VDEDe4tnrPeK%2Buwul6gtm3cIU%2BT3hfgQlx3%2BvG2dLw67dB3%0D%0AIec
2T0fRCht06S2p%2Ft44Rv%2FEZ%2FkosmfYd%2F2kSMJDASkAzyUUs3ZxX9BT37QKwYBu%0D
%0Au67zWLk0RWhVXk5Ksq0C%2F9KG2Pb5AVrEt9PYhRzaQHrxrcYXPRVT2ULx8QoFInG2%0D%0A
%2Bbwt%2B4Jn5Uo7fD56Mo%2BwfJMi2A9LYXBsnjDhAt3FiV%2BWrTBpI7BqJaAapf1H5%2Fpk%0D
%0AY67vgHbZ2uma57AJS4FM3Q%3D
%3D&amp;site_cd=S6314&amp;site_name=CyberSource</c:merchantURL>
    <c:reconciliationID>1234512345678800888</c:reconciliationID>
</c:apInitiateReply>
<c:reserved>
    <ics_message xmlns=""urn:schemas-cybersource-com:transaction-data:ics"">
        <ics_return_code>1000000</ics_return_code>
        <ics_rcode>1</ics_rcode>
        <ics_rflag>SOK</ics_rflag>
        <ap_initiate_rcode>1</ap_initiate_rcode>
        <request_id>6669371940396961403016</request_id>
        <ap_initiate_rflag>SOK</ap_initiate_rflag>
        <ics_rmsg>Request was processed successfully.</ics_rmsg>
        <ap_initiate_return_code>1900000</ap_initiate_return_code>

 <ap_initiate_merchant_url>https://
testpay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp?AppUrl=http%3A%2F
%2FgmailLite://
abc&amp;buyr_mail=null%40cybersource.com&amp;buyr_name=John+SMITH&amp;buyr_tel=999
9999999&amp;currency=410&amp;eng_flag=N&amp;escw_used=N&amp;good_mny=100&amp;good_
name=Television&amp;noti_url=https%3A%2F
%2Fwww.cybersource.com%2Fnotify%2Fkcp&amp;ordr_idxx=1234512345678800888&amp;pay_me
thod=100000000000&amp;quotaopt=0&amp;ret_url=http%3A%2F
```

```
%2Fkcp.cybersource.com%2Fkcp_return.jsp&amp;signature=n%2Fj7WnkAIBVcwA2mwMOhBHgEGM
hnbcYk8lQtt%2F7Sx5GnlD%2FImuhZAU0K8KR8MmNR%0D
%0AwzTF9mziNDbpu6XB2VDEDe4tnrPeK%2Buwul6gtm3cIU%2BT3hfgQlx3%2BvG2dLw67dB3%0D%0AIec
2T0fRCht06S2p%2Ft44Rv%2FEZ%2FkosmfYd%2F2kSMJDASkAzyUUs3ZxX9BT37QKwYBu%0D
%0Au67zWLk0RWhVXk5Ksq0C%2F9KG2Pb5AVrEt9PYhRzaQHrxrcYXPRVT2ULx8QoFInG2%0D%0A
%2Bbwt%2B4Jn5Uo7fD56Mo%2BwfJMi2A9LYXBsnjDhAt3FiV%2BWrTBpI7BqJaAapf1H5%2Fpk%0D
%0AY67vgHbZ2uma57AJS4FM3Q%3D
%3D&amp;site_cd=S6314&amp;site_name=CyberSource</ap_initiate_merchant_url>
        <ics_decision_reason_code>100</ics_decision_reason_code>
        <ap_initiate_rmsg>Request was processed successfully.</ap_initiate_rmsg>
        <ap_initiate.reason_code>100</ap_initiate.reason_code>
        <merchant_ref_number>TC55255-1</merchant_ref_number>

 <request_token>AxjnrwSTaYwg6/GulJCIAF8mxZM2jViyZtGrZu4cMGDhw4UafAybOyzQfIR+Qybf/Y
uLFOEnpNpjCDr8a6UkIgAAuQ1n</request_token>
        <ap_initiate_trans_ref_no>12345123456788008888</ap_initiate_trans_ref_no>
    </ics_message>
</c:reserved>
</c:replyMessage>
```

NVP Example: Initiate Sale Using the Simple Order API

Request

```
requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206"
apInitiateService_run=true
merchantID=test_merchantID
merchantReferenceCode=demo_merchant
billTo_firstName=John
billTo_lastName=Smith
billTo_street1=201 S. Division St.
billTo_street2=Unit 11
billTo_city=Ann Arbor
billTo_state=MI
billTo_postalCode=98005
billTo_country=KR
billTo_phoneNumber=408-557-1045
billTo_email=jsmith@example.com
purchaseTotals_currency=KRW
purchaseTotals_grandTotalAmount=1100
installment_frequency=0
otherTax_greenTaxAmount=10
apPaymentType=ACC
apInitiateService_productName=Television
apInitiateService_productDescription=Television
apInitiateService_reconciliationID=12345123456788OO888
apInitiateService_escrowAgreement=Y
apInitiateService_languageInterface=Korean
ap_appUrl=gmailLite://abc
```

Response

```
replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.206"
merchantReferenceCode=TC55255-1
requestID=6669371940396961403016
decision=ACCEPT
reasonCode=100
requestToken=AxjnrwSTaYwg6/GulJCIAF8mxZM2jViyZtGrZu4cMGDhw4UafAybOyzQfIR+Qybf/YuLF
OEnpNpjCDr8a6UkIgAAuQ1n
apInitiateReply_reasonCode=100
apInitiateReply_merchantURL=https://
testpay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp?
apInitiateReply_reconciliationID=ca308be3db8742ba9daef4326f21a511
ics_messsage="urn:schemas-cybersource-com:transaction-data:ics"
ics_message_ics_return_code=1000000
ics_message_ics_rcode=1
ics_message_rflag=SOK
ics_message_ap_initiate_rcode=1
ics_message_request_id=6669371940396961403016
ics_message_ap_initiate_rflag=SOK
ics_message_ics_rmsg=Request was processed succesfully
ics_message_ap_initiate_return_code=1900000
ics_message_ap_initiate_merchant_url=https://
testpay.kcp.co.kr/Pay/module/cyberSource/KCP_Gateway.jsp?AppUrl=http%3A%2F
%2FgmailLite://
abc&amp;amp;buyr_mail=null%40cybersource.com&amp;amp;buyr_name=John+SMITH&amp;amp;
buyr_tel=9999999999&amp;amp;currency=410&amp;amp;eng_flag=N&amp;amp;escw_used=N&am
p;amp;good_mny=100&amp;amp;good_name=Television&amp;amp;noti_url=https%3A%2F
%2Fwww.cybersource.com%2Fnotify%2Fkcp&amp;amp;ordr_idxx=12345123456788008888&amp;am
p;pay_method=100000000000&amp;amp;quotaopt=0&amp;amp;ret_url=http%3A%2F
%2Fkcp.cybersource.com%2Fkcp_return.jsp&amp;amp;signature=n%2Fj7WnkAIBVcwA2mwMOhBH
gEGMhnbcYk8lQtt%2F7Sx5GnlD%2FImuhZAU0K8KR8MmNR%0D
%0AwzTF9mziNDbpu6XB2VDEDe4tnrPeK%2Buwul6gtm3cIU%2BT3hfgQlx3%2BvG2dLw67dB3%0D%0AIec
2T0fRCht06S2p%2Ft44Rv%2FEZ%2FkosmfYd%2F2kSMJDASkAzyUUs3ZxX9BT37QKwYBu%0D
%0Au67zWLk0RWhVXk5Ksq0C%2F9KG2Pb5AVrEt9PYhRzaQHrxrcYXPRVT2ULx8QoFInG2%0D%0A
%2Bbwt%2B4Jn5Uo7fD56Mo%2BwfJMi2A9LYXBsnjDhAt3FiV%2BWrTBpI7BqJaAapf1H5%2Fpk%0D
%0AY67vgHbZ2uma57AJS4FM3Q%3D
%3D&amp;amp;site_cd=S6314&amp;amp;site_name=CyberSource
ics_message_ics_decision_reason_code=100
ics_message_ap_initiate_rmsg=Request was processed succesfully
ics_message_ap_initiate=100
ics_message_merchant_ref_number=TC55255
ics_message_request_token=AxjnrwSTaYwg6/GulJCIAF8mxZM2jViyZtGrZu4cMGDhw4UafAybOyzQ
fIR+Qybf/YuLFOEnpNpjCDr8a6UkIgAAuQ1n
ics_message_ap_initiate_trans_ref_no=1234512345678800888
```

Check Status

When the customer is redirected to your website after making a payment, you can request the status of the payment. Verifying a payment status is a follow-on transaction that uses the request ID returned from the initiate payment request. The initiate payment request ID links the check status request to the payment transaction.

> **Important:** Cybersource recommends that you request only the check status service when you receive the return URL from KCP. If you do not receive a return URL, wait 1 hour before requesting the check status service, then request it at hourly intervals.

Fields Specific to this Use Case

These API fields are required specifically for this use case.

apCheckStatusService_apInitiateRequestID

Set value to the value of the request ID returned in the payment response.

apCheckStatusService_run

Set value to `true`.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

Endpoint

Set the `apCheckStatusService_run` field to `true`.

Send the request to `https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`.

Required Fields for a Check Status Using the Simple Order API

Use these fields for a check status request.

apCheckStatusService_apInitiateRequestID

Set value to the value of the request ID returned in the payment response.

apCheckStatusService_run

Set value to `true`.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

merchantID

merchantReferenceCode

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Optional Fields for a Check Status Using the Simple Order API

Use these optional fields for a check status request.

billTo_email

billTo_firstName

billTo_lastName

billTo_phoneNumber

installment_frequency

invoiceHeader_merchantDescriptor

invoiceHeader_merchantDescriptorCity

invoiceHeader_merchantDescriptorContact

invoiceHeader_merchantDescriptorCountry

invoiceHeader_merchantDescriptorPostalCode

invoiceHeader_merchantDescriptorState

invoiceHeader_merchantDescriptorStreet

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Simple Order API Example: Check Status Request

Use this example as a reference for a successful check status request and response.

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206">
<merchantID>kcp_lvmh2</merchantID>
<merchantReferenceCode>TC_auth-1</merchantReferenceCode>
<clientLibrary>Java XML</clientLibrary>
<clientLibraryVersion>5.0.2</clientLibraryVersion>
<clientEnvironment>Linux/3.10.0-1160.66.1.el7.x86_64/Oracle
 Corporation/1.8.0_331</clientEnvironment>
<invoiceHeader>
    <merchantDescriptor>World largest online store</merchantDescriptor>
    <merchantDescriptorContact>Fancy front office</merchantDescriptorContact>
    <merchantDescriptorStreet>101 tcpip blvd</merchantDescriptorStreet>
    <merchantDescriptorCity>Boston</merchantDescriptorCity>
    <merchantDescriptorState>MA</merchantDescriptorState>
    <merchantDescriptorPostalCode>021081234</merchantDescriptorPostalCode>
    <merchantDescriptorCountry>US</merchantDescriptorCountry>
</invoiceHeader>
<installment>
    <frequency>0</frequency>
</installment>
<apPaymentType>NVP</apPaymentType>
<apCheckStatusService run="true">
    <apInitiateRequestID>6669375511836962803016</apInitiateRequestID>
</apCheckStatusService>
</requestMessage>
```

Response

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.206">
<c:merchantReferenceCode>TC_auth-1</c:merchantReferenceCode>
<c:requestID>6669377695686963003016</c:requestID>
<c:decision>ACCEPT</c:decision>
<c:reasonCode>100</c:reasonCode>
<c:requestToken>AxjnrwSTaYw1XlmOtH6IAF8mxZM2jViyZtGrZu4cMGDhywUafAybUSzQfIR+Qybf/Y
uLFDSipNpjC2cJlszTUgAA/BOX</c:requestToken>
<c:apCheckStatusReply>
    <c:reasonCode>100</c:reasonCode>
    <c:reconciliationID>1234512345678800890</c:reconciliationID>
    <c:paymentStatus>COMPLETED</c:paymentStatus>
    <c:processorTransactionID>22931979729374</c:processorTransactionID>
    <c:processorResponse>0000</c:processorResponse>
</c:apCheckStatusReply>
<c:reserved>
    <ics_message xmlns="urn:schemas-cybersource-com:transaction-data:ics">
        <ics_return_code>1000000</ics_return_code>
        <ap_check_status_rflag>SOK</ap_check_status_rflag>
        <ics_rmsg>Request was processed successfully.</ics_rmsg>
        <ap_check_status_rcode>1</ap_check_status_rcode>
        <ap_check_status_payment_status>COMPLETED</ap_check_status_payment_status>

 <request_token>AxjnrwSTaYw1XlmOtH6IAF8mxZM2jViyZtGrZu4cMGDhywUafAybUSzQfIR+Qybf/Y
uLFDSipNpjC2cJlszTUgAA/BOX</request_token>

 <ap_check_status_processor_transaction_id>22931979729374</ap_check_status_process
or_transaction_id>
        <request_id>6669377695686963003016</request_id>
        <ap_check_status_return_code>1900000</ap_check_status_return_code>
        <merchant_ref_number>TC_auth-1</merchant_ref_number>
        <ap_check_status.reason_code>100</ap_check_status.reason_code>
        <ics_decision_reason_code>100</ics_decision_reason_code>
        <ap_check_status_rmsg>Request was processed
 successfully.</ap_check_status_rmsg>
<ap_check_status_processor_response>0000</ap_check_status_processor_response>
        <ics_rflag>SOK</ics_rflag>
<ap_check_status_trans_ref_no>1234512345678800890</ap_check_status_trans_ref_no>
    <ics_rcode>1</ics_rcode>
    </ics_message>
</c:replyMessage>
```

NVP Example: Check Status Request Using the Simple Order API

Request

```
requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206"
merchantID=kcp_lvmh2
merchantReferenceCode=TC_auth-1
invoiceHeader_merchantDescriptor=World largest online store
invoiceHeader_merchantDescriptorContact=Fancy front office
invoiceHeader_merchantDescriptorStreet=101 tcpip blvd
invoiceHeader_merchantDescriptorCity=Boston
invoiceHeader_merchantDescriptorState=MA
invoiceHeader_merchantDescriptorPostalCode=021081234
invoiceHeader_merchantDescriptorCountry=US
installment_frequency=0
apPaymentType=NVP
apCheckStatusService_apInitiateRequestID=6669375511836962803016
```

Response

```
replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.206"
merchantReferenceCode=demo_merchantID
requestID=3771840623730181553725
decision=ACCEPT
reasonCode=100
requestToken=AxjnrwSTaYw1XlmOtH6IAF8mxZM2jViyZtGrZu4cMGDhywUafAybUSzQfIR+Qybf/YuLF
DSipNpjC2cJlszTUgAA/BOX
apCheckStatusReply_reasonCode=100
apCheckStatusReply_reconciliationID=ca308be3db8742ba9daef4326f21a511
apCheckStatusReply_paymentStatus=COMPLETED
apCheckStatusReply_processorTransactionID>20130530123456
apCheckStatusReply_processorResponse=0000
ics_messsage="urn:schemas-cybersource-com:transaction-data:ics"
ics_message_ics_return_code=1000000
ics_message_rflag=SOK
ics_messsage_rmsg=Request was processed succesfully.
ics_message_ap_checkstatus_rcode=1
ics_message_ap_check_status_payment_status=COMPLETED
ics_message_request_token=AxjnrwSTaYwg6/GulJCIAF8mxZM2jViyZtGrZu4cMGDhw4UafAybOyzQ
fIR+Qybf/YuLFOEnpNpjCDr8a6UkIgAAuQ1n
ics_message_ap_check_status_processor_transaction_id=22931979729374
ics_message_request_id=6669377695686963003016
ics_message_ap_check_status_return_code=1900000
ics_message_merchant_ref_number=TC_auth-1
ics_message_ap_check_status.reason_code=100
ics_message_ics_decision_reason_code=100
ics_message_ap_check_status_rmsg=Request was processed successfully.
ap_check_status_processor_response=0000
ap_check_status_processor_response_ics_rflag=SOK
ap_check_status_trans_ref_no=1234512345678800890
ap_check_status_trans_ref_no_ics_rcode=1
```

Check Status Testing Information

In the Cybersource test environment, you can simulate the specific error messages that you receive from transaction requests. By including certain values in your transaction requests, you can generate specific responses in the transaction reply messages. The simulated environment allows you to become familiar with the reply messages and develop methods for error handling.

Simple Order API Test Responses

To simulate a test, set the **apInitiateService_reconciliationID** field to a value listed in the Trigger Value column. The Cybersource response is returned in the **apCheckStatusReply_paymentStatus** field.

Check Status Trigger Values

Trigger Value	Processor Response Code	Cybersource Response Message	Processor Payment Status Code
The last digit of the transaction reference number is `0`.	`0000`	`PENDING`	`STAU`
The last digit of the transaction reference number is `1`.	`0000`	`DECLINED`	`STAF`
The last digit of the transaction reference number is `2`.	`0000`	`COMPLETED`	`STSC`
The last digit of the transaction reference number is `3`.	`0000`	`COMPLETED`	`STPC`
The last digit of the transaction reference number is `4`.	`0000`	`TRADE_NOT_EXIST`	`STXX`
The last digit of the transaction reference number is `5`.	`0000`	`CANCELLED`	`STSR`
The last digit of the transaction reference number is `6` or `7` or `8` or `9`.	`0000`	`COMPLETED`	`STSR`

Refund

A refund is a follow-on transaction that uses the request ID returned from the initiate sale request. The request ID links the refund transaction to the sale transaction.

> ! **Important:** Refunds are permitted up to sixty-days after the transaction is initiated. Cybersource recommends the sale status be `COMPLETED` before you refund a payment and return funds to the customer account. Before refunding a payment, request the check status service to retrieve the status of the sale.

Refund Restrictions

Multiple partial refunds are allowed. Partial refunds are not allowed for mobile billing payments or escrow payments. Partial refunds are not allowed when using the **otherTax_greenTaxAmount** field. Refunds using the API service are not allowed for escrow payments.

Fields Specific to this Use Case

These API fields are required specifically for this use case.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

apRefundService_apInitiateRequestID

Set value to the value of the request ID returned in the sale response.

apRefundService_run

Set value to `true`.

purchaseTotals_currency

Set value to `KRW`.

Endpoint

Set the `apRefundService_run` field to `true`.

Send the request to `https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`.

Required Fields for a Refund Using the Simple Order API

Use these fields for a refund request.

apPaymentType

Set value to either `ACC`, `KKP`, `MBP`, `NVP`, `PYC`, or `RBT`.

apRefundService_apInitiateRequestID

Set value to the value of the request ID returned in the sale response.

apRefundService_run

Set value to `true`.

merchantID

merchantReferenceCode

purchaseTotals_currency

Set value to `KRW`.

purchaseTotals_grandTotalAmount

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Optional Fields for a Refund Using the Simple Order API

Use these optional fields for a refund.

apRefundService_reason

billTo_email

billTo_firstName

billTo_lastName

billTo_phoneNumber

installment_frequency

otherTax_ greenTaxAmount

You cannot use this field for partial refunds.

purchaseTotals_taxAmount

Related information

Simple Order API Request Fields *(on page 42)*

API Field Reference for the Simple Order API

Simple Order API Example: Refund Request

Use this example as a reference for a successful refund request and response.

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206">
<merchantID>acc_kcp_mid</merchantID>
<merchantReferenceCode>TC_auth-1</merchantReferenceCode>
<clientLibrary>Java XML</clientLibrary>
<clientLibraryVersion>5.0.2</clientLibraryVersion>
<clientEnvironment>Linux/3.10.0-1160.66.1.el7.x86_64/Oracle
 Corporation/1.8.0_331</clientEnvironment>
<purchaseTotals>
    <currency>KRW</currency>
    <grandTotalAmount>1000</grandTotalAmount>
</purchaseTotals>
<installment>
    <frequency>0</frequency>
</installment>
<apPaymentType>PYC</apPaymentType>
<apRefundService run="true">
    <reason>Damaged Goods</reason>
    <apInitiateRequestID>6669534585326084103016</apInitiateRequestID>
</apRefundService>
</requestMessage>
```

Response

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.206">
<c:merchantReferenceCode>TC_auth-1</c:merchantReferenceCode>
<c:requestID>6669536290416091203016</c:requestID>
<c:decision>ACCEPT</c:decision>
<c:reasonCode>102</c:reasonCode>
<c:invalidField>c:grandTotalAmount</c:invalidField>
<c:requestToken>AxjjrwSTaY5oz2prpAXIAF9Rp8DKEZbNCvoI/UZtv/sXFiJ0WUm0xzFgUpu6jbQA
MCm+</c:requestToken>
<c:apRefundReply>
    <c:reasonCode>102</c:reasonCode>
</c:apRefundReply>
<c:reserved>
    <ics_message xmlns="urn:schemas-cybersource-com:transaction-data:ics">
        <invalidfield0>grand_total_amount</invalidfield0>
        <merchant_ref_number>TC_auth-1</merchant_ref_number>
        <request_id>6669536290416091203016</request_id>
        <ap_refund_rcode>0</ap_refund_rcode>
        <ics_rcode>0</ics_rcode>
        <ap_refund_return_code>1271000</ap_refund_return_code>
        <ap_refund_rflag>DINVALIDDATA</ap_refund_rflag>
        <ics_decision_reason_code>102</ics_decision_reason_code>
        <ap_refund_rmsg>The following request field(s) is either invalid or
 missing: grand_total_amount</ap_refund_rmsg>
        <ap_refund.reason_code>102</ap_refund.reason_code>
        <ics_rflag>DINVALIDDATA</ics_rflag>
        <ics_return_code>1000000</ics_return_code>

 <request_token>AxjjrwSTaY5oz2prpAXIAF9Rp8DKEZbNCvoI/UZtv/sXFiJ0WUm0xzFgUpu6jbQA
MCm+</request_token>
        <ics_rmsg>The following request field(s) is either invalid or missing:
 grand_total_amount</ics_rmsg>
        <parsed_field_names/>
    </ics_message>
</c:replyMessage>
```

NVP Example: Refund Request Using the Simple Order API

Request

```
requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.206"
merchantID=acc_kcp_mid
merchantReferenceCode=TC_auth-1
purchaseTotals_currency=KRW
purchaseTotals_grandTotalAmount=11
installment_frequency=0
apPaymentType=PYC
apRefundService_run=true
apRefundService_reason=Damaged goods
apRefundService_apInitiateRequestID=3770131092530181553725
```

Response

```
merchantReferenceCode=demo_merchantID
requestID=3771840623730181553725
decision=ACCEPT
reasonCode=100
purchaseTotals_currency=KRW
apRefundReply_reasonCode=100
apRefundReply_amount=11
apRefundReply_dateTime=2013-08-22T15:07:47Z
apRefundReply_processorTransactionID=20130530123456
merchantReferenceCode=TC_auth-1
requestID=6669536290416091203016
decision=ACCEPT
reasonCode=102
requestToken=AxjjrwSTaY5oz2prpAXIAF9Rp8DKEZbNCvoI/UZtv/sXFiJ0WUm0xzFgUpu6jbQAMCm+
apCheckStatusReply_reasonCode=102
ics_messsage="urn:schemas-cybersource-com:transaction-data:ics"
ics_message_merchant_ref_number=TC_auth-1
ics_message_request_id=6669377695686963003016
ics_message_rcode=0
ics_message_ap_refund_code=1271000
ics_message_ap_refund_rflag=DINVALIDDATA
ics_message_ics_decision_reason_code=102
ics_message_ap_refund_rmsg=The following request field(s) is either invalid or
 missing: grand_total_amount
ics_message_ap_refund.reason_code=102
ics_message_ics_rflag_DINVALIDDATA
ics_message_ics_return_code=1000000
ics_message_request_token=AxjjrwSTaY5oz2prpAXIAF9Rp8DKEZbNCvoI/UZtv/sXFiJ0WUm0xzFg
Upu6jbQAMCm+
ics_rmsg=The following request field(s) is either invalid or missing:
 grand_total_amount
```

Refund Testing Information

In the Cybersource test environment, you can simulate the specific error messages that you receive from transaction requests. By including certain values in your transaction requests, you can generate specific responses in the transaction reply messages. The simulated environment allows you to become familiar with the reply messages and develop methods for error handling.

Simple Order API Test Responses

To simulate a test, set the **apInitiateService_reconciliationID** field to a value listed in the Trigger Value column.

Refund Trigger Values

Cybersource Trigger Value	Processor Response Code	Cybersource Response
Any value excluding the below values.	`0000`	`success`
`110115230001`	`1000`	`Missing mandatory value`
`110115230002`	`2000`	`Invalid value, format error`
`110115230003`	`3000`	`System Error`
`110115230004`	`C000`	`Please contact the card issuer for further assistance.`
`110115230005`	`A000`	`Please contact your bank for further assistance.`
`110115230006`	`M000`	`Mobile billing is not available for your carrier plan.`
`110115230007`	`4000`	`New error`
`110115230008` or `110115230009` or `110115230010`	`0000`	`success`
`110115230011`	`2000`	`The request exceeds refund amount.`

Reference Information

This section contains reference information that is useful when using KCP.

Simple Order API Reason Codes

The **reasonCode** field contains additional data regarding the decision response of the transaction.

Reason Codes and Decisions

Reason Code	Decision
100	ACCEPT
101	REJECT
102	REJECT
150	ERROR

Simple Order API Field Descriptions

This section describes the Simple Order API fields mentioned in this guide.

For more information on the Simple Order API fields, see API Field Reference for the Simple Order API

Data Type Definitions

For more information about these data types, see the World Wide Web Consortium (W3C) XML Schema Part 2: Datatypes specification.

Data Type	Description
Date and Time	Format is YYYY-MM-DDThhmmssZ, where: • T separates the date and time • Z indicates Coordinated Universal Time (UTC).
Decimal	Number that includes a decimal point.
Integer	Whole number {..., -3, -2, -1, 0, 1, 2, 3, ...}
String	Sequence of letters, numbers, spaces, and special characters

Special Characters

Do not use the following special characters to initiate a sale. See Initiate Sale (on page 15).

Comma	Ampersand	Semicolon	New line	Backslash	Pipe line	Single Quotation Mark	Double Quotation Mark
,	&	;	\n	\	|	‘	“

Working with Multibyte Character Strings

Many international languages, such as Korean, Chinese, and Russian, require more than one ASCII byte per character. Therefore, a string of multibyte characters can exceed the allotted string length of some the Simple Order API fields.

For example, the **billTo_firstName** and **billTo_lastName** fields can each handle a string of 15 ASCII characters. When the first and last name fields are used together, they can handle 30 ASCII characters in total, including the blank space between the names.

First name + space + last name = 30 (maximum)

In many international languages, a short name that visually has only 3 or 4 characters may be equivalent to 10 or 15 bytes in UTF-8. Therefore, it is important to take into consideration the overall length of a multibyte string to be sure it fits into the allotted ASCII string length.

Important: For best success when working with international languages, make sure the total size of a multibyte string fits within the ASCII string length for a specific API field.

Simple Order API Request Fields

This section describes the request fields.

Related information

API Field Reference for the Simple Order API

apCheckStatusService_apInitiateRequestID

Identifier returned from the initiate payment service request.

Services

- **Check Status:** Required
- **Refund:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 26

Mapping Information

- **REST API Field:** No corresponding field. The service is requested with the `{id}` value in these endpoints:
  - Check Status: `https://api.cybersource.com/v2/status/{id}/refresh`
  - Refund: `https://api.cybersource.com/v2/captures/{id}/refunds`
- **SCMP API Field: ap_initiate_request_id**
- **Simple Order API Field:**
  - Check Status: **apCheckStatusService_apInitiateRequestID**
  - Refund: **apRefundService_apInitiateRequestID**

apCheckStatusService_run

Flag that specifies whether to include the check status service in the request.

Possible values:

- `true`: Include the check status service in the request.
- `false`: Do not include the check status service in the request. Default value.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field. The service is requested with the endpoint: `https://api.cybersource.com/v2/status/{id}/refresh`
- **SCMP API Field and Value: ics_applications**`=ics_ap_check_status`
- **Simple Order API Field: apCheckStatusService_run**

apInitiateService_escrowAgreement

Indicates whether to use the customer’s escrow agreement.

The escrow agreement can be used only if the transaction is greater than or equal to 50,000 KRW. You must enter the shipping details in the KCP administration panel for escrow payments.

Primarily used for real estate transactions.

> ❗ **Important:** Available only for bank transfer payments.

Possible values:

- `true`: Use the customer’s escrow agreement.
- `false`: Do not use the customer’s escrow agreement.

Services

- **Authorization:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Field: processingInformation.enableEscrowOption**
- **SCMP API Field: ap_initiate_escrow_agreement**
- **Simple Order API Field: apInitiateService_escrowAgreement**

apInitiateService_languageInterface

Language indicator for Korea Cyber Payment (KCP) services.

Possible values:

- `EN`: English
- `Korean`: Korean (default)

Services

- **Authorization:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 6

Mapping Information

- **REST API Field: buyerInformation.language**
- **SCMP API Field: ap_initiate_language_interface**
- **Simple Order API Field: apInitiateService_languageInterface**

apInitiateService_productDescription

Detailed description of item.

Services

- **Initiate Sale:** Optional

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 65

Mapping Information

- **REST API Field: orderInformation.lineItems.productDescription**
- **SCMP API Field: ap_initiate_product_description**
- **Simple Order API Field: apInitiateService_productDescription**

apInitiateService_productName

Product offered to the customer.

Services

- **Authorization:** Optional

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 30

Mapping Information

- **REST API Field: orderInformation.lineItems.productName**
- **SCMP API Field: ap_initiate_product_name**
- **Simple Order API Field: apInitiateService_productName**

apPaymentType

Three-digit value that indicates the payment type.

Possible values:

- `ACC`: Credit card payment.
- `AFP`: Afterpay and Clearpay.
- `KKP`: KakaoPay.
- `LPY`: LINE Pay.
- `MBP`: Mobile billing payment.
- `NVP`: Naver Pay.
- `PPY`: PayPay.
- `PYC`: Payco.
- `RBT`: Real-time bank transfer.
- `RPY`: Rakuten Pay.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 3

Mapping Information

- **REST API Field: paymentInformation.paymentType.method.name**
- **SCMP API Field: ap_payment_type**
- **Simple Order API Field: apPaymentType**

apRefundService_apInitiateRequestID

Identifier returned from the initiate payment service request.

Services

- **Check Status:** Required
- **Refund:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 26

Mapping Information

- **REST API Field:** No corresponding field. The service is requested with the `{id}` value in these endpoints:
  - Check Status: `https://api.cybersource.com/v2/status/{id}/refresh`
  - Refund: `https://api.cybersource.com/v2/captures/{id}/refunds`
- **SCMP API Field: ap_initiate_request_id**
- **Simple Order API Field:**
  - Check Status: **apCheckStatusService_apInitiateRequestID**
  - Refund: **apRefundService_apInitiateRequestID**

apRefundService_run

Flag that specifies whether to include the credit service in a request.

Possible values:

- `true`: Include the credit service in the request.
- `false`: Do not include the credit service in the request. Default value.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field. The service is requested with the endpoint: `https://api.cybersource.com/pts/v2/payments/{id}/refunds`
- **SCMP API Field and Value: ics_applications**`=ics_ap_refund`
- **Simple Order API Field: apRefundService_run**

ap_appUrl

App URL used by Korea Cyber Payment (KCP) to wake up the merchant app after payment processing.

Only alphanumeric characters are accepted.

Format: `{a-z}://`

Services

- **Sale:** Optional

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 512

Mapping Information

- **REST API Field: deviceInformation.appUrl**
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: ap_appUrl**

apInitiateService_returnURL

URL of the web page to which the customer is directed when the transaction is completed.

This URL overrides the return URL that was specified during merchant boarding.

Services

- **Authorization:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 256

Mapping Information

- **REST API Field: merchantInformation.returnUrl**
- **SCMP API Field: ap_initiate_return_url**
- **Simple Order API Field: apInitiateService_returnURL**

ap_refund_reason

Reason for the refund.

Services

- **Refund:** Optional

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 256

Mapping Information

- **REST API Field: processingInformation.refundOptions.reason**
- **SCMP API Field: ap_refund_reason**
- **Simple Order API Field: apRefundService_reason**

apInitiateService_run

Flag that specifies whether to include the initiate sale service in the request.

Possible values:

- `true`: Include the service in your request.
- `false`: Do not include the service in your request. Default value.

Services

- **Initiate Sale:** Required

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field. The service is requested with the endpoint: `https://api.cybersource.com/pts/v2/payments`
- **SCMP API Field and Value: ics_applications**`=ics_ap_initiate`
- **Simple Order API Field: apInitiateService_run**

billTo_city

City in the billing address.

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 50

Mapping Information

- **REST API Field: orderInformation.billTo.locality**
- **SCMP API Field: bill_city**
- **Simple Order API Field: billTo_city**

billTo_country

Country in the billing address.

Use two-character ISO Standard Country Codes.

Tax Calculation Service

- **U.S and Canadian Tax**: Required
- **International Tax and Value-Added Tax (VAT)**: Required

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 2

Mapping Information

- **REST API Field: orderInformation.billTo.country**
- **SCMP API Field: bill_country**
- **Simple Order API Field: billTo_country**

billTo_email

Customer's email address, including full domain name.

Format: name@host.domain

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 255

Mapping Information

- **REST API Field: orderInformation.billTo.email**
- **SCMP API Field: customer_email**
- **Simple Order API Field: billTo_email**

billTo_firstName

Customer name.

Value should match value on card.

Cybersource Latin American Processing

> **Important:**
>
> For an authorization request, Cybersource Latin American Processing concatenates **billTo_firstName** and **billTo_lastName** . If the concatenated value exceeds 30 characters, Cybersource Latin American Processing declines the authorization request.
>
> Cybersource Latin American Processing is the name of a specific processing connection. *Cybersource Latin American Processing* does not refer to the general topic of processing in Latin America. The information in this field description, or in this section of the field description, is for the specific processing connection called Cybersource Latin American Processing. It is not for any other Latin American processors.

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Cybersource Latin American Processing**
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** See field description
- **Worldpay VAP**
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** 25

Mapping Information

- **REST API Field: orderInformation.billTo.firstName**
- **SCMP API Field: customer_firstname**
- **Simple Order API Field: billTo_firstName**

billTo_lastName

Customer’s last name.

Value should match value on card.

Cybersource Latin American Processing

> **Important:**
>
> For an authorization request, Cybersource Latin American Processing concatenates **billTo_firstName** and **billTo_lastName** . If the concatenated value exceeds 30 characters, Cybersource Latin American Processing declines the authorization request.
>
> Cybersource Latin American Processing is the name of a specific processing connection. *Cybersource Latin American Processing* does not refer to the general topic of processing in Latin America. The information in this field description, or in this section of the field description, is for the specific processing connection called Cybersource Latin American Processing. It is not for any other Latin American processors.

Specifications

- **Cybersource Latin American Processing:**
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** See field description
- **Worldpay VAP:**
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** 25

Mapping Information

- **REST API Field: orderInformation.billTo.lastName**
- **SCMP API Field: customer_lastname**
- **Simple Order API Field: billTo_lastName**

billTo_phoneNumber

Customer's phone number.

Include the country code when the order is from outside the U.S.

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

For installment payments with Mastercard in Brazil, the value for this field corresponds to the following data in the TC 33 capture file:

- Record: CP07 TCR4
- Position: 40-50
- Field: Buyer Phone Number

The TC 33 capture file contains information about the payments and credits that a merchant submits to Cybersource. The processor creates the TC 33 capture file at the end of the day and sends it to the merchant’s acquirer. The acquirer uses this information to facilitate end-of-day clearing processing with payment networks.

Specifications

- **Visa Platform Connect for Installment Payments with Mastercard in Brazil:**
    - **Field Type:** Request
    - **Type:** String
    - **Length:** 11
- **Other Processors:**
    - **Field Type:** Request
    - **Type:** String
    - **Length:** 15

Mapping Information

- **REST API Field: orderInformation.billTo.phoneNumber**
- **SCMP API Field: customer_phone**
- **Simple Order API Field: billTo_phoneNumber**

billTo_postalCode

Postal code in the billing address.

The postal code must consist of five to nine digits.

When the billing country is the U.S., the nine-digit postal code must follow this format: [5 digits][dash][4 digits].

Example: 12345-6789

When the billing country is Canada, the six-digit postal code must follow this format: [alpha][numeric][alpha][space][numeric][alpha][numeric]

Example: A1B 2C3

American Express Direct

Before the postal code is sent to the processor, all non-alphanumeric characters are removed, and if the remaining value is longer than nine characters, the value is truncated starting from the right side.

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Comercio Latino**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 9
- **Visa Platform Connect**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 9

Mapping Information

- **REST API Field: orderInformation.billTo.postalCode**
- **SCMP API Field: bill_zip**
- **Simple Order API Field: billTo_postalCode**

billTo_state

State or province in the billing address.

For the U.S., Canada, and Mainland China, use the ISO 3166-2 format for two-character state, province, or territory codes. For the U.S. and Canada, see State, Province, and Territory Codes for the United States and Canada.

For all countries, use the ISO 3166-2 format when using this field for Payer Authentication.

Visa Platform Connect

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitations of the payment card networks prevent Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 2

Mapping Information

- **REST API Field: orderInformation.billTo.administrativeArea**
- **SCMP API Field: bill_state**
- **Simple Order API Field: billTo_state**

billTo_street1

First line of the billing street address as it appears in the payment card issuer records.

FDMS Nashville

When the street name is numeric, it must be sent in numeric format. For example, if the address is *One First Street*, it must be sent as *1 1st Street*.

Visa Platform Connect

> **Important:**
>
> When you populate billing street address 1 and billing street address 2, Visa Platform Connect concatenates the two values. When the concatenated value exceeds 40 characters, Visa Platform Connect truncates the value at 40 characters before sending it to Visa and the issuing bank. Truncating this value affects AVS results and therefore might also affect risk decisions and charge backs.

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitation of the payment card networks prevents Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Chase Paymentech Solutions**:
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** 20
- **Moneris**:
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** 50
- **Visa Platform Connect**:

    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 40
- **Worldpay VAP**:
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 35

Mapping Information

- **REST API Field: orderInformation.billTo.address1**
- **SCMP API Field: bill_address1**
- **Simple Order API Field: billTo_street1**

billTo_street2

Additional address information.

Example: Attention: Accounts Payable

Chase Paymentech Solutions, FDC Compass, and TSYS Acquiring Solutions

This value is used for AVS.

FDMS Nashville

The billing address 1 and billing address 2 fields together cannot exceed 20 characters.

Visa Platform Connect

> **Important:**
>
> When you populate billing street address 1 and billing street address 2, Visa Platform Connect concatenates the two values. When the concatenated value exceeds 40 characters, Visa Platform Connect truncates the value at 40 characters before sending it to Visa and the issuing bank. Truncating this value affects AVS results and therefore might also affect risk decisions and charge backs.

Credit card networks cannot process transactions that contain non-ASCII characters. Visa Platform Connect accepts and stores non-ASCII characters correctly and displays them correctly in reports. However, the limitation of the payment card networks prevents Visa Platform Connect from transmitting non-ASCII characters to the payment card networks. Therefore, Visa Platform Connect replaces non-ASCII characters with meaningless ASCII characters for transmission to the payment card networks.

Specifications

- **Chase Paymentech Solutions:**
  - **Field Type:** Request
  - **Data Type:** String
  - **Data Length:** 20
- **Moneris**

    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 50
- **Visa Platform Connect**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 40
- **Worldpay VAP**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 35

Mapping Information

- **REST API Field: orderInformation.billTo.address2**
- **SCMP API Field: bill_address2**
- **Simple Order API Field: billTo_street2**

installment_frequency (Request)

Frequency of the installment payments.

This field is supported only on Visa Platform Connect.

Possible values for standing-instruction MITs with Diners Club or Mastercard in India or with an India-issued card:

- `1`: Daily (for an installment payment or recurring payment).
- `2`: Weekly (for an installment payment or recurring payment).
- `3`: Fortnightly (for an installment payment or recurring payment).
- `4`: Monthly (for an installment payment or recurring payment).
- `5`: Quarterly (for an installment payment or recurring payment).
- `6`: Half-yearly (for an installment payment or recurring payment).
- `7`: Annually (for an installment payment or recurring payment).
- `8`: As needed (for an unscheduled COF transaction).

Possible values for other kinds of installment payments:

- `B`: Biweekly.
- `M`: Monthly.
- `W`: Weekly.

Crediario Installment Payments

When you do not include this field in a request for a Credario installment payment, a space character is sent to the processor.

The value for this field corresponds to the following data in the TC 33 capture file:

- Record: CP01 TCR9
- Position: 41
- Field: Installment Frequency

The TC 33 capture file contains information about the payments and credits that a merchant submits to Cybersource. The processor creates the TC 33 capture file at the end of the day and sends it to the merchant's acquirer. The acquirer uses this information to facilitate end-of-day clearing processing with payment networks.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 1

Mapping Information

- **REST API Field: installmentInformation.frequency**
- **SCMP API Field: installment_frequency**
- **Simple Order API Field: installment_frequency**

invoiceHeader_merchantDescriptor

Your business name.

This name is displayed on the cardholder's statement. When you include more than one consecutive space, extra spaces are removed.

Important: The value must consist of English characters.

Payouts on Chase Paymentech Solutions

Use one these formats:

- <12-character merchant name>*<9-character product description>
- <7-character merchant name>*<14-character product description>
- <3-character merchant name>*<18-character product description>

Payouts on FDC Compass and Visa Platform Connect

For a credit card bill payment, set the value for this field to the name of the originator providing the credit card bill payment service.

For a funds disbursement, set the value for this field to the name of the originator sending the funds disbursement.

For a prepaid load, set the value for this field to the name of the partner providing the reload service.

Payouts on FDC Nashville Global

You must be enabled as a government controlled merchant in the CyberSource system with the attribute CFG : <proc>_governmentControlled.

PIN Debits

When you do not include this value in a PIN debit request, Cybersource uses the merchant name in your account.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 23

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.name** This field requires the country name (2-Char ISO enum).
- **SCMP API Field: merchant_descriptor**
- **Simple Order API Field: invoiceHeader_merchantDescriptor**

invoiceHeader_merchantDescriptorCity

City for your business location.

This value might be displayed on the cardholder’s statement.

When you do not include this value in your PIN debit request, Cybersource uses the merchant city from your account.

> **Important:** The value must consist of English characters.

Payouts

For an OCT transaction, the only supported value is `Visa Direct`.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 13

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.locality**
- **SCMP API Field: merchant_descriptor_city**
- **Simple Order API Field: invoiceHeader_merchantDescriptorCity**

invoiceHeader_merchantDescriptorContact

Contact information for your business.

This value must be the city in which your store or outlet is located.

When you include more than one consecutive space, extra spaces are removed.

This value might be displayed on the cardholder’s statement.

Payouts

For Chase Paymentech Solutions, you must use one of the following formats:

- PCCCCCCCCCCCC
- NNN-NNN-NNNN
- NNN-NNN-NAAA
- NNN-NNN-AAAA
- NNN-AAAAAAA

where:

- A: Alphanumeric (alpha or numeric)
- C: Character (alpha or blank)
- N: Numeric
- P: Alpha

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 13

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.contact**
- **SCMP API Field: merchant_descriptor_contact**
- **Simple Order API Field: invoiceHeader_merchantDescriptorContact**

invoiceHeader_merchantDescriptorCountry

Country code for your business location.

Use the standard ISO Standard Country Codes.

Important: This value must consist of English characters.

This value might be displayed on the cardholder’s statement.

When you do not include this value in your request, Cybersource uses the merchant country from your account.

If your business is located in the U.S. or Canada and you include this field in a request, you must also include **invoiceHeader_merchantDescriptorState**.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 2

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.country**
- **SCMP API Field: merchant_descriptor_country**
- **Simple Order API Field: invoiceHeader_merchantDescriptorCountry**

invoiceHeader_merchantDescriptorPostalCode

Postal code for your business location.

Important: This value must consist of Engligh characters.

Important: Mastercard requires a postal code for any country that uses postal codes. You can provide the postal code in your Cybersource account or you can include this field in your request.

This value might be displayed on the cardholder's statement.

If your business is domiciled in the U.S., you can use a 5-digit or 9-digit postal code. A 9-digit postal code must follow this format: [5 digits][dash][4 digits] **Example:** 12345-6789

If your business is domiciled in Canada, use a 6-digit postal code. A 6-digit postal code must follow this format: [alpha][numeric][alpha][space] [numeric][alpha][numeric] **Example:** A1B 2C3

When you do not include this value in your PIN debit request, Cybersource uses the merchant postal code from your account.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 14

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.postalCode**
- **SCMP API Field: merchant_descriptor_postal_code**
- **Simple Order API Field: invoiceHeader_merchantDescriptorPostalCode**

invoiceHeader_merchantDescriptorState

State code or region code for your business location.

Use the standard state, province, and territory codes for the United States and Canada.

This field is supported only for businesses located in the U.S. or Canada.

Important: This value must consist of Engligh characters.

This value might be displayed on the cardholder’s statement.

When you do not include this value in your PIN debit request, Cybersource uses the merchant state from your account.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 2

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.administrativeArea**
- **SCMP API Field: merchant_descriptor_state**
- **Simple Order API Field: invoiceHeader_merchantDescriptorState**

invoiceHeader_merchantDescriptorStreet

Street address for your business location.

Important: This value must consist of English characters.

This value might be displayed on the cardholder’s statement.

When you include this value in your request, Cybersource recommends you also include the merchant descriptor country, merchant descriptor state, and merchant descriptor postal code in your request.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:**
  - Visa Platform Connect: 29
  - All other processors: 60

Mapping Information

- **REST API Field: merchantInformation.merchantDescriptor.address1**
- **SCMP API Field: merchant_descriptor_street**
- **Simple Order API Field: invoiceHeader_merchantDescriptorStreet**

merchantID

Your merchant ID.

Use the same merchant ID for evaluation, testing, and production.

Chase Paymentech Solutions

This field is supported for Payouts transactions only. It is not supported for standard credit card transactions. It is optional for Mastercard and Visa transactions.

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 30

Mapping Information

- **REST API Field:** No corresponding field.
- **SCMP API Field: merchant_id**
- **Simple Order API Field: merchantID**

merchantReferenceCode (Request)

Order reference or tracking number that you generate.

Provide a unique value for each transaction so that you can perform meaningful searches for the transaction. Refer to the *Getting Started with Cybersource Advanced* guide.

Card-Not-Present Transactions on FDC Nashville Global

Certain circumstances can cause the processor to truncate this value to 15 or 17 characters for Level II and Level III processing, which can cause a discrepancy between the value you submit and the value included in some processor reports.

Card-Present Transactions on FDC Nashville Global

When you do not send a valid value, Cybersource creates one for you. However, the value is not returned to you, so you cannot use the merchant reference number to track the order.

Card-Present Transactions on SIX

When a card-present request does not include a merchant reference number, this value is provided by the client software that is installed on the POS terminal.

PIN Debit Transactions

Requests for PIN debit reversals must include the same merchant reference number that was used in the transaction that is being reversed.

Specifications

- **Asia, Middle East, and Africa Gateway**:
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 40
- **China UnionPay**:
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 12
- **Elavon Americas**:

    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 39
- **FDC Nashville Global Card-Present Transactions:**
    - **Field Type:** Request
    - **Data Type:** Numeric
    - **Data Length:** 8
- **RuPay Credit Services:**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 24
- **RuPay Other Services:**
    - **Field Type:** Request
    - **Data Type:** String
    - **Data Length:** 50

Mapping Information

- **REST API Field: clientReferenceInformation.code**
- **SCMP API Field: merchant_ref_number**
- **Simple Order API Field: merchantReferenceCode**

otherTax_greenTaxAmount

Tax amount paid to government for green initiative.

Services

- **Sale:** Optional

Specifications

- **Field Type:** Request
- **Data Type:** Integer
- **Data Length:** 12

Mapping Information

- **REST API Field and Value: orderInformation.amountDetails.taxDetails[].type**`=green`
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: otherTax_greenTaxAmount**

purchaseTotals_currency (Request)

Currency used for the order.

For possible values, refer to ISO Standard Currency Codes.

For authorization reversal or capture services, you must use the same currency that was used for the authorization.

For the PIN debit reversal service, you must use the same currency that was used for the PIN debit purchase or PIN debit credit that you are reversing.

DCC with a Third-Party Provider

Customer’s billing currency.

Visa Platform Connect

For Mastercard installment payments in Peru, the value for this field corresponds to the following data in the TC 33 capture file:

- Record: CP01 TCR0
- Position: 108-110
- Field: Currency

Specifications

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Fields:**
  - **orderInformation.amountDetails.currency**
  - **reversalInformation.amountDetails.currency**
- **SCMP API Field: currency**
- **Simple Order API Field: purchaseTotals_currency**

purchaseTotals_grandTotalAmount

Grand total for the order.

If your request includes line items, do not include this field in your request.

This value cannot be negative. You can include a decimal point (.), but you cannot include any other special characters. The amount is truncated to the correct number of decimal places.

> ❗ **Important:** Some processors have specific requirements and limitations, such as maximum amounts and maximum field lengths.

Dynamic Currency Conversions

When this field is used in a request with Dynamic Currency Conversion, this field value must be denominated in the customer's billing currency.

Original Credit Transactions (OCT)

The amount must be less than or equal to 50,000 USD. Amount limits differ depending on limitations imposed by your acquirer in addition to local laws and regulations. Contact your acquirer for more information.

PIN Debit Transactions

If the transaction includes a cash-back amount, that amount must be included in this total amount. If the transaction includes a surcharge amount, that amount must be included in this total amount.

Zero Amount Authorizations

If your processor supports zero amount authorizations, you can set this field to `0` for the authorization to verify whether the card is lost or stolen.

Specifications

Comercio Latino

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 19

Other Processors

- **Field Type:** Request
- **Data Type:** String
- **Data Length:** 15

Mapping Information

- **REST API Fields:**
    - **orderInformation.amountDetails.totalAmount**
    - **reversalInformation.amountDetails.totalAmount**
- **SCMP API Field: grand_total_amount**
- **Simple Order API Field: purchaseTotals_grandTotalAmount**

Processor Field Names

The following list provides the Level II/Level III processor field name for each processor that supports Level II or Level III data.

- **Chase Paymentech Solutions:** N/A
- **FDC Compass:** N/A
- **FDC Nashville Global:** N/A
- **OmniPay Direct:** Line Item Total
- **RBS WorldPay Atlanta:** Amount
- **TSYS Acquiring Solutions:** N/A

purchaseTotals_taxAmount

Total tax amount for all the items.

Specifications

American Express Direct

- **Field Type:** Request
- **Data Type:** Decimal
- **Data Type:** String
- **Data Length for Canadian dollars:** 7
- **Data Length for U.S. dollas:** 9

Mapping Information

- REST API Field: **orderInformation.amountDetails.taxAmount**
- SCMP API Field: **total_tax_amount**
- Simple Order API Field: **purchaseTotals_taxAmount**

Processor Field Names

The following list provides the Level II/Level III processor field name for each processor that supports Level II or Level III data:

- American Express Direct: N/A

Simple Order API Response Fields

This section describes the response fields.

Related information

API Field Reference for the Simple Order API

apCheckStatusReply_paymentStatus

Payment status returned from the payment processor.

Possible values:

- `Pending`: Request received and waiting to be processed.
- `Settled`: Payment successfully processed.
- `Failed`: Payment failed.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 15

Mapping Information

- **REST API Field:** To be released soon
- **SCMP API Field: ap_check_status_payment_status**
- **Simple Order API Field: apCheckStatusReply_paymentStatus**

apCheckStatusReply_processorTransactionID

Transaction ID number generated by Korea Cyber Payment (KCP).

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 14

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field: ap_check_status_processor_transaction_id**
- **Simple Order API Field: apCheckStatusReply_processorTransactionID**

apCheckStatusReply_reasonCode

Numeric value that corresponds to the result of the check status request.

Specifications

- **Field Type:** Response
- **Data Type:** Integer
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: apCheckStatusReply_reasonCode**

apCheckStatusReply_reconciliationID

Reference number for the transaction.

You can use this value to reconcile Cybersource reports with processor reports.

This value is not returned for all processors.

Specifications

- **Field Type:** Response
- **Data Type:** Integer
- **Data Length:** 60

Mapping Information

- **REST API Field: clientReferenceInformation.reconciliationId**
- **SCMP API Field: ap_check_status_trans_ref_no**
- **Simple Order API Field: apCheckStatusReply_reconciliationID**

apInitiateReply_merchantURL

Redirect URL to the Korea Cyber Payment (KCP) plug-in. The URL has encrypted data that contains a unique reference ID and payment details for the order.

> **Important:** Some browsers restrict browser URL lengths, so the customer should be redirected to this URL using the code described in.

Services

- **Sale**

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** Encrypted

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field: ap_initiate_merchant_url**
- **Simple Order API Field: apInitiateReply_merchantURL**

apInitiateReply_reasonCode

Numeric value corresponding to the result of the `apInitiateService_run` request.

Services

- **Sale**

Specifications

- **Field Type:** Response
- **Data Type:** Integer
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: apInitiateReply_reasonCode**

apInitiateReply_reconciliationID or apInitiateService_reconciliationID

Reference number for the transaction.

> **Important:** For most Cybersource services, the response message includes a unique reconciliation ID that is assigned by Cybersource. For most payment processors, you can use this value to reconcile the transactions in your Cybersource reports with the transactions in your processor reports.
>
> For details, such as specific field names and information about processors that do not support reconciliation, see the implementation guide or developer guide for the payment method you are using.

Services

- **Sale**

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 32

Mapping Information

- **REST API Field: clientReferenceInformation.reconciliationId**
- **SCMP API Field: ap_initiate_trans_ref_no**
- **Simple Order API Field:**
    - **apInitiateReply_reconciliationID**
    - **apInitiateService_reconciliationID**

apRefundReply_amount

Authorized amount.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 15

Mapping Information

- **REST API Field:** To be released soon
- **SCMP API Field: ap_refund_amount**
- **Simple Order API Field: apRefundReply_amount**

apRefundReply_dateTime

Date and time when the service was requested.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 20

Mapping Information

- **REST API Field:** To be released soon
- **SCMP API Field: ap_refund_date_time**
- **Simple Order API Field: apRefundReply_dateTime**

apRefundReply_processorTransactionID

Transaction ID number generated by Korea Cyber Payment (KCP).

Services

- **Refund**

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 14

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field: ap_refund_processor_transaction_id**
- **Simple Order API Field: apRefundReply_processorTransactionID**

apRefundReply_reasonCode

Numeric value that corresponds to the result of the refund request.

Specifications

- **Field Type:** Response
- **Data Type:** Integer
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: apRefundReply_reasonCode**

decision

Summary of the result of the overall request.

Possible values:

- `ACCEPT`: Request succeeded.
- `ERROR`: System error occurred.
- `REJECT`: One or more of the service requests were declined.
- `REVIEW`: The order was flagged for review. This value is returned only when you use Decision Manager.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 6

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: decision**

merchantReferenceCode (Response)

Order reference or tracking number that you provided in the request.

If you included multi-byte characters in this field in the request, the returned value might include corrupted characters.

FDC Nashville Global

When a card-present request does not include a merchant reference number, this value is provided by the client software that is installed on the POS terminal.

Sometimes the processor truncates this value to 15 or 17 characters for Level II and Level III processing. This can cause a discrepancy between the value you submit and the value included in some processor reports.

SIX

When a card-present request does not include a merchant reference number, this value is provided by the client software that is installed on the POS terminal.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 50

Mapping Information

- **REST API Field: clientReferenceInformation.code**
- **SCMP API Field: merchant_ref_number**
- **Simple Order API Field: merchantReferenceCode**

purchaseTotals_currency (Response)

Currency used in the transaction.

Use a three-character ISO currency code. For a list of ISO currency codes, see: https://developer.cybersource.com/docs/cybs/en-us/currency-codes/reference/all/na/currency-codes/currency-codes.htmlISO Currency Codes

For an authorization reversal or capture, you must use the same currency that was used for associated authorization.

When using Dynamic Currency Conversion (DCC) with a Third-Party Provider:, use the customer's billing currency.

Visa Platform Connect

For Mastercard installment payments in Peru, the value for this field corresponds to the following data in the TC 33 capture file:

- Record: CP01 TCR5
- Position: 108-110
- Field: Financing Currency

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 5

Mapping Information

- **REST API Fields:**
    - **orderInformation.amountDetails.currency**
    - **creditAmountDetails.currency**
    - **refundAmountDetails.currency**
    - **reversalAmountDetails.currency**
- **SCMP API Field: currency**
- **Simple Order API Field: purchaseTotals_currency**

reasonCode

Numeric value that corresponds to the result of the overall request.

Specifications

- **Field Type:** Response
- **Data Type:** Integer
- **Data Length:** 5

Mapping Information

- **REST API Field:** No corresponding field
- **SCMP API Field:** No corresponding field
- **Simple Order API Field: reasonCode**

requestID

Request identifier that the client software generates.

Specifications

- **Field Type:** Response
- **Data Type:** String
- **Data Length:** 256

Mapping Information

- **REST API Field:** No corresponding field. The value is returned in the endpoint of the resource that was requested.
- **SCMP API Field: request_id**
- **Simple Order API Field: requestID**