Korean Cyber Payment Services

Using the Simple Order API

August 2019

CyberSource®
the power of payment

CyberSource Corporation HQ | P.O. Box 8999 | San Francisco, CA 94128-8999 | Phone: 800-530-9095

CyberSource Contact Information

For general information about our company, products, and services, go to http://www.cybersource.com.

For sales questions about any CyberSource Service, email sales@cybersource.com or call 650-432-7350 or 888-330-2300 (toll free in the United States).

For support information about any CyberSource Service, visit the Support Center at http://www.cybersource.com/support.

Copyright

© 2019 CyberSource Corporation. All rights reserved. CyberSource Corporation ("CyberSource") furnishes this document and the software described in this document under the applicable agreement between the reader of this document ("You") and CyberSource ("Agreement"). You may use this document and/or software only in accordance with the terms of the Agreement. Except as expressly set forth in the Agreement, the information contained in this document is subject to change without notice and therefore should not be interpreted in any way as a guarantee or warranty by CyberSource. CyberSource assumes no responsibility or liability for any errors that may appear in this document. The copyrighted software that accompanies this document is licensed to You for use only in strict accordance with the Agreement. You should read the Agreement carefully before using the software. Except as permitted by the Agreement, You may not reproduce any part of this document, store this document in a retrieval system, or transmit this document, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written consent of CyberSource.

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 Unpublished rights reserved under the copyright laws of the United States.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, CyberSource Payment Manager, CyberSource Risk Manager, CyberSource Decision Manager, and CyberSource Connect are trademarks and/or service marks of CyberSource Corporation.

All other brands and product names are trademarks or registered trademarks of their respective owners.

Contents

Recent Revisions to This Document 5

About This Guide 6
Audience and Purpose 6
Conventions 6
Important Statement 6
Text and Command Conventions 6
Related Documents 7
Customer Support 7

Chapter 1 Introduction 8
Payment Types 8
KCP Services 9
Initiate Payment Service Sequence 9
Merchant URL 11
Check Status Service 13
Refund Service 13

Chapter 2 Requesting KCP Services 14
Initiating Payment 14
Checking Payment Status 15
Refunding a Payment 16

Appendix A API Fields 17
Data Type Definitions 17
Special Characters 17
Working with Multibyte Character Strings 18
Request Fields 18
Reply Fields 21
Reason Codes 23

Appendix B Examples 24

Name-Value Pair Examples **24**

XML Examples **27**

Recent Revisions to This Document

Release	Changes
August 2019	Added information about multibyte character strings. See "Working with Multibyte Character Strings," page 18.
September 2018	Updated the “Initiate Payment Service” section. See "Initiate Payment Service Sequence," page 9. Updated the “Merchant URL” section. See "Merchant URL," page 11. Updated the “Initiate Payment” reply examples with encrypted string. See Appendix B, "Examples," on page 24.
April 2015	Updated the Important note regarding supported browsers and mobile devices. See the important note in Chapter 1, "Introduction," on page 8.
January 2014	Updated the “Related Documents” section. See "Related Documents," page 7.Added the **apCheckStatusReply_ processorTransactionID** API reply field. See page 21.
October 2013	Initial release.

About This 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

Important Statement

Important

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

Text and Command Conventions

Convention	Usage
bold	▪ Field and service names in text; for example: Include the **requestID** field. ▪ Items that you are instructed to act upon; for example: Click **Save**.
`screen text`	▪ XML elements. ▪ Code examples and samples.

Related Documents

- *Getting Started with CyberSource Advanced for the Simple Order API*—describes how to start using your CyberSource advanced account.
- *Business Center Overview*—describes the features and options available with your CyberSource advanced account using the Business Center.
- Simple Order API and SOAP Toolkit API Documentation and Downloads page.

Refer to the Support Center for complete CyberSource technical documentation:

http://www.cybersource.com/support_center/support_documentation

Customer Support

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

http://www.cybersource.com/support

CHAPTER 1

Introduction

Important All KCP payments are processed in South Korean Won (KRW). Contact KCP for a list of supported desktop browsers and mobile devices.

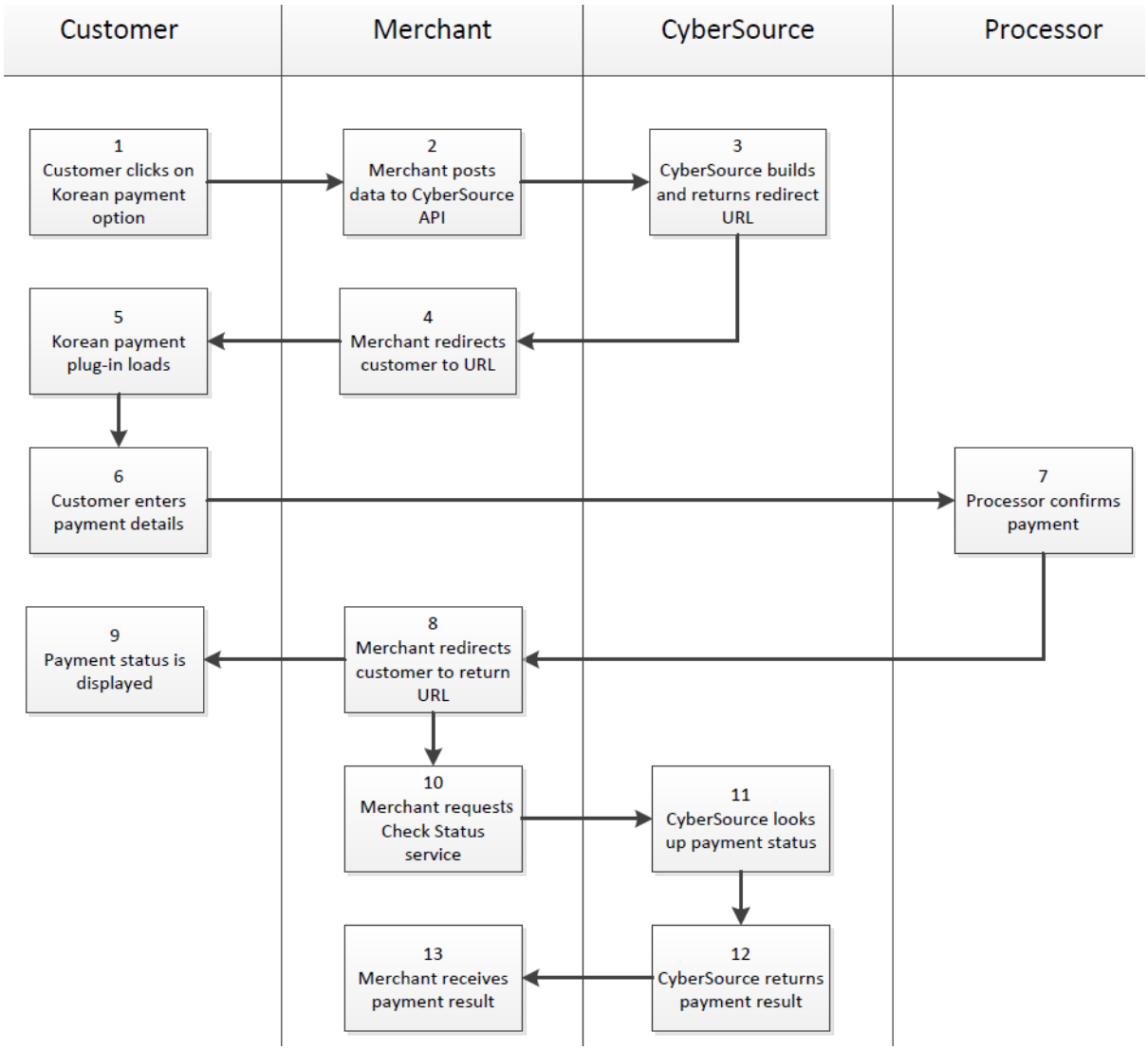
Payment Types

Table 1 Payment Types

Payment Type	Description
ACC	Credit card payment.
RBT	Real-time bank transfer.
MBP	Mobile billing payment.

KCP Services

Initiate Payment Service Sequence



1 The customer chooses the Korean payment type on your website's checkout or payment page.

2 Using the CyberSource API, you POST the merchant details and order information from the initiate payment service request to CyberSource.

3 CyberSource creates a signed and encrypted merchant URL (see "Merchant URL") to redirect the customer to the KCP web site. This signed and encrypted merchant URL is sent back to you.

4 The customer tis redirected to the URL that CyberSource sent you in Step 3. 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.

8 CyberSource stores the payment status. You can retrieve the payment status by requesting the check status service (see "Check Status Service," page 13).

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 (see "Check Status Service," page 13).

10 The merchant requests the check status service from CyberSource. CyberSource recommends that this service be requested immediately after Step 9 and every hour thereafter until the payment status is COMPLETED, DECLINED, or CANCELLED.

11 CyberSource verifies the payment status with the processor.

12 CyberSource sends the payment result to the merchant.

13 The merchant receives 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

The merchant URL consists of the following parameters.

Table 2 Merchant URL Parameters

Parameter	Description
KCP web site	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 blob of data.

The encrypted data described in Table 2 consists of the following parameters.

Table 3 Merchant URL Parameters

Parameter	Description
buyr_mail	Customer email.
buyr_name	Customer name.
buyr_tel	Customer telephone number.
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: ■ Yes: Y ■ No: N **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.
pay_method	Payment method. Possible values: ■ 100000000000—Credit card. ■ 010000000000—Bank transfer. ■ 000010000000—Mobile billing.

Table 3 Merchant URL Parameters (Continued)

Parameter	Description
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.

The merchant URL consists of an encrypted string of characters representing the payment details. The customer should be redirected to this URL. Internet Explorer restricts browser URL lengths, so use the following code when performing the redirect using client-side JavaScript:

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

$apInitiateReply->merchantURL is the merchant URL included in the CyberSource response to the Initiate request.

Note

A server-side redirect can also be used to perform the redirect.

Check Status Service

When the customer is redirected to your web site 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. See "Checking Payment Status," page 15.

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.

Refund Service

A refund is a follow-on transaction that uses the request ID returned from the initiate payment request. The request ID links the refund transaction to the payment transaction. See "Refunding a Payment," page 16.

Important Refunds are permitted up to 60 days after the transaction is initiated. CyberSource recommends that the payment 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 payment.

CHAPTER 2

Requesting KCP Services

Initiating Payment

To initiate a payment:

Step 1 Set the **apInitiateService_run** field to `true`.

Step 2 Include the following fields in the request:

- billTo_firstName
- billTo_lastName
- billTo_phoneNumber
- billTo_email
- apInitiateService_returnURL
- apInitiateService_productName
- apInitiateService_escrowAgreement
- apInitiateService_languageInterface—the default is `Korean`.
- installment_frequency—this field is 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.
- apPaymentType
- purchaseTotals_currency
- purchaseTotals_grandTotalAmount
- merchantID
- merchantReferenceCode

For detailed descriptions of all request and reply fields, see Appendix A, "API Fields," on page 17. For an initiate payment example, see Appendix B, "Examples," on page 24.

Checking Payment Status

To check the status of a payment:

Step 1 Set the **apCheckStatusService_run** field to `true`.

Step 2 Include the following fields in the request:

- apCheckStatusService_apInitiateRequestID—include the value returned in the initiate payment service reply. See "Reply Fields," page 21.
- apPaymentType
- merchantID
- merchantReferenceCode

For detailed descriptions of all request and reply fields, see Appendix A, "API Fields," on page 17. For a check status example, see Appendix B, "Examples," on page 24.

Refunding a Payment

Important Multiple partial refunds are allowed. Partial refunds are not allowed for mobile billing payments or escrow payments. Refunds using the API service are not allowed for escrow payments.

To refund a payment:

Step 1 Set the **apRefundService_run** field to `true`.

Step 2 Include the following fields in the request:

- apRefundService_requestID—include the value returned in the initiate payment service reply. See "Reply Fields," page 21.
- apPaymentType
- apRefundService_reason
- purchaseTotals_currency
- purchaseTotals_grandTotalAmount
- merchantID
- merchantReferenceCode

For detailed descriptions of all request and reply fields, see Appendix A, "API Fields," on page 17. For a refund example, see Appendix B, "Examples," on page 24.

API Fields

APPENDIX A

See *Getting Started with CyberSource Advanced for the Simple Order API* for a description of how name-value pair names relate to their corresponding XML element names.

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 in the Initiate Payment request (see page 14):

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 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 alloted 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.

Request Fields

Table 4 Request-Level for the Simple Order API

Field Name	Description	Used By: Required (R) or Optional (O)	Data Type & Length
apCheckStatusService_ apInitiateRequestID	Identifier returned from the initiate payment service request.	Check Status (R)	String (32)
apCheckStatusService_run	The check status service. Possible values: ■ true: include the service in your request. ■ false (default): do not include the service in your request.	Check Status (R)	String (5)

Table 4 Request-Level for the Simple Order API (Continued)

Field Name	Description	Used By: Required (R) or Optional (O)	Data Type & Length
apInitiateService_ escrowAgreement	Indicates whether to use the customer's escrow agreement. Possible values: ■ Y: yes, use the customer's escrow agreement. ■ N: no, do not use the customer's escrow agreement. **Note** 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. **Important** Available only for bank transfer payments.	Initiate Payment (R)	String (1)
apInitiateService_ languageInterface	Language indicator for the KCP plug-in. Possible values: ■ EN: English ■ `Korean` (default)	Initiate Payment (R)	String (1)
apInitiateService_productName	Product offered to the customer.	Initiate Payment (O)	String (30)
apInitiateService_returnURL	URL of the web page to which customer is directed when the transaction is completed. **Note** This URL overrides the return URL that was specified during merchant boarding.	Initiate Payment (R)	String (256)
apInitiateService_run	The initiate payment service. Possible values: ■ true: include the service in your request. ■ false (default): do not include the service in your request.	Initiate Payment (R)	
apPaymentType	Identifier for payment type. Possible values: ■ ACC: Credit Card Payment. ■ RBT: Real Time Bank Transfer. ■ MBP: Mobile Billing Payment.	Initiate Payment (R) Check Status (R) Refund (R)	String (5)

Table 4 Request-Level for the Simple Order API (Continued)

Field Name	Description	Used By: Required (R) or Optional (O)	Data Type & Length
apRefundService_ apInitiateRequestID	Identifier returned from the initiate payment service request.	Refund (R)	
apRefundService_reason	Reason for the refund.	Refund (O)	String (256)
apRefundService_run	The refund service. Possible values: ■ true: include the service in your request. ■ false (default): do not include the service in your request.	Refund (R)	
billTo_email	Customer's email address.	Initiate Payment (O) Check Status (O) Refund (O)	String (50)
billTo_firstName	Customer first name. **Note** The first and last name fields together cannot exceed 30 ASCII characters, including spaces. In UTF-8, the first and last name fields together cannot exceed an equivalent number of multibyte characters, including spaces.	Initiate Payment (O) Check Status (O) Refund (O)	String (15)
billTo_lastName	Customer last name. **Note** The first and last name fields together cannot exceed 30 ASCII characters, including spaces. In UTF-8, the first and last name fields together cannot exceed an equivalent number of multibyte characters, including spaces.	Initiate Payment (O) Check Status (O) Refund (O)	String (15)
billTo_phoneNumber	Customer phone number.	Initiate Payment (O) Check Status (O) Refund (O)	String (20)
installment_frequency	Frequency of payments. Possible values are 0 to 12. Default value: 12. **Note** This field is for credit card payments only.	Initiate Payment (O)	String (2)
merchantID	Your CyberSource merchant ID. Use the same merchant ID for evaluation, testing, and production.	Initiate Payment (R) Check Status (R) Refund (R)	String (50)

Table 4 Request-Level for the Simple Order API (Continued)

Field Name	Description	Used By: Required (R) or Optional (O)	Data Type & Length
merchantReferenceCode	Merchant-generated order reference or tracking number. CyberSource recommends that you send a unique value for each transaction so that you can perform meaningful searches for the transaction. For more information about tracking orders, see *Getting Started with CyberSource Advanced for the Simple Order API.*	Initiate Payment (R) Check Status (R) Refund (R)	String (50)
purchaseTotals_currency	Currency used for the order. Value: KRW	Initiate Payment (R) Check Status (R) Refund (R)	String (3)
purchaseTotals_ grandTotalAmount	Grand total for the order. **Important** The **grand_total_ amount** value should not contain a decimal point. For example, 210.00 should be 21000.	Initiate Payment (R) Refund (R)	Decimal (12)

Reply Fields

Table 5 Reply Fields for the Simple Order API

Field Name	Description	Returned By	Data Type & Length
apCheckStatusReply_ paymentStatus	Description of the payment status. Possible values: ■ PENDING ■ COMPLETED ■ DECLINED ■ CANCELLED	Check Status	String (15)
apCheckStatusReply_ processorTransactionID	KCP-generated transaction ID number.	Check Status	String (14)
apCheckStatusReply_ reasonCode	Numeric value corresponding to the result of the **apCheckStatusService_ run** request. See "Reason Codes," page 23.	Check Status	Integer (5)

Table 5 Reply Fields for the Simple Order API (Continued)

Field Name	Description	Returned By	Data Type & Length
apCheckStatusReply_ reconciliationID	Reference number for the transaction. For more information about tracking orders, see *Getting Started with CyberSource Advanced for the Simple Order API*.	Initiate Payment	String (32)
apInitiateReply_merchantURL	Redirect URL to the KCP plug in. The URL has encrypted data that contains a unique reference and payment details for the order. **Important** Internet Explorer restricts browser URL lengths, so the customer should be redirected to this URL using the code described on page 11.	Initiate Payment	Encrypted string of characters.
apInitiateReply_reasonCode	Numeric value corresponding to the result of the **apInitiateService_run** request.	Initiate Payment	Integer (5)
apInitiateReply_reconciliationID	Reference number for the transaction. For more information about tracking orders, see *Getting Started with CyberSource Advanced for the Simple Order API*.	Initiate Payment	String (32)
apRefundReply_amount	The total refund amount.	Refund	Decimal (15)
apRefundReply_dateTime	Local date and time of the refund. Format: YYYY-MM-DDThhmmssZ, where: ■ T separates the date and time ■ Z indicates Coordinated Universal Time (UTC).	Refund	Date and Time (14)
apRefundReply_ processorTransactionID	KCP-generated transaction ID number.	Refund	String (14)
apRefundReply_reasonCode	Numeric value corresponding to the result of the **apRefundService_run** request.	Refund	Integer (5)
decision	One-word description of the request result.	Initiate Payment Check Status Refund	String (50)

Table 5 Reply Fields for the Simple Order API (Continued)

Field Name	Description	Returned By	Data Type & Length
merchantReferenceCode	Merchant-generated order reference or tracking number. CyberSource recommends that you send a unique value for each transaction so that you can perform meaningful searches for the transaction. For more information about tracking orders, see *Getting Started with CyberSource Advanced for the Simple Order API.*	Initiate Payment Check Status Refund	String (50)
purchaseTotals_currency	Currency used for the order. Value: KWR		String (10)
reasonCode	Numeric value corresponding to the result of the request. See "Reason Codes," page 23.	Initiate Payment Check Status Refund	Integer (5)
requestID	Identifier for the request.	Initiate Payment Check Status Refund	String (32)

Reason Codes

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

Table 6 Reason Codes and Decisions

Reason Code	Decision
100	ACCEPT
101	REJECT
102	REJECT
150	ERROR

APPENDIX B

Examples

Name-Value Pair Examples

Example **Initiate Payment Request**

```
apInitiateService_run=true
merchantID=test_merchantID
merchantReferenceCode=demo_merchant
billTo_firstName=John
billTo_lastName=Smith
billTo_phoneNumber=408-557-1045
billTo_email=jsmith@example.com
purchaseTotals_currency=KRW
purchaseTotals_grandTotalAmount=1100
installment_frequency=0
apPaymentType=ACC
apInitiateService_returnURL=https://www.example.com
apInitiateService_productName=Television
apInitiateService_productDescription=Television
apInitiateService_escrowAgreement=Y
apInitiateService_languageInterface=EN
```

Example Initiate Payment Reply

```
merchantReferenceCode=demo_merchant
requestID=377183721333018155372 5
decision=ACCEPT
reasonCode=100
apInitiateReply_reasonCode=100
apInitiateReply_merchantURL=https://testpay.kcp.co.kr/Pay/module/cyberSource/KCP_
Gateway.jsp?encrypted_
data=QSOopVlTlwil6MIZshfzt2KIgoEckdwS%2FzFWmpYy0SLztaG%2FQDGOvCq2d98SbuZp%0D%0Al74o4G
KtR0sAVtr%2Fkdq1%2FfBLyh89KG3WupixaM09wy98BSAOXxau4hvx6gn4FMRF%0D%0AludYnfKPgP2Jr8Kav
7gNUodRA0Ulp8o1R3WgX1udjDvyIHhOfDjkcPSaXjFx4wHv%0D%0AfxGqH9LBihw4mNl3fNm6rLF5ndKC2TLA
wW%2B3njzB1SXG9sOvtT3ZBN0srNc8uxkd%0D%0AS06j74RhdrA95z74VIcvRh5N%2BARlQjy%2FNex0SBtBE
jH4Rt9WaHdbkNhreqQauq7E%0D%0AB5GA7cPaBly52kXCr6pqyHGcFvRW9rDT69jlg%2BYaxxjTCe8qLyjXKy
iltyi46K3J%0D%0ATR7wQFmccptvxgc0EPCCTQu6XFeKxSBVh%2BJjVXyMxsn%2FgUdb1DbLlmyqJXnT3aXw%
0D%0ANkHST0GedE05CjCv55opLAds5T3ZhPM6EtfnWUiyVRC3BCzskWtbLugAlZetA%2B4R%0D%0AywLK80y5
LjlvkVlqAFTCx1Nm5bvgcBce6qPYP%2FMKGcPcb5gVpTujXV7CCgS1y7nz%0D%0AKU71h6Fiis3kzpGqJDBwz
PTwtS2X07RM7kE4PNoBSbHVF3Rnkz6iNvpEgsA%2Fpj0R%0D%0APS1gd9Qh0OwEYrpZ%2FAD2fcIeU7F%2F51
3iF75h6acv9HUloQKAsvDdoukvVQpVlQeo%0D%0A%2FoIOMK7NVPid%2Flwg5RKySXaDSZvf6J0NGpxz8aZGc
GXq2KAq8DQy78HwSR735v62%0D%0A%2FbBhypJweXvE19WS9MS0Xa7YKMcOziITEH8yv2cckpvFrIgEISdqIv
LbnE3P6cxV%0D%0ApDUQIeWuPoNjG8RdDpJx1ztb2XetBD9Y68Lfq4nPodw7MdWA6C%2BT%2BFoCPPDhzKxe%
0D%0ATLHkoo7GGlE030qivqhE5eJnB5rZm9JGiitWUbaMYwDtdP85YGxixiJAU8XtWiin%0D%0ASbziau8m0v
MUB4hVKvnHdrCgYI1Grzngjz3iCqxDmsFeaTwRRkW%2BoC0dXY5AQhRN%0D%0ATgdvvqG8uNBKtpr8KRptH8d
7f9AjWKetQeKEr5y2G5wwE0GUpzdEHYmLxzEyxsAt%0D%0A0sRohXGxgpY1qdh6y3YxtFGo7JifeOOZzFDcns
%2B%2FTPp79WoR%2FQAhtm4AgcDZ16ZI%0D%0A6kre2VYkOuNhuLmTc7QaZxtoyMN5kbty%2FkQusaYYSmEwi
Qb%2BhFaHUWz8tHGQT%2Bb9%0D%0ALY5ScYSAtoImp7osWcQFa4DwETtBMMUqhix1zVDFxlMZmOyPsAdovnsX
3wGnNw%2BB%0D%0AvvHjnHhR9q7T4MTlm%2FYYA9qY18%2F%2FSvL%2BGXBdVZJBb8S%2BwpL7PP08mDVnSD9
yl1jk%0D%0ABQ53PE%2B1hwRbo6v4gDAK%2Bse049dscasIn8T2natsDG5LaL%2FW3ZrfNEmnhCgK2Apg%0D%
0At01RH9zRB5%2Fo89c6z3CM0lc9zPiq4eW0%2FIirjc0j8xgMjadSvTjOCrobENKebCAM%0D%0A8gJl%2FE8
9z0NZclBQ5WhFxeS8iHO%2Btpu54iB0T9g5n5ELTiA92pbCRiAA%2FHScwPlc%0D%0A1PlegfH9XaXmplZQan
TWDg1u6qBil9ZZGXyg82emcqqhFV5atSmMTZC9S26XOQO%2B%0D%0AGqs8a27q3Xi6PvJQpklW10kaF1Ow388
NkYH0IHo%3D
apInitiateReply_reconciliationID=ca308be3db8742ba9daef4326f21a511
```

Example Check Status Request

```
apCheckStatusService_run=true
merchantID=test_merchantID
merchantReferenceCode=demo_merchant
apPaymentType=ACC
apCheckStatusService_apInitiateRequestID=3770131092530181553725
```

Example Check Status Reply

```
merchantReferenceCode=demo_merchantID
requestID=377184062373018155372 5
decision=ACCEPT
reasonCode=100
apCheckStatusReply_paymentStatus=COMPLETED
apCheckStatusReply_processorTransactionID>20130530123456
apCheckStatusReply_reasonCode=100
apCheckStatusReply_reconciliationID=ca308be3db8742ba9daef4326f21a511
```

Example Refund Request

```
apRefundService_run=true
merchantID=test_merchantID
merchantReferenceCode=demo_merchant
purchaseTotals_currency=KRW
purchaseTotals_grandTotalAmount=11.00
apPaymentType=ACC
apRefundService_apInitiateRequestID=3770131092530181553725
apRefundService_reason=Damaged product.
```

Example Refund Reply

```
merchantReferenceCode=demo_merchantID
requestID=377184062373018155372 5
decision=ACCEPT
reasonCode=100
purchaseTotals_currency=KRW
apRefundReply_reasonCode=100
apRefundReply_amount=11.00
apRefundReply_dateTime=2013-08-22T15:07:47Z
apRefundReply_processorTransactionID=20130530123456
```

XML Examples

Example Initiate Payment Request

```
 <requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-
1.91">
   <merchantID>test_merchantID</merchantID>
   <merchantReferenceCode>demo_merchant</merchantReferenceCode>
   <billTo>
        <firstName>John</firstName>
        <lastName>Smith</lastName>
       <phoneNumber>408-557-1045</phoneNumber>
        <email>jsmith@example.com</email>
   </billTo>
    <purchaseTotals>
        <currency>KRW</currency>
        <grandTotalAmount>11.00</grandTotalAmount>
   </purchaseTotals>
   <installment>
        <frequency>0</frequency>
   </installment>
   <apPaymentType>ACC</apPaymentType>
   <apInitiateService run="true">
        <returnURL>https://www.example.com</returnURL>
        <productName>Television</productName>
        <productDescription>Television</productDescription>
        <escrowAgreement>Y</escrowAgreement>
        <languageInterface>EN</languageInterface>
   </apInitiateService>
  </requestMessage>
```

Example Initiate Payment Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.91">
    <c:merchantReferenceCode>demo_merchant</c:merchantReferenceCode>
    <c:requestID>3771837213330181553725</c:requestID>
    <c:decision>ACCEPT</c:decision>
    <c:reasonCode>100</c:reasonCode>
    <c:apInitiateReply>
        <c:reasonCode>100</c:reasonCode>
        <c:merchantURL>https://testpay.kcp.co.kr/Pay/module/cyberSource/KCP_
        Gateway.jsp?encrypted_
        data=QSOopVlTlwil6MIZshfzt2KIgoEckdwS%2FzFWmpYy0SLztaG%2FQDGOvCq2d98SbuZp%0D%0
        Al74o4GKtR0sAVtr%2Fkdq1%2FfBLyh89KG3WupixaM09wy98BSAOXxau4hvx6gn4FMRF%0D%0Alud
        YnfKPgP2Jr8Kav7gNUodRA0Ulp8o1R3WgX1udjDvyIHhOfDjkcPSaXjFx4wHv%0D%0AfxGqH9LBihw
        4mNl3fNm6rLF5ndKC2TLAwW%2B3njzB1SXG9sOvtT3ZBN0srNc8uxkd%0D%0AS06j74RhdrA95z74V
        IcvRh5N%2BARlQjy%2FNex0SBtBEjH4Rt9WaHdbkNhreqQauq7E%0D%0AB5GA7cPaBly52kXCr6pqy
        HGcFvRW9rDT69jlg%2BYaxxjTCe8qLyjXKyiltyi46K3J%0D%0ATR7wQFmccptvxgc0EPCCTQu6XFe
        KxSBVh%2BJjVXyMxsn%2FgUdb1DbLlmyqJXnT3aXw%0D%0ANkHST0GedE05CjCv55opLAds5T3ZhPM
        6EtfnWUiyVRC3BCzskWtbLugAlZetA%2B4R%0D%0AywLK80y5LjlvkVlqAFTCx1Nm5bvgcBce6qPYP
        %2FMKGcPcb5gVpTujXV7CCgS1y7nz%0D%0AKU71h6Fiis3kzpGqJDBwzPTwtS2X07RM7kE4PNoBSbH
        VF3Rnkz6iNvpEgsA%2Fpj0R%0D%0APS1gd9Qh0OwEYrpZ%2FAD2fcIeU7F%2F513iF75h6acv9HUlo
        QKAsvDdoukvVQpVlQeo%0D%0A%2FoIOMK7NVPid%2Flwg5RKySXaDSZvf6J0NGpxz8aZGcGXq2KAq8
        DQy78HwSR735v62%0D%0A%2FbBhypJweXvE19WS9MS0Xa7YKMcOziITEH8yv2cckpvFrIgEISdqIvL
        bnE3P6cxV%0D%0ApDUQIeWuPoNjG8RdDpJx1ztb2XetBD9Y68Lfq4nPodw7MdWA6C%2BT%2BFoCPPD
        hzKxe%0D%0ATLHkoo7GGlE030qivqhE5eJnB5rZm9JGiitWUbaMYwDtdP85YGxixiJAU8XtWiin%0D
        %0ASbziau8m0vMUB4hVKvnHdrCgYI1Grzngjz3iCqxDmsFeaTwRRkW%2BoC0dXY5AQhRN%0D%0ATgd
        vvqG8uNBKtpr8KRptH8d7f9AjWKetQeKEr5y2G5wwE0GUpzdEHYmLxzEyxsAt%0D%0A0sRohXGxgpY
        1qdh6y3YxtFGo7JifeOOZzFDcns%2B%2FTPp79WoR%2FQAhtm4AgcDZ16ZI%0D%0A6kre2VYkOuNhu
        LmTc7QaZxtoyMN5kbty%2FkQusaYYSmEwiQb%2BhFaHUWz8tHGQT%2Bb9%0D%0ALY5ScYSAtoImp7o
        sWcQFa4DwETtBMMUqhix1zVDFxlMZmOyPsAdovnsX3wGnNw%2BB%0D%0AvvHjnHhR9q7T4MTlm%2FY
        YA9qY18%2F%2FSvL%2BGXBdVZJBb8S%2BwpL7PP08mDVnSD9yl1jk%0D%0ABQ53PE%2B1hwRbo6v4g
        DAK%2Bse049dscasIn8T2natsDG5LaL%2FW3ZrfNEmnhCgK2Apg%0D%0At01RH9zRB5%2Fo89c6z3C
        M0lc9zPiq4eW0%2FIirjc0j8xgMjadSvTjOCrobENKebCAM%0D%0A8gJl%2FE89z0NZclBQ5WhFxeS
        8iHO%2Btpu54iB0T9g5n5ELTiA92pbCRiAA%2FHScwPlc%0D%0A1Plegfh9XaXmplZQanTWDg1u6qB
        il9ZZGXyg82emcqqhFV5atSmMTZC9S26XOQO%2B%0D%0AGqs8a27q3Xi6PvJQpklW10kaF1Ow388Nk
        YH0IHo%3D

        <c:reconciliationID>ca308be3db8742ba9daef4326f21a511</c:reconciliationID>
    </c:apInitiateReply>
</c:replyMessage>
```

Example Check Status Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-
1.91">
    <merchantID>test_merchantID</merchantID>
    <merchantReferenceCode>demo_merchant</merchantReferenceCode>
    <apPaymentType>ACC</apPaymentType>
    <apCheckStatusService run="true">
        <apInitiateRequestID>3770131092530181553725</apInitiateRequestID>
    </apCheckStatusService>
</requestMessage>
```

Example Check Status Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-
1.91">
    <c:merchantReferenceCode>demo_merchantID</c:merchantReferenceCode>
    <c:requestID>3771840623730181553725</c:requestID>
    <c:decision>ACCEPT</c:decision>
    <c:reasonCode>100</c:reasonCode>
    <c:purchaseTotals>
    <c:apCheckStatusReply>
        <c:paymentStatus>COMPLETED</c:paymentStatus>
        <c:processorTransactionID>20130530123456</
        c:processorTransactionID>
        <c:reasonCode>100</c:reasonCode>
        <c:reconciliationID>ca308be3db8742ba9daef4326f21a511</
        c:reconciliationID>
    </c:apCheckStatusReply>
</c:replyMessage>
```

Example Refund Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-
1.91">
    <merchantID>test_merchantID</merchantID>
    <merchantReferenceCode>demo_merchant</merchantReferenceCode>
    <purchaseTotals>
       <currency>KRW</currency>
       <grandTotalAmount>11.00</grandTotalAmount>
    </purchaseTotals>
    <apPaymentType>ACC</apPaymentType>
    <apRefundService run="true">
       <apInitiateRequestID>3770131092530181553725</apInitiateRequestID>
       <reason>Damaged product.</reason>
    </apRefundService>
</requestMessage>
```

Example Refund Reply

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-
1.91">
   <c:merchantReferenceCode>demo_merchantID</c:merchantReferenceCode>
   <c:requestID>3771840623730181553725</c:requestID>
   <c:decision>ACCEPT</c:decision>
   <c:reasonCode>100</c:reasonCode>
   <c:purchaseTotals>
   <c:currency>KRW</c:currency>
   <c:apRefundReply>
      <c:reasonCode>100</c:reasonCode>
      <c:amount>5</c:amount>
      <c:dateTime>2013-08-22T15:07:47Z</c:dateTime>
      <c:processorTransactionID>20130530123456</
      c:processorTransactionID>
   </c:apRefundReply>
</c:replyMessage>
```