

Credit Card Services

Simple Order API

American Express

Developer Guide



© 2022. 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 Unpublished rights reserved under the copyright laws of the United States.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. 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: 22.04

Contents

- Recent Revisions to This Document..... 6**
- About This Guide..... 7**
- Introduction to Payments..... 8**
 - Financial Institutions and Payment Networks..... 8
 - Acquiring (Merchant) Financial Institutions..... 8
 - Issuing (Customer) Financial Institutions..... 9
 - Payment Networks..... 9
 - Payment Processors..... 10
 - Payment Types..... 10
 - Co-Badged Cards..... 10
 - Co-Branded Cards..... 10
 - Credit Cards..... 11
 - Debit Cards..... 11
 - Prepaid Cards..... 11
 - Private Label Cards..... 11
 - Quasi-Cash..... 12
 - Types of Transactions..... 12
 - Card-Present Transactions..... 12
 - Card-Not-Present Transactions..... 12
 - International Transactions..... 12
- Payment Processing..... 14**
 - Authorizations..... 14
 - Online Authorizations..... 14
 - Offline Authorizations..... 14
 - Authorization Workflow..... 15
 - Capture..... 15
 - Capture Workflow..... 16
 - Sales..... 16
 - Sales Workflow..... 16
- Order Tracking..... 18**
 - Request IDs..... 18
 - Reconciliation IDs..... 18

Payment Card Processing.....	19
Processing Basic Authorizations.....	20
Required Fields for Processing Basic Authorizations Using SO APIs.....	20
Authorizing a Payment Using SO APIs.....	20
Example: Processing a Basic Authorization Using the SO API.....	21
Basic Authorization with Line Itemization.....	23
Using Line Items.....	23
Required Fields.....	25
Authorizing a Payment Using SO APIs.....	25
Processing a Basic Authorization with Line Items and DCC.....	26
Captures.....	27
Overview.....	27
Workflow.....	27
Required Fields for a Capture.....	28
Capturing an Authorization.....	29
Example: Capture (NVP).....	30
Example: Capture (XML).....	31
Sales.....	31
Overview.....	32
Dual-Message Processing.....	32
Required Fields for a Sale.....	32
Performing a Sale.....	35
Example: Sale (NVP).....	35
Example: Sale (XML).....	37
Follow-On Credits.....	38
Overview.....	38
Workflow.....	39
Required Fields for a Follow-On Credit.....	40
Crediting a Payment with a Follow-On Credit.....	40
Example: Follow-On Credit (NVP).....	41
Example: Follow-On Credit (XML).....	42
Voids.....	42
Overview.....	43
Required Fields for a Void.....	43
Optional Fields for a Void.....	44

Voiding a Capture or Credit.....	44
Example: Void (NVP).....	45
Example: Void (XML).....	45
Testing the Credit Card Services.....	47
Requirements for Testing Credit Card Services.....	47
Testing the Credit Card Services.....	47
Using Amounts to Simulate Errors.....	49
Required Features for Credit Card Processing.....	50
Debit and Prepaid Card Features.....	51
Real-Time Reversals.....	52
Optional Features.....	53

Recent Revisions to This Document

22.04

Update Guide with Updated Content

Added new Introduction section and the Creating a Basic Authentication Request section to the guide.

Risk Mitigation for Testing

Added an important note about mitigating risk during transaction testing. See [Testing Requirements \(on page 47\)](#).

22.03

22.02

22.01

This revision contains only editorial changes and no technical updates.

21.12

21.11

21.10

21.09

About This Guide

This section describes the audience and purpose of this guide as well as conventions and related documentation. See below for information about how to use this guide and where to find further information.


Audience and Purpose


This guide is written for application developers who want to use the Simple Order API to integrate payment card processing into an order management system.

Implementing the credit card services requires software development skills. You must write code that uses the API request and response fields to integrate the credit card services into your existing order management system.

Conventions

These 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 the Technical Documentation Portal for complete technical documentation:

<https://docs.cybersource.com/en/index.html>

Customer Support

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

<http://www.cybersource.com/support>

Introduction to Payments

With Cybersource payment services, you can process payment cards (tokenized or nontokenized), digital payments such as Apple Pay and Google Pay, and customer ID transactions. You can process payments across the globe and across multiple channels with scalability and security. Cybersource supports an extensive list of payment cards and offers a wide choice of gateways and financial institutions, all through one connection.

Financial Institutions and Payment Networks

A network of Financial Institutions and Payment Networks enables payment services. These entities work together and complete the full payment cycle, from the cardholder purchasing a good or service to the money being transferred from the cardholder's account to the merchant's account.

Acquiring (Merchant) Financial Institutions

An acquiring (merchant) financial institution, also known as an acquirer, offers accounts to businesses that accept payment cards. Before you can accept payments, you must have a merchant account from an acquirer. Your merchant account must be configured to process card-not-present, card-present, or mail-order/telephone-order (MOTO) transactions.

Each acquirer has connections to a limited number of payment processors. You must choose a payment processor that your acquirer supports.

Expect to be charged these fees:

- **Discount rates**—Your acquirer charges a fee and collects a percentage of every transaction. The combination of the fee and the percentage is called the *discount rate*. These charges can be *bundled* (combined into a single charge) or *unbundled* (charged separately) depending on your acquirer and other factors.
- **Interchange fees**—Payment networks such as Visa or Mastercard each have a base fee, called the *interchange fee*, for each type of transaction. Your acquirer and processor can show you ways to reduce this fee.
- **Chargebacks**—When members dispute charges, you can incur *chargebacks*. A chargeback occurs when a charge on a customer's account is reversed. Your acquirer removes the money from your account and could charge you a fee for the chargeback.

To help prevent chargebacks, you should:

- Use accurate merchant descriptors, so customers will recognize the transactions on their statements
- Provide good customer support
- Ensure rapid problem resolution
- Maintain a high level of customer satisfaction
- Minimize fraudulent transactions

When chargebacks become excessive, the following could happen:

- You may be required to undertake business process changes to reduce the number chargebacks, and if the chargebacks are not reduced to a satisfactory level, your account may be terminated.
- If you receive a large number of chargebacks or if a large number of your transactions involve fraud, your acquiring institution may increase your discount rate or revoke your merchant account. Contact your sales representative for information about products that can help prevent fraud.

Issuing (Customer) Financial Institutions

An issuing (customer) financial institution, also known as an issuer, provides payment cards to and underwrites lines of credit for their customers. The issuer provides monthly statements and collects payments. Issuer must follow the rules of the payment card companies to which they belong.

Payment Networks

Payment networks manage communications between acquiring financial institutions and issuing financial institutions. They also develop industry standards, support their brands, and establish fees for acquiring institutions.

Some payment networks, such as Visa and Mastercard, are trade associations that do not issue cards. Instead, issuers are members of these associations, and they issue cards under license from the association.

Other networks, such as Discover and American Express, issue their own cards. Before you process cards from these companies, you must sign agreements with them.

Payment Processors

Payment processors connect with acquirers. Before you can accept payments, you must register with a payment processor. Your acquirer may require you to use a payment processor that has a business relationship with that financial institution.

Your payment processor will assign one or more merchant IDs (MID) to your business. These unique codes identify your business during payment transactions.

Payment Types

Payments can be made using these payment types:

- Co-badged cards
- Co-branded cards
- Credit cards
- Debit cards
- Prepaid cards
- Private label cards
- Quasi-cash

Co-Badged Cards

Cobadged cards are credit cards that integrate two or more payment networks.

Co-Branded Cards

Co-branded cards are credit cards that are branded with a merchant's logo, brand, or other identifier as well as the payment network logo. These cards are not limited for use at the branded merchant and can be used at any merchant that accepts credit cards.

Credit Cards

Credit cards allow cardholders to borrow money to pay for goods and services from merchants that accept credit cards.

Debit Cards

Debit cards are linked to a cardholder's checking account. These cards allow merchants who accept the debit card to deduct funds directly from the linked institution's account.

You can process debit cards using these services:

- Credit card services
- PIN debit services

When you use the Simple Order API in XML format, you must use version 1.52 or later of the XML schema to implement partial authorizations or balance responses.

Prepaid Cards

Prepaid cards allow cardholders to pay for goods and services using money stored directly on the card.

You can process prepaid cards using these services:

- Credit card services
- PIN debit services

When you use the Simple Order API in XML format, you must use version 1.52 or later of the XML schema to implement partial authorizations or balance responses.

Private Label Cards

Private label cards are issued by private companies. They allow cardholders to borrow money to pay for goods exclusively at the issuing company's stores.

Quasi-Cash

Quasi-cash transactions involve instruments that are directly convertible to cash such as web wallets, travelers checks, and lottery tickets.

Types of Transactions

This topic provides information about card-present, card-not-present, and international transactions.

Card-Present Transactions

When a customer uses a card that is physically present in a retail environment to make a purchase, the purchase is known as a card-present transaction.

Card-Not-Present Transactions

When a customer provides a card number, but you do not have access to the physical card, the purchase is known as a card-not-present transaction. This type of transaction typically occurs over the Internet or through a call center. To process card-not-present transactions, use the credit card services described in this guide.

Card-not-present transactions pose an additional level of risk to your business because you cannot directly verify the customer's identification.

International Transactions

When processing international transactions, there are issues that must be considered.

Merchant Remittance Funding

In conjunction with processing international transactions, you can request that transaction proceeds be converted to a currency other than the currency in which the transaction took place for funding into an operating account. Currency conversion uses a foreign exchange rate to calculate how much the transaction currency is worth in terms of the funding currency. The foreign exchange rate might

be explicitly stated as a rate or implicitly stated as a transaction amount. The funded amount and can vary from day to day. The foreign exchange rate might also include a mark-up for the foreign exchange risk, sales commissions, and handling costs.

Payment Processing

At the heart of payment processing is the various services used to process payments. These services allow customers to purchase goods and services, merchants receive payments from the customer's accounts, merchants to provide refunds, and merchants to void transactions.

Authorizations

An authorization confirms that a payment card account includes enough funds to pay for a purchase. There are two basic types of authorizations:

- Online authorization
- Offline authorization

Online Authorizations

Online authorizations provide immediate fund availability confirmations. Further, the customer's financial institution also reduces the amount of credit available in the customer's account, setting aside the authorized funds for the merchant to capture a later time. Authorizations for most payment cards are processed online. Typically, it is safe to start fulfilling the order once you receive an authorization confirmation.

An online authorization confirmation and the subsequent hold on funds will expire after a specific length of time. Thus it is important to capture funds in a timely manner. Most authorizations expire within five to seven days, and the issuing bank sets expiration time interval.

The issuing bank does not inform Cybersource when an authorization confirmation expires. By default, the authorization information for each transaction remains in the Cybersource database for 60 days after the authorization date. To capture an authorization that has expired with the issuing bank, you can resubmit the authorization request.

Offline Authorizations

Online transactions require an internet connection. In situations where the internet is not available, due to an outage for example, merchants can continue to take credit card payments using offline transactions. An offline authorization is an authorization request for which you do not receive an immediate confirmation about the availability of funds.

Offline authorizations have a higher level of risk than do online transactions because they do not provide fund availability confirmations or set aside the funds for later capture. Further, it can take up to five days to receive payment confirmations for offline transactions. To mitigate this risk, you can choose to only fulfill orders after receiving payment confirmations.

Authorization Workflow



1. The customer purchases goods or service from the merchant using a payment card.
2. The merchant sends an authorization request to the acquiring (merchant) bank.
3. The acquiring (merchant) bank forwards the request to the payment network.
4. The payment network forwards the authentication request to the issuer (consumer) bank.
5. If funds are available, the issuer (consumer) bank reserves the amount of the authorization request and returns an authorization approval to the payment network. If the issuer (consumer) bank denies the request, it returns an authorization denial.
6. The payment network forwards the message to the acquiring (merchant) bank.
7. The acquiring (merchant) bank forwards the message to the merchant.

Capture

A capture is a follow-on transaction to an authorization. It is used to transfer the authorized funds from the customer's account to the merchant account. To link the authentication transaction to the capture transaction, you include a request ID in your capture request. This request ID is returned to you during the authentication response.

Captures are typically not performed in real time. They are placed in a batch file and sent to the processor and the processor settles all of the captures at one time. In most cases these batch files are sent and processed outside of the merchant's business hours. It usually takes two to four days for the acquiring financial institution to deposit the funds into the merchant account.

When fulfilling only part of a customer's order, do not capture the full amount of the authorization. Capture only the cost of the delivered items. When you deliver the remaining items, request a new authorization, and then capture the new authorization.

Capture Workflow



1. Merchant sends one or more transaction capture requests to the Acquiring (Merchant) bank.
2. Merchant bank sends the capture package to the payment network.
3. The payment network forwards the capture package to correct the Issuer (Customer) banks.
4. Customer banks settle the transactions and transfers the money to the Acquirer (Customer) bank.

! Important: The payment processor does not notify Cybersource that the money has been transferred. To ensure all captures are processed correctly, you should reconcile your capture requests with the capture reports from your processor.

Sales

A sale is a bundled authorization and capture. Some processors and acquirers require a sale transaction instead of using separate authorization and capture requests. For other processors and acquirers, you can request a sale instead of a separate authorization and capture when you provide the goods or services immediately after taking an order.

There are two types of sale processing:

- Dual Message Processing
- Single Message Processing

Sales Workflow



1. The customer purchases goods or service from the merchant using a payment card.
2. The merchant sends an authorization request to the acquiring (merchant) bank.

3. The acquiring (merchant) bank forwards the request to the payment network.
4. The payment network forwards the authentication request to the issuer (consumer) bank.
5. If funds are available, the issuer sends amount of the authorization request to the acquirer and returns an authorization approval to the payment network. If the issuer (customer) bank denies the request, it returns an authorization denial.
6. The payment network forwards the message to the acquiring (merchant) bank.
7. The acquiring (merchant) bank forwards the message to the merchant.

Order Tracking

Use request IDs and reconciliation IDs to track your orders.

Request IDs

For all services, the request ID is returned in response messages in the **requestID** field.

For each of the following services, use the specified field for the request ID in a request message:

- Capture—**ccCaptureService_authRequestID**
- Credit—**ccCreditService_captureRequestID**
- Void—**voidService_voidRequestID**

Reconciliation IDs

For each of the following services, the reconciliation ID is returned in the specified field in a response message:

- Capture—**ccCaptureReply_reconciliationID**

The reconciliation ID is returned for all processors except JCN Gateway.

- Credit—**ccCreditReply_reconciliationID**

The reconciliation ID is returned for all processors except JCN Gateway.

Related information

[Getting Started with Cybersource Advanced for the Simple Order API](#)

Payment Card Processing

This section describes the services for processing payment card transactions.

Processing Basic Authorizations

This topic shows the minimal set of information required to perform a successful authorization.

Required Fields for Processing Basic Authorizations Using SO APIs

billTo_city

billTo_country

billTo_email

billTo_firstName

billTo_lastName

billTo_postalCode

billTo_state

billTo_street1

card_accountNumber

card_expirationMonth

card_expirationYear

ccAuthService_run

Set this field to [true](#).

merchantID

merchantReferenceCode

purchaseTotals_currency

purchaseTotals_grandTotalAmount

Either **purchaseTotals_grandTotalAmount** or **item_#_unitPrice** must be included in the request.

Authorizing a Payment Using SO APIs

1. Send the service request to one of these endpoints:

- Internet endpoint:

<https://ics2ws.ic3.com/commerce/1.x/transactionProcessor>

- Akamai endpoint:

<https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor>

2. Check the response message to make sure that the request was successful. A value of **ACCEPT** for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Example: Processing a Basic Authorization Using the SO API

Request

```
billTo_city=Sao Paulo
billTo_country=BR
billTo_email=null@cybersource.com
billTo_firstname=Julia
billTo_lastname=Fernandez
billTo_postalCode=01310-000
billTo_state=SP
billTo_street1=R. Augusta
card_accountNumber=41111111XXXXXXXXX
card_expirationMonth=12
card_expirationYear=2023
ccAuthService_run=true
merchant_id=MID23
merchant_referenceCode=Merchant_REF
purchaseTotals_currency=mxn
purchaseTotals_grandTotalAmount=100
```

Response

```
additional_processor_response=e1cdcafc-cdbb-4ef7-8788-a1234e844805
request_id=6461515866500167772420
decision=ACCEPT
reasonCode=100
merchantReferenceCode=Merchant_REF
purchaseTotals_currency=mxn
cardCategory=FccAuthService_reconciliationID=ZUDCXJO8KZRFXQJJ
ccAuthReply_amount=100.00
ccAuthReply_avsCode=5
ccAuthReply_authorizationCode=570110
```

ccAuthReply_processorResponse=1
ccAuthReply_authorizedDateTime=2022-03-01T161947Z
ccAuthReply_paymentNetworkTransactionID=111222

Basic Authorization with Line Itemization

This section will show you how to send a basic authorization request and retrieve a successful response while utilizing line itemization. Using line items, merchants can provide details for each item purchased to the purchaser.

To include line items in an authorization, you remove the **purchaseTotals_grandTotalAmount** field used in a basic authorization and include one or more **item_#_** fields, starting with the **item_0_** fields.

Using Line Items

When using line items the following fields are required for each line item:

- **item_#_unitPrice**
- **item_#_quantity**
- **item_#_productCode**
- **item_#_productSKU** (optional when **item_#_productCode** is set to **default**, **shipping_only**, **handling_only**, or **shipping_and_handling**)
- **item_#_productName** (optional when **item_#_productCode** is set to **default**, **shipping_only**, **handling_only**, or **shipping_and_handling**)

At a minimum, you must include the **item_#_unitPrice** field to include a line item in an authorization. When this is the only field added, the system sets the following fields as shown:

- **item_#_productCode**: default
- **item_#_quantity**: 1

For example, you could include the following three line items:

```
item_0_unitPrice=10.00
item_1_unitPrice=5.99
item_1_quantity=3
item_1_productCode=shipping_only
item_2_unitPrice=29.99
item_2_quantity=3
item_2_productCode=electronic_good
item_2_productSKU=12384569
item_2_productName=receiver
```

Optional Line Item Fields

The following fields can be used to provide more line item information. For more information on each field, see the field reference guide:

- **item_#_buyerRegistration**
- **item_#_commodityCode**
- **item_#_nationalTax**
- **item_#_OrderAcceptanceCity**
- **item_#_OrderAcceptanceCountry**
- **item_#_OrderAcceptancePostalCode**
- **item_#_OrderAcceptanceState**
- **item_#_OrderOriginCity**
- **item_#_OrderOriginCountry**
- **item_#_OrderOriginPostalCode**
- **item_#_OrderOriginState**
- **item_#_otherTax_#_passengerFirstName**
- **item_#_otherTax_#_passengerLastName**
- **item_#_productCode**
- **item_#_productDescription**
- **item_#_productName**
- **item_#_productSKU**
- **item_#_quantity**
- **item_#_shippingDestinationType**
- **item_#_taxTypeApplied**
- **item_#_taxTypeCode**
- **item_#_unitPrice**

Required Fields

billTo_city

billTo_country

billTo_email

billTo_firstName

billTo_lastName

billTo_postalCode

billTo_state

billTo_street1

card_accountNumber

card_expirationMonth

card_expirationYear

ccAuthService_run

Set this field to `true`.

merchantID

merchantReferenceCode

Required when **billTo_personalID** is included in the request.

purchaseTotals_currency

purchaseTotals_grandTotalAmount

Either **purchaseTotals_grandTotalAmount** or **item_#_unitPrice** must be included in the request.

Authorizing a Payment Using SO APIs

1. Send the service request to one of these endpoints:

- Internet endpoint:

<https://ics2ws.ic3.com/commerce/1.x/transactionProcessor>

- Akamai endpoint:

<https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor>

2. Check the response message to make sure that the request was successful. A value of **ACCEPT** for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Processing a Basic Authorization with Line Items and DCC

Request

```
billTo_city=Palo Alto
billTo_country=US
billTo_email=null@cybersource.com
billTo_firstname=Julia
billTo_lastname=Fernandez
billTo_postalCode=94053
billTo_state=CA
billTo_street1=123 Main St.
card_accountNumber=41111111XXXXXXX
card_expirationMonth=12
card_expirationYear=2023
ccAuthService_run=true
dcc_dccIndicator=1
merchant_id=MID23
merchant_referenceCode=Merchant_REF
purchaseTotals_currency=usd
item_0_unitPrice=10.00
item_1_unitPrice=5.99
item_1_quantity=3
item_1_productCode=shipping_only
item_2_unitPrice=29.99
item_2_quantity=3
item_2_productCode=electronic_good
item_2_productSKU=12384569
item_2_productName=receiver
purchaseTotals_exchangeRate=0.91
purchaseTotals_originalAmount=107.33
purchaseTotals_originalCurrency=eur
```

Response

```
additional_processor_response=e1cdcafc-cdbb-4ef7-8788-a1234e844805
request_id=6461515866500167772420
```

```
decision=ACCEPT
reasonCode=100
merchantReferenceCode=Merchant_REF
purchaseTotals_currency=usd
cardCategory=FccAuthService_reconciliationID=ZUDCXJO8KZRFXQJJ
ccAuthReply_amount=117.94
ccAuthReply_avsCode=5
ccAuthReply_authorizationCode=570110
ccAuthReply_processorResponse=1
ccAuthReply_authorizedDateTime=2022-03-01T161947Z
ccAuthReply_paymentNetworkTransactionID=111222
```

Captures

Captures are supported for all processors and acquirers.

Overview

When you are ready to fulfill a customer's order and transfer funds from the customer's bank to your bank, capture the authorization for that order.

When fulfilling only part of a customer's order, do not capture the full amount of the authorization. Capture only the cost of the items that you ship. When you ship the remaining items, request a new authorization, and then capture the new authorization.

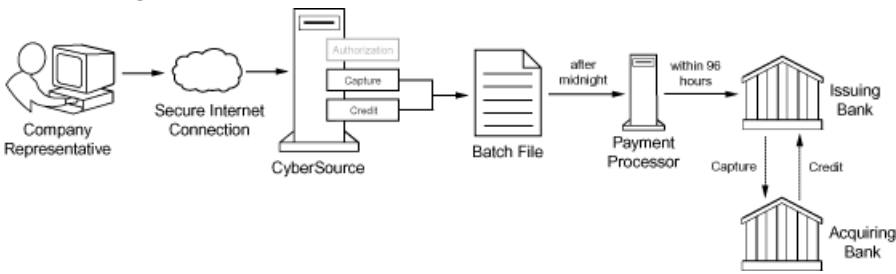
A capture is a follow-on transaction that uses the request ID returned from an authorization. The main purpose of a follow-on transaction is to link two transactions. The request ID links the follow-on transaction to the original transaction. The authorization request ID is used to look up the customer's billing and account information in the Cybersource database, so you are not required to include those fields in the capture request. The original transaction and follow-on transaction are linked in the database and in the Business Center.

Workflow

Unlike authorizations, a capture does not happen in real time. All of the capture requests for a day are placed in a batch file and sent to the processor. In most cases, the batch is settled at night. It usually takes two to four days for your acquiring bank to deposit funds into your merchant bank account.

The following figure shows the steps that occur when you request a capture.

Processing a Capture



1. You send a request for capture over a secure Internet connection.
2. Cybersource validates the order information then stores the capture request in a batch file.
3. After midnight, Cybersource sends the batch file to your payment processor.
4. The processor settles the capture request by transferring funds from the customer's bank account to your bank account.

The processor does not notify Cybersource when a transaction is declined. To ensure that all captures are processed, reconcile your system's reports with the reports from your processor.

Due to the potential delay between authorization and capture, the authorization might expire with the issuing bank before you request capture. Most authorizations expire within five to seven days. If an authorization expires with the issuing bank before you request the capture, your bank or processor might require you to resubmit an authorization request and include a request for capture in the same message.

Cybersource is not informed by the issuing bank when an authorization expires. By default, the authorization remains in the Cybersource database for 60 days after the authorization date, even after it expires with the issuing bank.

Required Fields for a Capture

The following fields are required in a request for a capture:

ccCaptureService_authRequestID

Set this field to the request ID that was included in the authorization response message. Optional when the authorization and capture requests are bundled.

ccCaptureService_run

Set this field to `true`.

item_#_productName

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

item_#_productSKU

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

item_#_quantity

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

merchantID

merchantReferenceCode

purchaseTotals_currency

purchaseTotals_grandTotalAmount

Either **purchaseTotals_grandTotalAmount** or **item_#_unitPrice** must be included in the request.

shipTo_country

Required when any shipping address information is included in the request. Otherwise, this field is optional.

shipTo_postalCode

Required when any shipping address information is included in the request and shipping to the U.S. or Canada. Otherwise, this field is optional.

Related information

[API Field Reference for the Simple Order API](#)

Capturing an Authorization

Do not include any of these services in the request:

- Credit
- Risk update
- Services for other payment methods
- Advanced Fraud Screen

To capture an authorization:

1. Request the service. Set the **ccCaptureService_run** field to `true`, and send the request to one of these endpoints:
 - Internet endpoint:

<https://ics2ws.ic3.com/commerce/1.x/transactionProcessor>

- Akamai endpoint:

<https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor>

2. Include the required fields in the request.
3. Include optional fields in the request as needed.
4. Check the response message to make sure that the request was successful. A value of **ACCEPT** for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Example: Capture (NVP)

Request

```
ccCaptureService_authRequestID=0305782650000167905080
merchantID=Napa Valley Vacations
merchantReferenceCode=482046C3A7E94F5BD1FE3C66C
ccCaptureService_run=true
item_0_unitPrice=49.95
purchaseTotals_currency=USD
```

Response

```
requestID=1019827520348290570293
merchantReferenceCode=482046C3A7E94F5BD1FE3C66C
decision=ACCEPT
reasonCode=100
ccCaptureReply_amount=49.95
purchaseTotals_currency=USD
ccCaptureReply_reasonCode=100
ccCaptureReply_reconciliationID=1094820975023470
```

Example: Capture (XML)

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.37">
  <merchantID>Napa Valley Vacations</merchantID>
  <merchantReferenceCode>482046C3A7E94F5BD1FE3C66C</merchantReferenceCode>
  <item id="0">
    <unitPrice>49.95</unitPrice>
    <quantity>1</quantity>
  </item>
  <purchaseTotals>
    <currency>USD</currency>
  </purchaseTotals>
  <ccCaptureService run="true">
    <authRequestID>0305782650000167905080</authRequestID>
  </ccCaptureService>
</requestMessage>
```

Response

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.37">
  <c:merchantReferenceCode>482046C3A7E94F5BD1FE3C66C</c:merchantReferenceCode>
  <c:requestID>1019827520348290570293</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:ccCaptureReply>
    <c:reasonCode>100</c:reasonCode>
    <c:amount>49.95</c:amount>
    <c:reconciliationID>1094820975023470</c:reconciliationID>
  </c:ccCaptureReply>
</c:replyMessage>
```

Sales

Sales are supported for all processors and acquirers.

Overview

A *sale* is a bundled authorization and capture. Some processors and acquirers require a sale transaction instead of separate authorization and capture requests. For other processors and acquirers, you can request a sale instead of a separate authorization and capture when you ship a customer's goods immediately after taking the order. A sale is typically used for electronic goods and for services that you can turn on immediately.

Related information

[Relaxed Requirements for Address Data and Expiration Date](#)

Dual-Message Processing

Dual-message processing is supported for all processors and acquirers.

To perform a sale, request the authorization and capture services at the same time. When the authorization is successful, the capture is processed immediately and the response message includes results for the authorization and for the capture. When the authorization is declined, the capture is not processed and the response message includes results only for the authorization.

For debit cards and prepaid cards, the issuing bank can approve a partial amount when the balance on the card is less than the requested authorization amount and when the transaction is enabled for partial authorization. When this happens, the capture is not processed. However, you can submit a capture request for the approved amount. For a limited number of processors and card types, partial authorizations are supported for credit cards in addition to debit cards and prepaid cards.

Required Fields for a Sale

The following fields are required in a request for a sale:

billTo_city

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_country

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_email

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_firstName

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_lastName

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_postalCode

Required when the billing country is the U.S. or Canada except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_state

Required when the billing country is the U.S. or Canada except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

billTo_street1

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

card_accountNumber

card_cardType

Required on Visa Platform Connect for all card types. On other processors, required for certain card types. It is strongly recommended that you send the card type even if it is optional for your processor. Omitting the card type can cause the transaction to be processed with the wrong card type.

card_expirationMonth

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

card_expirationYear

Required except when your account is configured for relaxed requirements for address data and expiration date. **Important:** It is your responsibility to determine whether a field is required for the transaction that you are requesting.

ccAuthService_commerceIndicator

Required for requests that include payer authentication data. Otherwise, it is optional.

ccAuthService_run

Set this field to `true`.

ccCaptureService_run

Set this field to `true`.

item_#_productName

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

item_#_productSKU

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

item_#_quantity

Required when the product code is not `default` or one of the values for shipping or handling. Otherwise, this field is optional.

merchantID**merchantReferenceCode****purchaseTotals_currency****purchaseTotals_grandTotalAmount**

Either **purchaseTotals_grandTotalAmount** or **item_#_unitPrice** must be included in the request.

shipTo_city

Required when any shipping address information is included in the request and shipping to the U.S. or Canada. Otherwise, this field is optional.

shipTo_country

Required when any shipping address information is included in the request. Otherwise, this field is optional.

shipTo_postalCode

Required when any shipping address information is included in the request and shipping to the U.S. or Canada. Otherwise, this field is optional.

shipTo_state

Required when any shipping address information is included in the request and shipping to the U.S. or Canada. Otherwise, this field is optional.

shipTo_street1

Required when any shipping address information is included in the request. Otherwise, this field is optional.

Related information

[API Field Reference for the Simple Order API](#)

Performing a Sale

Do not include any of these services in the request:

- Credit
- Risk update
- Services for other payment methods
- Advanced Fraud Screen

To perform a sale:

1. Request the service. Set the **ccAuthService_run** field to `true` and the **ccCaptureService_run** field to `true`, and send the request to one of these endpoints:

- Internet endpoint:

`https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`

- Akamai endpoint:

`https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor`

2. Include the required fields in the request.
3. Include optional fields in the request as needed.
4. Check the response message to make sure that the request was successful. A value of `ACCEPT` for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Example: Sale (NVP)

Request

```
ccAuthService_run=true
ccCaptureService_run=true
merchantID=Napa Valley Vacations
merchantReferenceCode=482046C3A7E94F5
```

```
billTo_firstName=John
billTo_lastName=Doe
billTo_street1=1295 Charleston Rd.
billTo_city=Mountain View
billTo_state=CA
billTo_postalCode=94043
billTo_country=US
billTo_phoneNumber=650-965-6000
billTo_email=jdoe@example.com
item_0_unitPrice=49.95
item_0_quantity=1
purchaseTotals_currency=USD
card_expirationMonth=12
card_expirationYear=2015
card_accountNumber=4111111111111111
card_cvNumber=999 <!-- Required only for Visa Platform Connect. -->
card_cardType=001
card_usage=C <!-- Required only for China UnionPay. -->
cardTypeSelectionIndicator=1 <!-- Required only for China UnionPay. -->
```

Response

Most processors do not return all of the fields in this example.

```
requestID=0305782650000167905080
decision=ACCEPT
reasonCode=100
merchantReferenceCode=482046C3A7E94F5
purchaseTotals_currency=USD
ccAuthReply_reconciliationID=ABCDE12345FGHIJ67890
ccAuthReply_cardCategory=F^
ccAuthReply_cardGroup=0
ccAuthReply_reasonCode=100
ccAuthReply_amount=49.95
ccAuthReply_accountBalance=50.05
ccAuthReply_authorizationCode=123456
ccAuthReply_avsCode=Y
ccAuthReply_avsCodeRaw=YYY
ccAuthReply_processorResponse=A
ccAuthReply_paymentNetworkTransactionID=3312345
ccCaptureReply_amount=49.95
ccCaptureReply_reasonCode=100
ccCaptureReply_reconciliationID=1094820975023470
```

Example: Sale (XML)

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.23">
  <merchantID>Napa Valley Vacations</merchantID>
  <merchantReferenceCode>482046C3A7E94F5</merchantReferenceCode>
  <billTo>
    <firstName>John</firstName>
    <lastName>Doe</lastName>
    <street1>1295 Charleston Rd.</street1>
    <city>Mountain View</city>
    <state>CA</state>
    <postalCode>94043</postalCode>
    <country>US</country>
    <phoneNumber>650-965-6000</phoneNumber>
    <email>jdoe@example.com</email>
  </billTo>
  <item id="0">
    <unitPrice>49.95</unitPrice>
    <quantity>1</quantity>
  </item>
  <purchaseTotals>
    <currency>USD</currency>
  </purchaseTotals>
  <card>
    <accountNumber>4111111111111111</accountNumber>
    <expirationMonth>12</expirationMonth>
    <expirationYear>2015</expirationYear>
    <card_cvNumber>999</card_cvNumber> <!-- Required only for Visa Platform Connect. -->
    <cardType>001</cardType>
    <usage>C</usage> <!-- Required only for China UnionPay. -->
  </card>
  <ccAuthService run="true"/>
  <ccCaptureService run="true"/>
  <cardTypeSelectionIndicator>1</cardTypeSelectionIndicator> <!-- Required only for China
  UnionPay. -->
</requestMessage>
```

Response

Most processors do not return all of the fields in this example.

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.23">
  <c:merchantReferenceCode>482046C3A7E94F5</c:merchantReferenceCode>
```

```
<c:requestID>0305782650000167905080</c:requestID>
<c:decision>ACCEPT</c:decision>
<c:reasonCode>100</c:reasonCode>
<c:purchaseTotals>
  <c:currency>USD</c:currency>
</c:purchaseTotals>
<c:ccAuthReply>
  <c:reasonCode>100</c:reasonCode>
  <c:amount>49.95</c:amount>
  <c:authorizationCode>123456</c:authorizationCode>
  <c:avsCode>Y</c:avsCode>
  <c:avsCodeRaw>YYY</c:avsCodeRaw>
  <c:processorResponse>A</c:processorResponse>
  <c:reconciliationID>ABCDE12345FGHIJ67890</c:reconciliationID>
  <c:accountBalance>50.05</c:accountBalance>
  <c:paymentNetworkTransactionID>3312345</c:paymentNetworkTransactionID>
  <c:cardCategory>F^</c:cardCategory>
  <c:cardGroup>0</c:cardGroup>
</c:ccAuthReply>
<c:ccCaptureReply>
  <c:reasonCode>100</c:reasonCode>
  <c:amount>49.95</c:amount>
  <c:reconciliationID>1094820975023470</c:reconciliationID>
</c:ccCaptureReply>
</c:replyMessage>
```

Follow-On Credits

Follow-on credits are supported for all processors.

Overview

When a request for a credit is successful, the issuing bank for the payment card takes money out of your merchant bank account and returns it to the customer. It usually takes two to four days for your acquiring bank to transfer funds from your merchant bank account.

Carefully control access to the credit service. Do not request this service directly from your customer interface. Instead, incorporate this service as part of your customer service process.

 **Warning:** Follow the preceding guidelines to prevent unauthorized credits.

A *follow-on credit* uses the request ID returned from a capture to link the credit to the capture in the Cybersource database. The capture request ID is used to look up the customer's billing and account information in the database, so you are not required to include those fields in the credit request. However, when you combine a request for a follow-on credit with a request for another service, such as the tax calculation service, you must provide the customer's billing and account information. The capture and follow-on credit are linked in the database and in the Business Center.

Unless otherwise specified, you must request a follow-on credit within 60 days of the authorization.

You can request multiple follow-on credits against a single capture. To perform multiple partial follow-on credits, use the same capture request ID in each follow-on credit request.

Related information

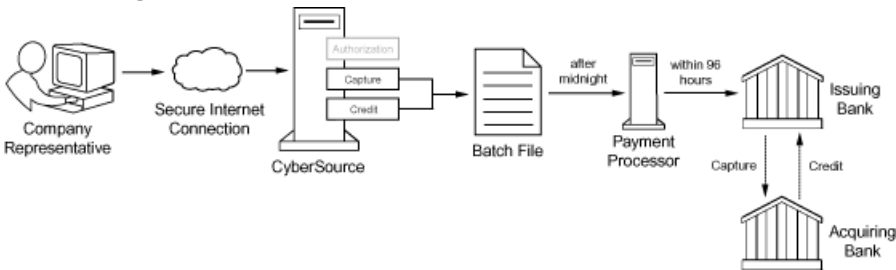
[Required Features for Credit Card Processing \(on page 50\)](#)

Workflow

A credit does not happen in real time. All of the credit requests for a day are placed in a batch file and sent to the processor. In most cases, the batch is settled at night.

The following figure shows the steps that occur when you request a credit.

Processing a Credit



1. You send a request for credit over a secure Internet connection.
2. Cybersource validates the order information then stores the credit request in a batch file.
3. After midnight, Cybersource sends the batch file to your payment processor.
4. The processor settles the credit request by transferring funds from your bank account to the customer's bank account.

The processor does not notify Cybersource when a transaction is declined. To ensure that all credits are processed, reconcile your system's reports with the reports from your processor.

Required Fields for a Follow-On Credit

These fields are required in a request for a follow-on credit:

card_cardType

Required on Visa Platform Connect for all card types. On other processors, required for certain card types. It is strongly recommended that you send the card type even if it is optional for your processor. Omitting the card type can cause the transaction to be processed with the wrong card type.

ccCreditService_captureRequestID

Set to the request ID that was included in the capture response message.

ccCreditService_run

Set this field to [true](#).

merchantID

merchantReferenceCode

purchaseTotals_currency

purchaseTotals_grandTotalAmount

Either **purchaseTotals_grandTotalAmount** or **item_#_unitPrice** must be included in the request.

shipTo_country

Required when any shipping address information is included in the request. Otherwise, this field is optional.

shipTo_postalCode

Required when any shipping address information is included in the request and shipping to the U.S. or Canada. Otherwise, this field is optional.

Related information

[API Field Reference for the Simple Order API](#)

Crediting a Payment with a Follow-On Credit

Do not include any of these services in the request:

- Any other credit card services
- Risk update
- Services for other payment methods

To credit a payment:

1. Request the service. Set the **ccCreditService_run** field to `true`, and send the request to one of these endpoints:

- Internet endpoint:

`https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`

- Akamai endpoint:

`https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor`

2. Include the required fields in the request.

3. Include optional fields in the request as needed.

4. Check the response message to make sure that the request was successful. A value of `ACCEPT` for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Example: Follow-On Credit (NVP)

Request

```
merchantID=Napa Valley Vacations
merchantReferenceCode=482046C3A7E94F5BD1FE3C66C
purchaseTotals_grandTotalAmount=1694.00
purchaseTotals_currency=EUR
ccCreditService_run=true
ccCreditService_captureRequestID=1019827520348290570293
```

Response

```
merchantReferenceCode=482046C3A7E94F5BD1FE3C66C
requestID=1019827520348290570293
decision=ACCEPT
reasonCode=100
purchaseTotals_currency=EUR
ccCreditReply_reasonCode=100
ccCreditReply_amount=1694.00
ccCreditReply_reconciliationID=C3A7E94F5BD1FE3C64820466C
```

Example: Follow-On Credit (XML)

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.86">
  <merchantID>Napa Valley Vacations</merchantID>
  <merchantReferenceCode>482046C3A7E94F5BDC66C</merchantReferenceCode>
  <purchaseTotals>
    <currency>EUR</currency>
    <grandTotalAmount>1694.00</grandTotalAmount>
  </purchaseTotals>
  <ccCreditService run="true">
    <captureRequestID>1019827520348290570293</captureRequestID>
  </ccCreditService>
</requestMessage>
```

Response

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.86">
  <c:merchantReferenceCode>482046C3A75BD1FE3C66C</c:merchantReferenceCode>
  <c:requestID>9057101982752034820293</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>EUR</c:currency>
  </c:purchaseTotals>
  <c:ccCreditReply>
    <c:reasonCode>100</c:reasonCode>
    <c:amount>1694.00</c:amount>
    <c:reconciliationID>C3A7E94F5BD1FE3C64820466C</c:reconciliationID>
  </c:ccCreditReply>
</c:replyMessage>
```

Voids

This section describes voids and the required fields and provides an example.

Overview

A void cancels a capture or credit request that you submitted. A transaction can be voided only when the capture or credit request has not already been submitted to your processor. Capture and credit requests are usually submitted to your processor once a day, so your window for successfully voiding a capture or credit request is small. A void request is declined when the capture or credit request has already been sent to the processor.

A void is a follow-on transaction that uses the request ID returned from a capture or credit. The main purpose of a follow-on transaction is to link two transactions. The request ID links the follow-on transaction to the original transaction. The capture or credit request ID is used to look up the customer's billing and account information in the Cybersource database, so you are not required to include those fields in the void request. The original transaction and follow-on transaction are linked in the database and in the Business Center.

You cannot perform a follow-on credit for a capture that has been voided.

You cannot undo a void.

When you void a capture, a hold remains on the unused credit card funds. If you are not going to re-capture the authorization and if your processor supports authorization reversal after void, it is recommended that you request an authorization reversal to release the hold on the unused credit card funds.

Related information

[Required Features for Credit Card Processing \(on page 50\)](#)

Required Fields for a Void

The following fields are required in a request for a void:

merchantID

merchantReferenceCode

voidService_voidRequestID

Set this field to the request ID that was included in the capture or credit response message.

voidService_run

Set this field to [true](#).

Related information

[API Field Reference for the Simple Order API](#)

Optional Fields for a Void

Additional optional fields are available for specific features.

You can include the following optional fields in a request for a void:

orderRequestToken

voidService_voidRequestToken

Related information

[API Field Reference for the Simple Order API](#)

Voiding a Capture or Credit

Do not include any other services in the request.

To void a capture or credit:

1. Request the service. Set the **voidService_run** field to `true`, and send the request to one of these endpoints:
 - Internet endpoint:
`https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`
 - Akamai endpoint:
`https://ics2wsa.ic3.com/commerce/1.x/transactionProcessor`
2. Include the required fields in the request.
3. Include optional fields in the request as needed.
4. Check the response message to make sure that the request was successful. A value of `ACCEPT` for the **decision** field indicates success. For information about reason codes, see [Reason Codes for the Simple Order API](#).

Example: Void (NVP)

Request

```
merchantID=Napa Valley Vacations  
merchantReferenceCode=482046C3A7E94F5  
voidService_run  
voidService_voidRequestID
```

Response

Most processors do not return all of the fields in this example.

```
requestID=0305782650000167905080  
decision=ACCEPT  
reasonCode=100  
merchantReferenceCode=482046C3A7E94F5  
purchaseTotals_currency=USD  
ccAuthReply_reconciliationID=ABCDE12345FGHIJ67890  
ccAuthReply_cardCategory=F^  
ccAuthReply_cardGroup=0  
ccAuthReply_reasonCode=100  
ccAuthReply_amount=49.95  
ccAuthReply_accountBalance=50.05  
ccAuthReply_authorizationCode=123456  
ccAuthReply_avsCode=Y  
ccAuthReply_avsCodeRaw=YYY  
ccAuthReply_processorResponse=A  
ccAuthReply_paymentNetworkTransactionID=3312345
```

Example: Void (XML)

Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.23">  
  <merchantID>Napa Valley Vacations</merchantID>  
  <merchantReferenceCode>482046C3A7E94F5</merchantReferenceCode>  
  <voidService_run="true"/>  
</voidService>  
</requestMessage>
```

Response

Most processors do not return all of the fields in this example.

```
<c:replyMessage xmlns:c="urn:schemas-cybersource-com:transaction-data-1.23">
  <c:merchantReferenceCode>482046C3A7E94F5</c:merchantReferenceCode>
  <c:requestID>0305782650000167905080</c:requestID>
  <c:decision>ACCEPT</c:decision>
  <c:reasonCode>100</c:reasonCode>
  <c:purchaseTotals>
    <c:currency>USD</c:currency>
  </c:purchaseTotals>
  <c:ccAuthReply>
    <c:reasonCode>100</c:reasonCode>
    <c:amount>49.95</c:amount>
    <c:authorizationCode>123456</c:authorizationCode>
    <c:avsCode>Y</c:avsCode>
    <c:avsCodeRaw>YYY</c:avsCodeRaw>
    <c:processorResponse>A</c:processorResponse>
    <c:reconciliationID>ABCDE12345FGHIJ67890</c:reconciliationID>
    <c:accountBalance>50.05</c:accountBalance>
    <c:paymentNetworkTransactionID>3312345</c:paymentNetworkTransactionID>
    <c:cardCategory>F^</c:cardCategory>
    <c:cardGroup>0</c:cardGroup>
  </c:ccAuthReply>
</c:replyMessage>
```

Testing the Credit Card Services

To ensure that requests are processed correctly, you must test the basic success and error conditions for each service you plan to use.

Requirements for Testing Credit Card Services

! Important: Before you can test, you must contact customer support to activate the credit card services and configure your account for testing. You must also contact your processor to set up your processor account.

! Important: When building your connection to the Cybersource payment gateway, ensure that you implement controls to prevent card testing or card enumeration attacks on your platform. See our [Best Practices Guide](#). If we detect card testing, a card enumeration attack, or other suspicious or fraudulent activity associated with your merchant ID, we might enable a fraud management tool on your behalf in order to mitigate the attack. For more information, see the fraud management documentation in the Business Center.

- Use your regular merchant ID when you test your system.
- Use a real combination for the city, state, and postal code.
- Use a real combination for the area code and telephone number.
- Use a nonexistent account and domain name for the customer's email address.
- URL for the Simple Order API test server:

<https://ics2wstesta.ic3.com/commerce/1.x/transactionProcessor>

Testing the Credit Card Services

Use the payment card numbers in the following list to test the authorization, capture, and credit services. Remove the spaces from the test card numbers when sending them to the test system. Do not use real payment card numbers. To test card types that are not included in the list, use an account number that is in the card's BIN range. For best results, try each test with a different service request and with different test payment card numbers.

Payment Card Numbers for Testing

Payment Card Type	Test Card Number
American Express	3782 8224 6310 005
JCB	3566 1111 1111 1113
Maestro (International)	5033 9619 8909 17 5868 2416 0825 5333 38
Maestro (UK Domestic)	6759 4111 0000 0008
The issue number is not required for Maestro (UK Domestic) transactions	6759 5600 4500 5727 054 5641 8211 1116 6669
Mastercard	2222 4200 0000 1113 2222 6300 0000 1125 5555 5555 5555 4444
UATP	1354 1234 5678 911
Visa	4111 1111 1111 1111

Using Amounts to Simulate Errors

You can simulate error messages by requesting authorization, capture, or credit services with specific amounts that trigger the error messages. These triggers work only on the test server, not on the production server.

Each payment processor uses its own error messages. For more information, see:

https://developer.cybersource.com/api/developer-guides/dita-cybs-so-test-data/so_overview.html

Required Features for Credit Card Processing

This section describes the following kinds of features:

- Features that are required by your processor, an issuer, or a payment card company.
- Features that are required by a country or government.

Debit and Prepaid Card Features

Debit cards and prepaid cards can be processed using the credit card services. You can use PIN debit services to process debit cards if your processor supports PIN debit transactions.

When you use the Simple Order API in XML format, you must use version 1.52 or later of the XML schema to implement partial authorizations or balance responses.

Real-Time Reversals

There are two kinds of real-time reversals:

- A *full authorization reversal* is a service that you can request.

If you do not capture a partial authorization and if full authorization reversals are supported for your processor and card type, you must request a full authorization reversal to release the hold that the authorization placed on the customer's funds. The amount of the reversal must be the amount that was authorized, not the amount that was requested.

- An *automatic partial authorization reversal* is performed automatically under certain conditions.

When you capture a partial authorization for an amount that is less than the approved amount, Cybersource automatically performs a partial authorization reversal if it is supported for your processor and card type. Cybersource performs the automatic partial authorization reversal before sending the capture request to the processor.

Some processors perform an automatic partial authorization reversal when an interchange benefit is available. These processors do not allow Cybersource to perform this functionality.

Related information

[Authorization Reversals](#)

Optional Features

For information about the optional features that are available for credit card processing, see the *Credit Card Services Optional Features Supplement for the Simple Order API*.