Chase Pay

Using the SCMP API

May 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

Restricted Rights Legends

Trademarks

Contents

Recent Revisions to This Document

Release	Changes
May 2019	This revision contains only editorial changes and no technical updates.
July 2018	Updated information about recurring payments. See "Recurring Payments," page 11.
December 2017	This revision contains only editorial changes and no technical updates.
March 2016	Initial release.

About This Guide

Audience and Purpose

This document is written for merchants who want to enable customers to use Chase Pay to pay for in-app transactions. This document provides an overview of integrating Chase Pay and CyberSource services into an order management system and describes how to request the CyberSource API to process authorizations and recurring payments.

Conventions

Notes and Important Statements

Note

A *Note* contains helpful suggestions or references to material not contained in the document.

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 **ics_applications** field. ■ Items that you are instructed to act upon; for example: Click **Save**.
`screen text`	■ XML elements. ■ Code examples and samples. ■ Text that you enter in an API environment; for example: Set the **ics_applications** field to `ics_auth`.

Related Documents

CyberSource Documents:

-

- *Getting Started with CyberSource Advanced for the SCMP API* (PDF | HTML)
- SCMP API Documentation and Downloads page
- *Credit Card Services Using the SCMP API* (PDF | HTML)
- *Payment Network Tokenization Using the SCMP API* (PDF | HTML)

Chase Pay documents on the Chase Paymentech developer center:

- Chase Pay Service Specification
- Chase Pay Companion Guide

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

Introduction

1

Chase Pay

For the Customer:

- Chase Pay enables registered Chase customers to use credit cards that are stored in their digital wallet as payment methods when making purchases using their mobile device.
- Chase Pay requires the customer to enter only a user name and password to pay for goods. It eliminates the need to enter account, shipping, and billing information. The customer logs in to your Chase Pay account and chooses the card with which to pay.

For the Merchant:

- You integrate the Chase Pay lightbox (iFrame).
- Request the Chase Pay API to retrieve the payment network token (see "Payment Network Tokenization," page 8), the expiry date, the cryptogram, and other payment data associated with the transaction.
- Request the CyberSource authorization service and includes the Payment Network Token, the expiry date, the cryptogram, and other payment data associated with the transaction.
- This method is best if your business has a fraud management solution or a records management system that requires payment data relating to transactions.

Payment Network Tokenization

Payment network tokenization enables you to request a payment transaction with a payment network token instead of a primary account number (PAN).

For in-app transactions, payment network tokenization uses some of the CyberSource payer authentication request fields. This approach simplifies your implementation if your order management system already uses payer authentication. See *Payment Tokenization Using the SCMP API* (PDF | HTML).

Requirements

You must:

- Create a CyberSource merchant evaluation account, if you do not have one already:

  https://www.cybersource.com/register/

- Have a merchant account with Chase Paymentech Solutions.

- Install the CyberSource SCMP API Client.

Supported Card Type

CyberSource supports the Visa card type for Chase Pay transactions.

How Chase Pay Works

1. Choose the Chase Pay button. A JavaScript tag embedded within the checkout page renders the Chase Pay button.

2. The browser sends a POST request for the purchase selection to the merchant web server.

3. Your (merchant) web server initiates a MerchantSession request to the Chase Pay Services.

4. Chase Pay Services returns a Digital Session ID to the merchant web server, and it is used in all subsequent request messages.

5. Your web server returns the Digital Session ID to the browser, which sends a POST to the Chase Pay site including the Digital Session ID.

6 The Digital Session ID is validated against the open session.

7 A lightbox is returned and displayed to the customer. The customer authenticates using their Chase Pay credentials, which initiates a session with their Chase Pay wallet.

8 The customer confirms their payment options and shipping preferences, and the lightbox closes. The browser sends a POST to your web server, which includes the resulting Digital Session ID.

9 Your web server initiates a GetCheckoutData request to the Chase Pay Services and includes the Digital Session ID. The GetCheckoutData request retrieves the customer's payment and address information.

10 Chase Pay Services returns the requested data to your web server, including:

- Account number (DPAN).
- Cryptogram (authentication verification value).
- ECI (transaction type).
- Address information (optional).

11 Your web server formats and displays the payment confirmation page to the customer. At this point, the customer reviews and confirms their final payment details using the credentials and the address information that is displayed.

12 The browser sends a POST request to your web server.

13 Instead of including all the normally required fields for an authorization request, you should include the following fields:

- Account number (DPAN)—include in the **customer_cc_number** field.
- Cryptogram (authentication verification value)—include in the **cavv** and **xid** fields.
- ECI (transaction type)—include in the **payment_network_token_transaction_type** field.

14 CyberSource processes the authorization, and the response is sent back to you. It is displayed to the customer confirming the purchase.

CHAPTER 2

Requesting the CyberSource Service

Authorization Service

Note

Your payment processor can include additional API reply fields that are not documented in this guide. See *Credit Card Services Using the SCMP API* for detailed descriptions of any additional API reply fields.

To request an authorization using a Visa card:

Note

See "API Request Fields," page 13, and "API Reply Fields," page 15, for detailed descriptions of each API.

Step 1 Set the **customer_cc_number** field to the payment network token value.

Step 2 Set the **customer_cc_expmo** and **customer_cc_expyr** fields to the payment network token expiration date values.

Step 3 Set the **cavv** field to the 3D Secure cryptogram of the payment network token.

Important

For a 40-byte cryptogram, split the cryptogram into two 20-byte binary values (block A and block B). Send the first 20-byte value (block A) in the cardholder authentication verification value (CAVV) field. Send the second 20-byte value (block B) in the transaction ID (XID) field.

Step 4 Set the **xid** field to the 3D Secure cryptogram of the payment network token.

Step 5 Set the **payment_network_token_transaction_type** field to `1`.

Step 6 Set the **e_commerce_indicator** field to `internet`.

Step 7 Set the **payment_solution** field to `007`.

Example 1 Authorization Request

```
card_type=001
currency=usd
customer_cc_expmo=12
customer_cc_expyr=2021
customer_cc_number=4650100000000839
e_commerce_indicator=internet
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=demomerchant
merchant_ref_number=demorefnum
cavv=ABCDEFabcdefABCDEFabcdef0987654321234567
xid=ABCDEFabcdefABCDEFabcdef0987654321234567
payment_network_token_transaction_type=1
payment_netowrk_token_requestor_id=1234567890
solution_type=007
```

Example 2 Authorization Reply

```
request_token=Ahj/7wSR5C/kX63O2hAKIkGLNkwcsmrSHHlU5tGHRT/hHgzc8BT/hHgk
currency=usd
request_id=4465837560045000001541
auth_rflag=SOK
ics_rmsg=Request was processed successfully.
auth_auth_amount=100.00
auth_rcode=1
auth_trans_ref_no=13209254CGJSMQCQ
auth_auth_code=888888
auth_rmsg=Request was processed successfully.
ics_rflag=SOK
auth_auth_response=100
auth_avs_raw=I1
auth_auth_time=2015-11-03T204917Z
merchant_ref_number=demorefnum
ics_rcode=1
```

Recurring Payments

The recurring payments feature is described in the Payment Network Tokenization Guide. See *Payment Network Tokenization Using the SCMP API* (PDF | HTML).

APPENDIX

A

SCMP API Fields

Data Type Definitions

Data Type	Description
Date and time	Format is YYYY-MM-DDThhmmssZ, where: ▪ T separates the date and the time. ▪ Z indicates Coordinated Universal Time (UTC), which equals Greenwich Mean Time (GMT). Example: 2016-08-11T22:47:57Z equals August 11, 2016, at 22:47:57 (10:47:57 p.m.)
Decimal	Number that includes a decimal point Examples: 23.45, -0.1, 4.0, 90809.0468
Integer	Whole number {..., -3, -2, -1, 0, 1, 2, 3, ...}
Nonnegative integer	Whole number greater than or equal to zero {0, 1, 2, 3, ...}
Positive integer	Whole number greater than zero {1, 2, 3, ...}
String	Sequence of letters, numbers, spaces, and special characters

API Request Fields

Note

Unless otherwise noted, all fields are order and case insensitive, and the fields accept special characters such as @, #, and %.

Table 1 API Request Fields

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
card_type	Type of card to authorize. Value: `001` for Visa.	ics_auth (O)	String (3)
cavv	Cryptogram for payment network tokenization transactions. The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.	ics_auth (R)	String (40)
currency	Currency used for the order: USD.	ics_auth (R)	String (5)
customer_cc_cv_ number	CVN.	ics_auth (O)	Nonnegative integer (4)
customer_cc_expmo	Two-digit month in which the payment network token expires. Format: MM. Possible values: `01` through `12`.	ics_auth (R)	String (2)
customer_cc_expyr	Four-digit year in which the payment network token expires. Format: YYYY.	ics_auth (R)	Nonnegative integer (4)
customer_cc_number	The payment network token value. Populate this field with the decrypted **DPAN** value.	ics_auth (R)	Nonnegative integer (20)
e_commerce_indicator	For a payment network tokenization transaction. Value: `internet` for the Visa card type.	ics_auth (O)	String (20)
grand_total_amount	Grand total for the order. This value cannot be negative. You can include a decimal point (.), but you cannot include any other special characters. CyberSource truncates the amount to the correct number of decimal places.	ics_auth (R)	Decimal (60)
ics_applications	CyberSource services to process for the request: `ics_auth`	ics_auth (R)	String (255)
merchant_id	Your CyberSource merchant ID. Use the same merchant ID for evaluation, testing, and production.	ics_auth (R)	String (30)

Table 1 API Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
merchant_ref_number	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 information about tracking orders, see *Getting Started with CyberSource Advanced for the SCMP API*.	ics_auth (R)	String (50)
payment_network_ token_transaction_type	Type of transaction that provided the token data. This value does not specify the token service provider; it specifies the entity that provided you with information about the token. Set the value for this field to 1.	ics_auth (R)	String (1)
payment_solution	Identifies Chase Pay as the payment solution that is being used for the transaction: Set the value for this field to `007`. This unique ID differentiates digital solution transactions within the CyberSource platform for reporting purposes.	ics_auth (R)	String (3)
payment_network_ requestor_id	Value that identifies your business and indicates that the cardholder's account number is tokenized. This value is assigned by the token service provider and is unique within the token service provider's database.	ics_auth (R)	Integer (11)
xid	Cryptogram for payment network tokenization transactions. The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.	ics_auth (R)	String (40)

API Reply Fields

Important

Because CyberSource can add reply fields, reply codes, and reply flags at any time:

- You must parse the reply data according to the names of the fields instead of the field order in the reply. For more information about parsing reply fields, see the documentation for your client.
- Your error handler should be able to process new reply codes and reply flags without problems.
- Your error handler should use the **ics_rcode** field to determine the result if it receives a reply flag that it does not recognize.

Note

Your payment processor can include additional API reply fields that are not documented in this guide. See *Credit Card Services Using the SCMP API* for detailed descriptions of additional API reply fields.

Table 2 Reply Fields

Field	Description	Returned By	Data Type & Length
auth_auth_amount	Amount that was authorized.	ics_auth	Decimal (15)
auth_auth_avs	AVS result code. See *Credit Card Services Using the SCMP API* for a detailed list of AVS values.	ics_auth	String (1)
auth_auth_code	Authorization code. Returned only when the processor returns this value.	ics_auth	String (7)
auth_auth_response	For most processors, this is the error message sent directly from the bank. Returned only when the processor returns this value.	ics_auth	String (10)
auth_auth_time	Time of authorization in UTC. See "Data Type Definitions," page 12.	ics_auth	Date and time (20)
auth_avs_raw	AVS result code sent directly from the processor. Returned only when the processor returns this value.	ics_auth	String (10)
auth_rcode	Indicates whether the entire request was successful. Possible values: ■ `-1`: An error occurred. ■ `0`: The request was declined. ■ `1`: The request was successful.	ics_auth	Integer (1)

Table 2 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_rflag	One-word description of the result of the entire request. See *Credit Card Services Using the SCMP API* for a detailed list of rflag values.	ics_auth	String (50)
auth_rmsg	Message that explains the reply flag **auth_rflag**. Do not display this message to the customer, and do not use this field to write an error handler.	ics_auth	String (255)
auth_trans_ref_no	Reference number for the transaction. This value is not returned for all processors.	ics_auth	String (60)
currency	Currency used for the order. For the possible values, see the *ISO Standard Currency Codes*.	ics_auth	String (5)
ics_rcode	Indicates whether the service request was successful. Possible values: ▪ `-1`: An error occurred. ▪ `0`: The request was declined. ▪ `1`: The request was successful.	ics_auth	Integer (1)
ics_rflag	One-word description of the result of the entire request. See *Credit Card Services Using the SCMP API* for a detailed list of rflag values.	ics_auth	String (50)
ics_rmsg	Message that explains the reply flag **ics_rflag**. Do not display this message to the customer, and do not use this field to write an error handler.	ics_auth	String (255)
merchant_ref_number	Order reference or tracking number that you provided in the request. If you included multi-byte characters in this field in the request, the returned value might include corrupted characters.	ics_auth	String (50)
request_id	Identifier for the request generated by the client.	ics_auth	String (26)
request_token	Request token data created by CyberSource for each reply. The field is an encoded string that contains no confidential information such as an account or card verification number. The string can contain a maximum of 256 characters.	ics_auth	String (256)