Apple 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: 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.
April 2019	Added support for tokenized transactions using a network token with 3D Secure or SecureCode. See "Option 1: Merchant Decryption," page 21. Added the following request fields that support tokenized transactions using a network token with 3D Secure or SecureCode (see "API Request Fields," page 40): ■ directory_server_transaction_id ■ network_token_cryptogram ■ pa_specification_version Added the following reply field that supports tokenized transactions using a network token with 3D Secure or SecureCode (see "API Reply Fields," page 47): directory_server_transaction_id Added support for the processor *Elavon Americas*. See "Supported Processors, Card Types, and Optional Features," page 17. Added support for merchant-initiated transactions as an optional feature for the following processors (see "Supported Processors, Card Types, and Optional Features," page 17): ■ Chase Paymentech Solutions ■ CyberSource through VisaNet ■ Elavon Americas Added support for subsequent authorizations as an optional feature for the following processors (see "Supported Processors, Card Types, and Optional Features," page 17): ■ FDC Nashville Global ■ JCN Gateway Added support for the following optional features by Elavon Americas (see "Supported Processors, Card Types, and Optional Features," page 17): ■ Multiple partial captures ■ Recurring payments

Release	Changes
March 2019	Added support for the processor *Credit Mutuel-CIC*. See "Supported Processors, Card Types, and Optional Features," page 17. Added support for recurring payments as an optional feature for the processors *Credit Mutuel-CIC* and *SIX*. See "Supported Processors, Card Types, and Optional Features," page 17.
February 2019	Updated the Apple Pay response payload value for the **e_commerce_indicator** field. See "Option 1: Merchant Decryption," page 21, and "e_commerce_indicator," page 43. Updated the JavaScript for obtaining a Base64-encoded value. See "CyberSource Decryption," page 15.
August 2018	This revision contains only editorial changes and no technical updates.
July 2018	All processors: updated information about optional features. See "Supported Processors, Card Types, and Optional Features," page 17. Added support for the processor *Worldpay VAP*. See "Supported Processors, Card Types, and Optional Features," page 17.

About This Guide

Audience and Purpose

This document is written for merchants who want to use Apple Pay in an iOS application and use information from Apple to process payments through CyberSource. This document provides an overview for integrating Apple and CyberSource services into an order management system.

Conventions

Note 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 **card_accountNumber** 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 **ccAuthService_run** field to `true`.

Related Documents

CyberSource Documents:

- *Business Center Overview* (PDF | HTML)
- *Classic Reporting Developer Guide* (PDF | HTML)
- *Credit Card Services Using the SCMP API* (PDF | HTML)
- *Payment Network Tokenization Using the SCMP API* (PDF | HTML)

Apple Documents:

- *PassKit Framework Reference*

Refer to the Support Center for complete CyberSource technical documentation:

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

Customer Support

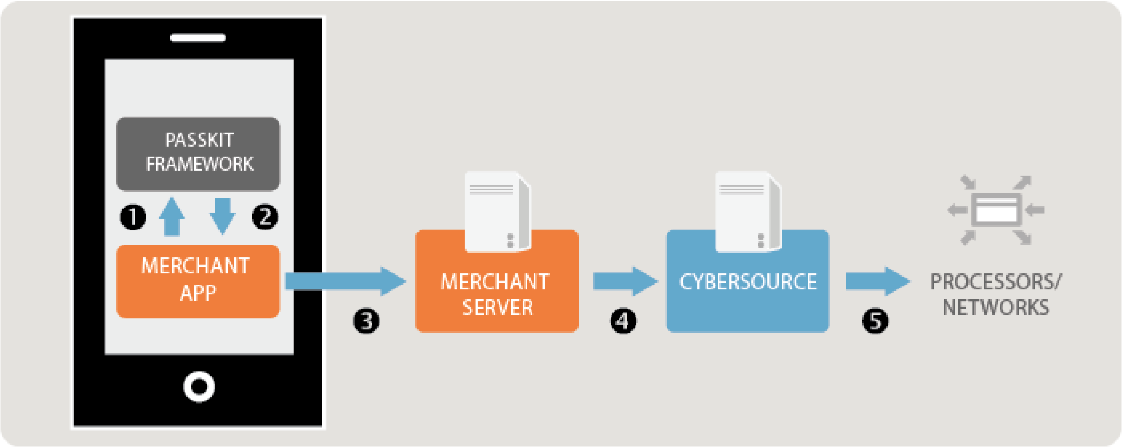
For support information about any CyberSource service, visit the Support Center:

http://www.cybersource.com/support

Apple Pay Integrations

In-App Transactions Using the CyberSource API

Merchant Decryption



1. When the customer chooses to pay with Apple Pay, you use the Apple PassKit Framework to request the encrypted payment data from Apple.

2. Apple uses the Secure Element to create a payment token (the **PKPaymentToken** structure) and encrypt the token's payment data (the **paymentData** field of the **PKPaymentToken** structure) before it sends it your application.

3. You forward the encrypted payment data to your e-commerce back-end system to decrypt. For information on decryption, see:

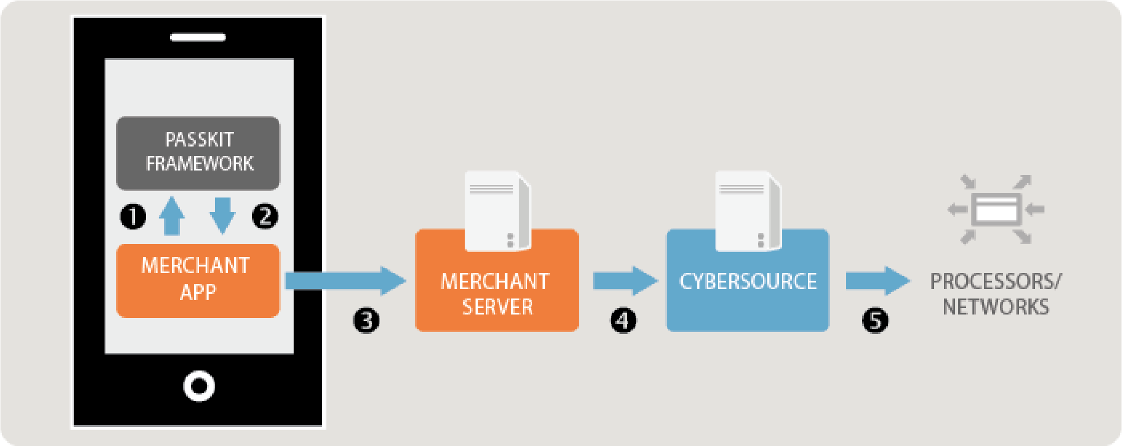
   https://developer.apple.com/library/ios/documentation/PassKit/Reference/PaymentTokenJSON/PaymentTokenJSON.html#//apple_ref/doc/uid/TP40014929-CH8-SW1

4. Using the CyberSource API, you submit the authorization request and include the decrypted payment data. See "Option 1: Merchant Decryption," page 21.

5 CyberSource forwards the information to the payment network, including your processor and the relevant payment card company.

Important You must use the Business Center or one of the CyberSource API services to capture, credit, or void the authorization. See *Credit Card Services Using the SCMP API.*

CyberSource Decryption



1 When the customer chooses to pay with Apple Pay, you use the Apple PassKit Framework to request the encrypted payment data from Apple.

2 Apple uses the Secure Element to create a payment token (the **PKPaymentToken** structure) and encrypt the token’s payment data (the **paymentData** field of the **PKPaymentToken** structure) before it sends it your application.

3 You forward the encrypted payment data to your e-commerce back-end system.

4 Using the CyberSource API, you submit the authorization request. In the **encrypted_payment_data** field include the Base64 encoded value obtained from the **paymentData** field of the **PKPaymentToken** structure. See "Option 2: CyberSource Decryption," page 30.

5 CyberSource decrypts the payment data and forwards the information to the payment network, including your processor and the relevant payment card company.

Important You must use the Business Center or one of the CyberSource API services to capture, credit, or void the authorization. See *Credit Card Services Using the SCMP API.*

Web Transactions

Merchant Decryption

1. When the customer chooses to pay with Apple Pay, you use the Apple Pay JavaScript to request the encrypted payment data from Apple.

2. Apple uses the Secure Element to create a payment token (the **PKPaymentToken** structure) and encrypt the token's payment data (the **paymentData** field of the **PKPaymentToken** structure) before it sends it your application using the **onpaymentauthorized** callback function.

3. You forward the encrypted payment data to your e-commerce back-end system to decrypt. For information on decryption, see:

   https://developer.apple.com/library/ios/documentation/PassKit/Reference/PaymentTokenJSON/PaymentTokenJSON.html#//apple_ref/doc/uid/TP40014929-CH8-SW1

4. Using the CyberSource API, you submit the authorization request and include the decrypted payment data. See "Option 2: CyberSource Decryption," page 30.

5. CyberSource forwards the information to the payment network, including your processor and the relevant payment card company.

Important You must use the Business Center or one of the CyberSource API services to capture, credit, or void the authorization. See *Credit Card Services Using the SCMP API.*

CyberSource Decryption

1. When the customer chooses to pay with Apple Pay, you use the Apple Pay JavaScript to request the encrypted payment data from Apple.

2. Apple uses the Secure Element to create a payment token (the **PKPaymentToken** structure) and encrypt the token's payment data (the **paymentData** field of the **PKPaymentToken** structure) before it sends it your application via the **onpaymentauthorized** callback function.

3. You forward the encrypted payment data to your e-commerce back-end system.

4. Using the CyberSource API, you submit the authorization request. In the **encrypted_payment_data** field include the Base64 encoded value obtained from the **paymentData** field of the **PKPaymentToken** structure. See "Option 2: CyberSource Decryption," page 30.

5 CyberSource decrypts the payment data and forwards the information to the payment network, including your processor and the relevant payment card company.

Important You must use the Business Center or one of the CyberSource API services to capture, credit, or void the authorization. See *Credit Card Services Using the SCMP API.*

Requirements

Important You must be an *Admin* or *Team Agent* user of your Apple Developer Program account.

For details on each requirement below, see:

https://developer.apple.com/support/apple-pay-domain-verification/

To configure your requirements:

Step 1 Register your merchant ID.

Note If you are currently processing In-App transactions, you can use the same merchant ID for processing Web transactions.

Step 2 Create or upload a Certificate Signing Request (CSR), which is used to encrypt the payment information during the payment process.

If you are using the merchant decryption method (see "Option 1: Merchant Decryption," page 21), create a CSR.

If you are using the CyberSource decryption method (see "Option 2: CyberSource Decryption," page 30), upload the CSR that you created in the Business Center (see "Enrolling in Apple Pay," page 18).

Note If you are currently processing In-App transactions, you can use the same CSR for processing Web transactions.

Step 3 Register your domain. Registration is required in order to use Apple Pay on your web site.

Step 4 Create a Merchant Identity Certificate. This certificate is required in order to connect to the Apple servers.

Note All optional features are described in *Payment Network Tokenization Using the SCMP API*.

Apple Pay JavaScript

Use the Apple Pay JavaScript to accept Apple Pay payments on your web site. The Apple Pay JavaScript tests that Apple Pay exists on your web site, displays the Apple Pay sheet, and receives the payment token.

Apple Pay Button

Important When a customer clicks or taps an Apple Pay button, it must invoke the Apple Pay payment sheet.

For information on how to use Apple Pay buttons and the button styles, see:

https://developer.apple.com/apple-pay/Apple-Pay-Identity-Guidelines.pdf

You can use CSS templates provided by Apple to display the Apple Pay button on your web site. There are two templates: *logo only* button and *buy with* button. For more information, see Displaying the Apple Pay Button.

ApplePaySession Class

The **ApplePaySession** class manages the payment process on your web site. The **ApplePaySession** object is the entry point for Apple Pay on your web site.

Before displaying the Apple Pay button (see "Apple Pay Button," page 13) or creating an Apple Pay session (see "Create ApplePaySession Object," page 14), ensure that the Apple Pay JavaScript API is available and enabled on the device.

To enable the Apple Pay JavaScript API:

Step 1 Verify that the **window.ApplePaySession** class exists.

Step 2 Call its **canMakePayments** or **canMakePaymentsWithActiveCard** method:

- canMakePayments—verifies that the device is enabled for Apple Pay.
- canMakePaymentsWithActiveCard—verifies that the device is enabled for Apple Pay and the customer has a card stored on the device. You can call this method only if Apple Pay is the default payment method during your checkout flow, or if you want to add the Apple Pay button to your product detail page.

Create ApplePaySession Object

There are two required arguments when creating an **ApplePaySession** object:

- Version number—the API version is 1.
- Payment request—the **PaymentRequest** dictionary contains the information required in order to display the payment form.

When the session is created, call its **begin** method to display the payment form. This method can be called only when invoked by a user’s request.

Merchant Validation

When the payment form is displayed, the **onvalidatemerchant** callback function is called and provides a URL to pass to your server for validating the merchant session. Refer to the Merchant Validation section.

Payment Confirmation

When the customer confirms the payment by clicking or tapping the Apple Pay button, the **onpaymentauthorized** callback function is called and provides the payment token.

Merchant Decryption

Forward the encrypted payment data to your e-commerce back-end system to decrypt. For information on decryption, see:

> https://developer.apple.com/library/ios/documentation/PassKit/Reference/PaymentTokenJSON/PaymentTokenJSON.html#//apple_ref/doc/uid/TP40014929-CH8-SW1

Using the CyberSource API, submit the authorization request and include the decrypted payment data. See "Option 1: Merchant Decryption," page 21.

CyberSource Decryption

Forward the encrypted payment data to your e-commerce back-end system.

Using the CyberSource API, submit the authorization request. In the **encrypted_payment_data** field include the Base64 encoded value obtained from the **paymentData** object. Example 1 shows the JavaScript for obtaining this value. See "Option 2: CyberSource Decryption," page 30.

Example 1 JavaScript for Obtaining a Base64-Encoded Value

```
session.onpaymentauthorized = function (event) {

var paymentDataString = JSON.stringify(event.payment.token.paymentData);

var paymentDataBase64 = btoa(paymentDataString);

…

}
```

CHAPTER 2

Getting Started

Requirements

- CyberSource account. If you do not already have a CyberSource account, contact your local CyberSource sales representative. You can find your local Sales office here: http://www.cybersource.com/locations/
- Merchant account with a supported processor (see Table 1, "Processors, Card Types, and Optional Features," on page 17).
- You must have an *Admin* or *Team Agent* user of the Apple Pay Developer account.

Important

Apple Pay relies on payment network tokenization. You can sign up for Apple Pay only if both of the following statements are true:

- Your processor supports payment network tokenization.
- CyberSource supports payment network tokenization with your processor.

If one or both of the preceding statements are not true, you must take one of the following actions before you can sign up for Apple Pay:

- Obtain a new merchant account with a processor that supports payment network tokenization.
- Wait until your processor supports payment network tokenization.

Supported Processors, Card Types, and Optional Features

Note All optional features, except split shipments, are described in *Payment Network Tokenization Using the SCMP API* (PDF | HTML). Split shipments are described in *Credit Card Services Using the SCMP API* (PDF | HTML).

Table 1 Processors, Card Types, and Optional Features

Processor	Card Types	Optional Features
American Express Direct	American Express	▪ Multiple partial captures ▪ Recurring payments
Barclays	Visa, Mastercard, Maestro (International), Maestro (UK Domestic)	▪ Multiple partial captures ▪ Recurring payments
Chase Paymentech Solutions	Visa, Mastercard, American Express, Discover, Maestro (International)	▪ Merchant-Initiated transactions ▪ Multiple partial captures ▪ Recurring payments
Credit Mutuel-CIC	Visa, Mastercard, Cartes Bancaires	Recurring Payments
CyberSource through VisaNet. The supported acquirers are: ▪ Australia and New Zealand Banking Group Ltd. (ANZ) ▪ CitiBank Singapore Ltd. ▪ Global Payments Asia Pacific ▪ Vantiv ▪ Westpac	Visa, Mastercard	▪ Merchant-Initiated transactions ▪ Recurring payments ▪ Split shipments
Elavon Americas	Visa, Mastercard, American Express, JCB, Discover	▪ Merchant-Initiated transactions ▪ Multiple partial captures ▪ Recurring payments
FDC Compass	Visa, Mastercard, American Express	▪ Multiple partial captures ▪ Recurring payments
FDC Nashville Global	Visa, Mastercard, American Express, Discover	▪ Multiple partial captures ▪ Recurring payments ▪ Subsequent authorizations
GPN	Visa, Mastercard, American Express	▪ Recurring payments ▪ Split shipments

Table 1 Processors, Card Types, and Optional Features (Continued)

Processor	Card Types	Optional Features
JCN Gateway	JCB	▪ Multiple partial captures ▪ Subsequent authorizations
OmniPay Direct. The supported acquirers are: ▪ Bank of America Merchant Services ▪ First Data Merchant Solutions (Europe) ▪ Global Payments International Acquiring	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	▪ Multiple partial captures ▪ Recurring payments
SIX	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	Recurring Payments
Streamline	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	▪ Multiple partial captures ▪ Recurring payments ▪ Subsequent authorizations
TSYS Acquiring Solutions	Visa, Mastercard, American Express	▪ Multiple partial captures ▪ Recurring payments
Worldpay VAP Worldpay VAP was previously called *Litle*.	Visa, Mastercard	Recurring Payments

Enrolling in Apple Pay

To enroll for Apple Pay:

Step 1 Log in to the Business Center:

- Test transactions: https://ebctest.cybersource.com
- Live transactions: https://ebc.cybersource.com

a Under **Account Management** in the left navigation panel, choose **Digital Payment Solutions**.

b Click **Sign Up**. Follow the steps to verify your account information and accept the agreement on the Apple Pay Developers web site.

Step 2 Generate a Certificate Signing Request (CSR).

a Enter your **Apple Merchant ID** that you registered in the Certificates, Identifiers and Profiles area of the Member Center on the Apple web site.

Important CyberSource decryption method—Step b and Step c are required.

Merchant decryption method—Step b is required only for saving your Apple Pay merchant ID. The CSR must be obtained directly from Apple.

b Click **Generate CSR** to save your Apple Pay merchant ID and to generate a CSR that is associated with your merchant ID.

c Submit the CSR to Apple.

Go to the Apple web site and upload the CSR. Apple provides you with an Apple Pay Certificate for your Apple Merchant ID. For information about adding certificates to your Apple Merchant ID, see the *PassKit Framework Reference*.

Important A CSR submitted to Apple expires after 25 months. CyberSource recommends generating and submitting a new CSR prior to the expiration date. See "Generating a New CSR," page 20.

Step 3 Obtain the Apple Pay Certificate.

If you do not have the Apple Pay Certificate, complete the process that is described in the *PassKit Framework Reference.* The Apple Pay Certificate is required for creating an iOS application. The Apple Pay Certificate is not needed for payment processing with CyberSource.

Step 4 Test your software. See "Requesting the Authorization Service," page 21.

Note If you are using a CyberSource test account, you must connect to the Apple developer system and not to the Apple production system.

Important After you complete your testing, you must create a new CSR for the CyberSource production system, and you must use that CSR for the Apple production system. Until you perform these steps, you cannot enable payments in your iOS application.

Step 5 Repeat Steps 1, 2, 3, and 5 with your CyberSource production account and the Apple production account.

Generating a New CSR

To generate a new CSR:

Step 1 Log in to the Business Center:

- Test transactions: https://ebctest.cybersource.com
- Live transactions: https://ebc.cybersource.com

Step 2 Under **Account Management** in the left navigation panel, choose **Digital Payment Solutions**.

Step 3 Click **Enabled**.

Step 4 Generate a New CSR:

a Enter the Apple Merchant ID that you registered in the Certificates, Identifiers, and Profiles area of the Member Center on the Apple web site.

b Click **Generate New CSR**.

The new CSR replaces the previous CSR in the list. The previous CSR continues to be active until its expiration date (25 months from the date it was generated.)

c Download and submit the new CSR to Apple.

Single Transaction Report

Go to the Business Center and use the Single Transaction Report to obtain information about your transactions:

- In the Business Center, use the Transaction Search page to identify Apple transactions. You can search for transactions by date, application type, customer name, and other transaction identifiers.
- For information about the Single Transaction Report, see the *Classic Reporting Developer Guide* (PDF | HTML).

CHAPTER 3

Requesting the Authorization Service

Option 1: Merchant Decryption

Visa Transaction

To request an authorization for a Visa transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

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 values from the payment network token expiration date.

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

Step 4 Set the **network_token_cryptogram** field to the network token cryptogram.

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

Step 6 Set the **e_commerce_indicator** field to the ECI value contained in the Apple Pay response payload (`5=vbv` and `7=internet`).

Step 7 Set the **payment_solution** field to `001`.

Example 2 Authorization Request (Visa)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=001
cavv=EHuWW9PiBkWvqE5juRwDzAUFBAk=
currency=USD
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=4650100000000839
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
e_commerce_indicator=internet
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_network_token_transaction_type=1
payment_solution=001
```

Example 3 Authorization Reply (Visa)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=15356267CR2XF23W
currency=USD
ics_rcode=1
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697369261766124501541
request_token=Ahj/7wSR/UowD7HRf/RKIsdagry/dhgsdrhshv/4ee3Y6L/6JQAAA9xYR
```

Mastercard Transaction

To request an authorization for a Mastercard transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

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 values from the payment network token expiration date.

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

Step 4 Set the **network_token_cryptogram** field to the network token cryptogram.

Step 5 Set the **ucaf_collection_indicator** field to `2`.

Step 6 Set the **payment_network_token_transaction_type** field to `1`.

Step 7 Set the **e_commerce_indicator** field to `spa`.

Step 8 Set the **payment_solution** field to `001`.

Example 4 Authorization Request (Mastercard)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=002
currency=usd
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=5555555555554444
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
e_commerce_indicator=spa
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=med123
merchant_ref_number=ref123
ucaf_authentication_data=ABCDEFabcdefABCDEFabcdef0987654321234567
ucaf_collection_indicator=2
payment_network_token_transaction_type=1
payment_solution=001
```

Example 5 Authorization Reply (Mastercard)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=15356268CR2XF23X
currency=USD
ics_rcode=1
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697369268106124601541
request_token=Ahj/7wSR/UoVm1bMmziHSZjMECT/h+KjMHSB04gwGA2dDjQoxQAAA6xdr
```

American Express Transaction

To request an authorization for an American Express transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

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 values from the payment network token expiration date.

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

Important Include the whole 20-byte cryptogram in the **cavv** field. For a 40-byte cryptogram, split the cryptogram into two 20-byte binary values (block A and block B). Set the **cavv** field to the block A value and set the **xid** field to the block B value.

Step 4 Set the **network_token_cryptogram** field to the network token cryptogram.

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

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

Step 7 Set the **payment_solution** field to `001`.

Example 6 Authorization Request (American Express)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=003
cavv=EHuWW9PiBkWvqE5juRwDzAUFBAk=
currency=USD
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=4650100000000839
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
e_commerce_indicator=aesk
grand_total_amount=100
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_network_token_transaction_type=1
payment_solution=001
```

Example 7 Authorization Reply (American Express)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=15356269CR2XF23Y
currency=USD
ics_rcode=1
ics_return_code=1000000
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697369273896124701541
request_token=Ahj/7wSR/UowJcJsefb4e64b4e64756hjrd8/P6lGBLhJRpbZQAAAPxNY
```

Discover Transaction

To request an authorization for a Discover transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

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 values from the payment network token expiration date.

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

Step 4 Set the **network_token_cryptogram** field to the network token cryptogram.

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

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

Step 7 Set the **payment_solution** field to `001`.

Example 8 Authorization Request (Discover)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=004
cavv=EHuWW9PiBkWvqE5juRwDzAUFBAk=
currency=USD
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=6011111111111117
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
e_commerce_indicator=dipb
grand_total_amount=100
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_network_token_transaction_type=1
payment_solution=001
```

Example 9 Authorization Reply (Discover)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=15356269CR2XF23Y
currency=USD
ics_rcode=1
ics_return_code=1000000
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697369273896124701541
request_token=Ahj/7wSR/UowJcJsefb4e64b4e64756hjrd8/P6lGBLhJRpbZQAAAPxNY
```

JCB Transaction

To request an authorization for a JCB transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

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 values from the payment network token expiration date field.

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

Step 4 Set the **payment_network_token_transaction_type** field to `1`.

Step 5 Set the **eci_raw** field to the ECI value contained in the Apple Pay response payload.

Step 6 Set the **payment_solution** field to `001`.

Example 10 Authorization Request (JCB)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=007
currency=usd
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=3566111111111113
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
eci_raw=05
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=med123
cavv=EHuWW9PiBkWvqE5juRwDzAUFBAk=
payment_network_token_transaction_type=1
payment_solution=001
```

Example 11 Authorization Reply (JCB)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=15356268CR2XF23X
currency=USD
ics_rcode=1
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697369268106124601541
request_token=Ahj/7wSR/UoVm1bMmziHSZjMECT/h+KjMHSB04gwGA2dDjQoxQAAA6xdr
```

Option 2: CyberSource Decryption

Visa Transaction

To request an authorization for a Visa transaction:

Note

See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

Step 1 Set the **encrypted_payment_data** field to the base64 encoded value obtained from the **paymentData** property of the **PKPaymentToken** object. See "CyberSource Decryption," page 10.

Step 2 Set the **encrypted_payment_descriptor** field to `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`

Step 3 Set the **payment_solution** field to `001`.

Example 12 Authorization Request (Visa)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
currency=USD
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
card_type=001
encrypted_payment_data=eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9
encrypted_payment_descriptor=RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U
encrypted_payment_encoding=Base64
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_solution=001
```

Example 13 Authorization Reply (Visa)

```
auth_auth_amount=100.00
auth_auth_avs=X
auth_auth_code=888888
auth_auth_response=100
auth_avs_raw=I1
auth_rcode=1
auth_rflag=SOK
auth_rmsg=Request was processed successfully.
auth_trans_ref_no=35363393DQMME5FP
currency=USD
ics_rcode=1
ics_rflag=SOK
ics_rmsg=Request was processed successfully.
merchant_ref_number=ref123
request_id=4697330206876530801545
request_token=Ahj/7BT/w1nwRbSB04gwhQybdU2yMTRCrwJyP6kjUeh08Z7iQAAA/wTV
token_expiration_month=07
token_expiration_year=2025
token_prefix=411111
token_suffix=1111
```

Mastercard Transaction

To request an authorization for a Mastercard transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

Step 1 Set the **encrypted_payment_data** field to the base64 encoded value obtained from the **paymentData** property of the **PKPaymentToken** object. See "CyberSource Decryption," page 10.

Step 2 Set the **encrypted_payment_descriptor** field to `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`

Step 3 Set the **payment_solution** field to `001`.

Example 14 Authorization Request (Mastercard)

```
bbill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=002
currency=USD
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
encrypted_payment_data=eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9
encrypted_payment_descriptor=RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U
encrypted_payment_encoding=Base64
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_solution=001
```

Example 15 Authorization Reply (Mastercard)

```
request_token=Ahj/7wSR5C/p6oJEy1gKIkGLNkwcsmrWHHlU5tGHST/hHgzdACT/hVB3c
currency=usd
request_id=4465838340055000001541
auth_rflag=SOK
ics_rmsg=Request was processed successfully.
auth_auth_amount=100.00
auth_rcode=1
auth_trans_ref_no=13209255CGJSMQCR
auth_auth_code=888888
auth_rmsg=Request was processed successfully.
ics_rflag=SOK
auth_auth_response=100
auth_avs_raw=I1
auth_auth_time=2015-11-03T205035Z
merchant_ref_number=ref123
ics_rcode=1
token_prefix=128945
token_suffix=2398
token_expirationMonth=08
token_expirationYear=2021
```

American Express Transaction

To request an authorization for an American Express transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

Step 1 Set the **encrypted_payment_data** field to the base64 encoded value obtained from the **paymentData** property of the **PKPaymentToken** object. See "CyberSource Decryption," page 10.

Step 2 Set the **encrypted_payment_descriptor** field to `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`

Step 3 Set the **payment_solution** field to `001`.

Example 16 Authorization Request (American Express)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=003
currency=USD
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
encrypted_payment_data=eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9
encrypted_payment_descriptor=RRklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U
encrypted_payment_encoding=Base64
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_solution=001
```

Example 17 Authorization Reply (American Express)

```
request_token=Ahj/7wSR5C/wGXKw1xAKIkGLNkwcsmraHHlU5tGHaT/hHgzecDT/h6BBL
currency=usd
request_id=4465839210285000001541
auth_rflag=SOK
ics_rmsg=Request was processed successfully.
auth_auth_amount=100.00
auth_rcode=1
auth_trans_ref_no=13209256CGJSMQCZ
auth_auth_code=888888
auth_rmsg=Request was processed successfully.
ics_rflag=SOK
auth_auth_response=100
auth_avs_raw=I1
auth_auth_time=2015-11-03T205202Z
merchant_ref_number=ref123
ics_rcode=1
token_prefix=593056
token_suffix=0842
token_expirationMonth=08
token_expirationYear=2021
```

Discover Transaction

To request an authorization for a Discover transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

Step 1 Set the **encrypted_payment_data** field to the base64 encoded value obtained from the **paymentData** property of the **PKPaymentToken** object. See "CyberSource Decryption," page 10.

Step 2 Set the **encrypted_payment_descriptor** field to `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`

Step 3 Set the **payment_network_token_transaction_type** field to `1`.

Step 4 Set the **payment_solution** field to `001`.

Example 18 Authorization Request (Discover)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=004
currency=USD
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
encrypted_payment_data=eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9
encrypted_payment_descriptor=RRklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U
encrypted_payment_encoding=Base64
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=mid123
merchant_ref_number=ref123
payment_network_token_transaction_type=1
payment_solution=001
```

Example 19 Authorization Reply (Discover)

```
request_token=Ahj/7wSR5C/wGXKw1xAKIkGLNkwcsmraHHlU5tGHaT/hHgzecDT/h6BBL
currency=usd
request_id=4465839210285000001541
auth_rflag=SOK
ics_rmsg=Request was processed successfully.
auth_auth_amount=100.00
auth_rcode=1
auth_trans_ref_no=13209256CGJSMQCZ
auth_auth_code=888888
auth_rmsg=Request was processed successfully.
ics_rflag=SOK
auth_auth_response=100
auth_avs_raw=I1
auth_auth_time=2015-11-03T205202Z
merchant_ref_number=ref123
ics_rcode=1
token_prefix=601111
token_suffix=1117
token_expirationMonth=08
token_expirationYear=2021
```

JCB Transaction

To request an authorization for a JCB transaction:

Note See the Relaxed Requirements for Address Data and Expiration Date page and "API Request Fields," page 40, for details and field descriptions.

Step 1 Set the **encrypted_payment_data** field to the base64 encoded value obtained from the **paymentData** property of the **PKPaymentToken** object.

Step 2 Set the **encrypted_payment_decryptor** field to `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`.

Step 3 Set the **payment_solution** field to `001`.

Example 20 Authorization Request (JCB)

```
bill_address1=123 Main Street
bill_address2=Suite 12345
bill_city=Small Town
bill_country=US
bill_state=CA
bill_zip=98765
card_type=007
currency=usd
customer_cc_expmo=12
customer_cc_expyr=2031
customer_cc_number=5555555555554444
customer_email=js@example.com
customer_firstname=Jane
customer_lastname=Smith
customer_phone=999-999-9999
eci_raw=05
grand_total_amount=100.00
ics_applications=ics_auth
merchant_id=med123
cavv=EHuWW9PiBkWvqE5juRwDzAUFBAk=
payment_network_token_transaction_type=1
payment_solution=001
```

Example 21 Authorization Reply (JCB)

```
request_token=Ahj/7wSR5C/p6oJEy1gKIkGLNkwcsmrWHHlU5tGHST/hHgzdACT/hVB3c
currency=usd
request_id=4465838340055000001541
auth_rflag=SOK
ics_rmsg=Request was processed successfully.
auth_auth_amount=100.00
auth_rcode=1
auth_trans_ref_no=13209255CGJSMQCR
auth_auth_code=888888
auth_rmsg=Request was processed successfully.
ics_rflag=SOK
auth_auth_response=100
auth_avs_raw=I1
auth_auth_time=2015-11-03T205035Z
merchant_ref_number=ref123
ics_rcode=1
token_prefix=128945
token_suffix=2398
token_expirationMonth=08
token_expirationYear=2021
```

Additional CyberSource Services

Refer to the *Credit Card Services Using the SCMP API* for information on how to request these follow-on services.

Table 2 CyberSource Services

CyberSource Service	Description
Capture	A follow-on service that uses the request ID returned from the previous authorization. The request ID links the capture to the authorization. This service transfers funds from the customer’s account to your bank and usually takes two to four days to complete.
Sale	A sale is a bundled authorization and capture. Request the authorization and capture services at the same time. CyberSource processes the capture immediately.
Auth Reversal	A follow-on service that uses the request ID returned from the previous authorization. An auth reversal releases the hold that the authorization placed on the customer’s credit card funds. Use this service to reverse an unnecessary or undesired authorization.

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

Relaxed Requirements for Address Data and Expiration Date

To enable relaxed requirements for address data and expiration date, contact CyberSource Customer Support to have your account configured for this feature. For details about relaxed requirements, see the Relaxed Requirements for Address Data and Expiration Date page.

API Request Fields

Note

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

Table 3 Request Fields

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
bill_address1	First line of the billing street address.	ics_auth (R)[2]	CyberSource through VisaNet: String (40) All other processors: String (60)
bill_address2	Additional address information. **Example** Attention: Accounts Payable	ics_auth (O)	CyberSource through VisaNet: String (40) All other processors: String (60)
bill_city	City of the billing address.	ics_auth (R)[2]	String (50)
bill_country	Country of the billing address. Use the two-character *ISO Standard Country Codes*.	ics_auth (R)[2]	String (2)
bill_state	State or province of the billing address. For an address in the U.S. or Canada, use the *State, Province, and Territory Codes for the United States and Canada*.	ics_auth (R)[2]	String (2)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
bill_zip	Postal code for the billing address. The postal code must consist of 5 to 9 digits. When the billing country is the U.S., the 9-digit postal code must follow this format: [5 digits][dash][4 digits] **Example** 12345-6789 When the billing country is Canada, the 6-digit postal code must follow this format: [alpha][numeric][alpha][space] [numeric][alpha][numeric] **Example** A1B 2C3	ics_auth (R)[2]	CyberSource through VisaNet: String (9) All other processors: String (10)
card_type	Type of card to authorize. Possible values: - `001`: Visa - `002`: Mastercard - `003`: American Express - `004`: Discover	ics_auth (O)	String (3)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
cavv	***Visa*** 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. ***American Express*** For a 20-byte cryptogram, set this field to the cryptogram for payment network tokenization transactions. For a 40-byte cryptogram, set this field to block A of the cryptogram for payment network tokenization transactions (see "American Express Transaction," page 25). The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats. ***Discover*** Cryptogram for payment network tokenization transactions. The value for this field can be a 20 or 40-character hex binary. All cryptograms use one of these formats. ***CyberSource through VisaNet*** The value for this field corresponds to the following data in the TC 33 capture file[1]: ■ Record: CP01 TCR8 ■ Position: 77-78 ■ Field: CAVV version and authentication action.	ics_auth (R)	String (40)
currency	Currency used for the order: USD	ics_auth (R)	String (5)
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.	ics_auth (R)	Nonnegative integer (20)
customer_email	Customer's email address.	ics_auth (R)[2]	String (255)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
customer_firstname	Customer's first name. For a credit card transaction, this name must match the name on the card.	ics_auth (R)[2]	String (60)
customer_lastname	Customer's last name. For a credit card transaction, this name must match the name on the card.	ics_auth (R)[2]	String (60)
customer_phone	Customer's phone number. CyberSource recommends that you include the country code when the order is from outside the U.S.	ics_auth (O)	String (15)
directory_server_transaction_id	Identifier generated during the authentication transaction by the Mastercard Directory Server and passed back with the authentication results.	ics_auth (O)	String (36)
e_commerce_indicator	For a payment network tokenization transaction. The values are required for the merchant decryption method (see "Option 1: Merchant Decryption," page 21). Possible values: ■ aesk: American Express card type ■ spa: Mastercard card type ■ vbv: Visa card type mapped for Apple Pay transactions with eCommerce commerce indicator of 5 internet: Visa card type mapped for Apple Pay transactions with eCommerce commerce indicator of 7 ■ dipb: Discover card type	ics_auth (See description)	String (20)
eci_raw	Raw electronic commerce indicator (ECI).	ics_auth	String (2)
encrypted_payment_data	The encrypted payment data value. Populate this field with the encrypted payment data obtained from the **paymentData** field of the **PKPaymentToken** structure. See the *PassKit Framework Reference*.	ics_auth (R)	String (3072)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
encrypted_payment_descriptor	Format of the encrypted payment data. The value for Apple Pay is: `RklEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQVlNRU5U`	ics_auth (R)	String (128)
encrypted_payment_encoding	Encoding method used to encrypt the payment data: `Base64`	ics_auth (R)	String (6)
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 service 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)
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)
network_token_cryptogram	Token authentication verification value cryptogram. For token-based transactions with 3D Secure or SecureCode, you must submit both types of cryptograms: network token and 3D Secure/ SecureCode. The value for this field must be 28-character Base64 or 40-character hex binary. All cryptograms use one of these formats.	ics_auth (O)	String (40)
pa_specification_version	The 3D Secure version that you used for Secured Consumer Authentication (SCA); for example, 3D Secure version 1.0.2 or 2.0.0.	ics_auth (O)	String (20)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
payment_network_ token_assurance_level	Confidence level of the transaction. This value is assigned by the token service provider. **Note** This field is supported only for CyberSource through VisaNet and FDC Nashville Global.	ics_auth (O)	String (2)
payment_network_ token_device_tech_ type	Type of technology used in the device to store token data. Possible value: 001: Secure Element (SE) Smart card or memory with restricted access and encryption to prevent data tampering. For storing payment credentials, a SE is tested against a set of requirements defined by the payment networks. **Note** This field is supported only for FDC Compass.	ics_auth (O)	Integer (3)
payment_network_ token_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. **Note** This field is supported only for CyberSource through VisaNet, FDC Nashville Global, and Chase Paymentech Solutions.	ics_auth (O)	Integer (1)
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. An application on the customer's mobile device provided the token data.	ics_auth (R)	String (1)
payment_solution	Identifies Apple Pay as the payment solution that is being used for the transaction: Set the value for this field to 001. **Note** This unique ID differentiates digital solution transactions within the CyberSource platform for reporting purposes.	ics_auth (R)	String (3)
ucaf_authentication_ data	Cryptogram for payment network tokenization transactions with Mastercard.	ics_auth (R)	String (32)
ucaf_collection_ indicator	Required field for payment network tokenization transactions with Mastercard. Set the value for this field to 2.	ics_auth (R)	String with numbers only (1)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

[2] This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Table 3 Request Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
xid	***Visa*** 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. ***American Express*** For a 20-byte cryptogram, set this field to the cryptogram for payment network tokenization transactions. For a 40-byte cryptogram, set this field to block A of the cryptogram for payment network tokenization transactions (see "American Express Transaction," page 25). 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)

1 The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

2 This field is optional if your CyberSource account is configured for relaxed requirements for address data and expiration date. See "Relaxed Requirements for Address Data and Expiration Date," page 39. **Important** It is your responsibility to determine whether a field is required for the transaction you are requesting.

Offer-Level Fields

Table 4 Offer-Level Fields

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
amount	Per-item price of the product. This value cannot be negative.You can include a decimal point (.), but you cannot include any other special characters.	ics_auth (See description)	Decimal (15)
merchant_product_sku	Identification code for the product. This field is required when the **product_code** value is not `default` or one of the values related to shipping and/or handling.	ics_auth (See description)	String (255)
product_code	Type of product. This value is used to determine the product category: electronic, handling, physical, service, or shipping. The default is `default`.	ics_auth (See description)	String (255)
product_name	Name of the product. This field is required when the **product_code** value is not `default` or one of the values related to shipping and/or handling.	ics_auth (See description)	String (255)

Table 4 Offer-Level Fields (Continued)

Field	Description	Used By: Required (R) or Optional (O)	Data Type (Length)
quantity	The default is 1. This field is required when the **product_code** value is not `default` or one of the values related to shipping and/or handling.	ics_auth (See description)	Integer (10)
tax_amount	Total tax to apply to the product. This value cannot be negative.	ics_auth (See description)	String (15)

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 5 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 value is the error message sent directly from the bank. Returned only when the processor returns this value.	ics_auth	String (10)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_avs_raw	AVS result code sent directly from the processor. Returned only when the processor returns this value.	ics_auth	String (10)
auth_payment_card_service	Mastercard service that was used for the transaction. Mastercard provides this value to CyberSource. Possible value: 53: Mastercard card-on-file token service This field is returned only for CyberSource through VisaNet. ***CyberSource through VisaNet*** The value for this field corresponds to the following data in the TC 33 capture file[1]: - Record: CP01 TCR6 - Position: 133-134 Field: Mastercard Merchant on-behalf service. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth	String (2)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant’s acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_payment_card_service_result	Result of the Mastercard card-on-file token service. Mastercard provides this value to CyberSource. Possible values: ▪ C: Service completed successfully. ▪ F: One of the following: • Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 81 for an authorization or authorization reversal. • Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 01 for a tokenized request. • Token requestor ID is missing or formatted incorrectly. ▪ I: One of the following: • Invalid token requestor ID. • Suspended or deactivated token. • Invalid token (not in mapping table). ▪ T: Invalid combination of token requestor ID and token. ▪ U: Expired token. ▪ W: Primary account number (PAN) listed in electronic warning bulletin.This field is returned only for CyberSource through VisaNet. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth	String (1)
auth_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)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_reversal_ payment_card_service	Mastercard service that was used for the transaction. Mastercard provides this value to CyberSource. Possible value: 53: Mastercard card-on-file token service This field is returned only for CyberSource through VisaNet. ***CyberSource through VisaNet*** The value for this field corresponds to the following data in the TC 33 capture file[1]: ■ Record: CP01 TCR6 ■ Position: 133-134 Field: Mastercard Merchant on-behalf service. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth_ reversal	String (2)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_reversal_payment_card_service_result	Result of the Mastercard card-on-file token service. Mastercard provides this value to CyberSource. Possible values: ▪ C: Service completed successfully. ▪ F: One of the following: • Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 81 for an authorization or authorization reversal. • Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 01 for a tokenized request. • Token requestor ID is missing or formatted incorrectly. ▪ I: One of the following: • Invalid token requestor ID. • Suspended or deactivated token. • Invalid token (not in mapping table). ▪ T: Invalid combination of token requestor ID and token. ▪ U: Expired token. ▪ W: Primary account number (PAN) listed in electronic warning bulletin.This field is returned only for CyberSource through VisaNet. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth_reversal	String (1)
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)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_transaction_ qualification	Type of authentication for which the transaction qualifies as determined by the Mastercard authentication service, which confirms the identity of the cardholder. Mastercard provides this value to CyberSource. Possible values: ■ 1: Transaction qualifies for Mastercard authentication type 1. ■ 2: Transaction qualifies for Mastercard authentication type 2. This field is returned only for CyberSource through VisaNet. ***CyberSource through VisaNet*** The value for this field corresponds to the following data in the TC 33 capture file[1]: ■ Record: CP01 TCR6 ■ Position: 132 Field: Mastercard Member Defined service. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth	String (1)
card_suffix	Last four digits of the cardholder's account number. This field is returned only for tokenized transactions. You can use this value on the receipt that you give to the cardholder. **Note** This field is returned only for CyberSource through VisaNet and FDC Nashville Global. ***CyberSource through VisaNet*** The value for this field corresponds to the following data in the TC 33 capture file[1]: ■ Record: CP01 TCRB ■ Position: 85 ■ Field: American Express last 4 PAN return indicator.	ics_auth	String (4)
currency	Currency used for the order. For the possible values, see the *ISO Standard Currency Codes*.	ics_auth	String (5)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
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)
payment_network_ token_account_status	Possible values: ■ N: Nonregulated ■ R: Regulated This field is returned only for CyberSource through VisaNet. **Note** This field is returned only for CyberSource through VisaNet.	ics_auth	String (1)
payment_network_ token_assurance_level	Confidence level of the tokenization. This value is assigned by the token service provider. **Note** This field is returned only for CyberSource through VisaNet. and CyberSource through VisaNet and FDC Nashville Global.	ics_auth	String (2)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
payment_network_ token_original_card_ category	Mastercard product ID associated with the primary account number (PAN). For the possible values, see "MasterCard Product IDs" in *Credit Card Services Using the SCMP API*. ***CyberSource through VisaNet*** For the possible values, see "Mastercard Product IDs" in *Credit Card Services for CyberSource through VisaNet Using the SCMP API*. **Note** This field is returned only for Mastercard transactions on CyberSource through VisaNet.	ics_auth	String (3)
payment_network_ token_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. This value is returned only if the processor provides it. **Note** This field is supported only for CyberSource through VisaNet and FDC Nashville Global. **Note** This field is returned only for CyberSource through VisaNet. and CyberSource through VisaNet and FDC Nashville Global.	ics_auth	Integer (11)
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)
token_expiration_month	Month in which the token expires. CyberSource includes this field in the reply message when it decrypts the payment blob for the tokenized transaction. Format: MM. Possible values: `01` through `12`.	ics_auth	String (2)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
token_expiration_year	Year in which the token expires. CyberSource includes this field in the reply message when it decrypts the payment blob for the tokenized transaction. Format: YYYY.	ics_auth	String (4)
token_prefix	First six digits of token. CyberSource includes this field in the reply message when it decrypts the payment blob for the tokenized transaction.	ics_auth	String (6)
token_suffix	Last four digits of token. CyberSource includes this field in the reply message when it decrypts the payment blob for the tokenized transaction.	ics_auth	String (4)

[1] The TC 33 Capture file contains information about the purchases and refunds that a merchant submits to CyberSource. CyberSource through VisaNet creates the TC 33 Capture file at the end of the day and sends it to the merchant's acquirer, who uses this information to facilitate end-of-day clearing processing with payment card companies.