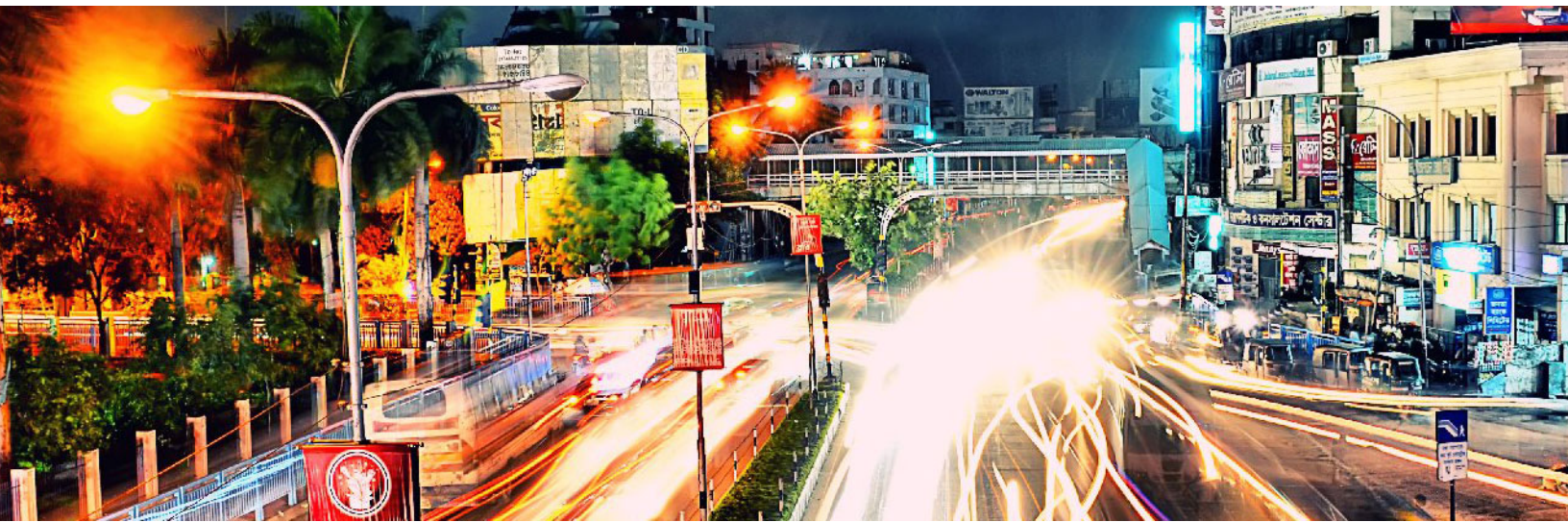


# Apple Pay

## Using the SCMP API



**CyberSource<sup>®</sup>**  
A Visa Solution

## 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](mailto: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

© 2020. 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, and the CyberSource logo 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.

**Revision:** May 2020

# Contents

## **Recent Revisions to This Document** 5

### **About This Guide** 6

Audience and Purpose 6

Conventions 6

    Important Statements 6

    Text and Command Conventions 6

Related Documents 7

Customer Support 7

---

## **Chapter 1** **Getting Started** 8

Requirements 8

Supported Processors, Card Types, and Optional Features 9

Enrolling in Apple Pay 11

    Generating a New CSR 12

Transaction Reports 12

---

## **Chapter 2** **Apple Pay Integrations** 13

In-App Transactions Using the CyberSource API 13

    Merchant Decryption 13

    CyberSource Decryption 14

Web Transactions 15

    Merchant Decryption 15

    CyberSource Decryption 15

    Requirements 16

    Apple Pay JavaScript 17

        Apple Pay Button 17

        ApplePaySession Class 17

        Create ApplePaySession Object 18

        Merchant Validation 18

        Payment Confirmation 18

Merchant Decryption	18
CyberSource Decryption	19

---

<b>Chapter 3</b>	<b>Requesting the Authorization Service</b>	<b>20</b>
	Option 1: Merchant Decryption	20
	Visa Transaction	20
	Mastercard Transaction	22
	American Express Transaction	24
	Discover Transaction	26
	JCB Transaction	27
	Option 2: CyberSource Decryption	29
	Visa Transaction	29
	Mastercard Transaction	30
	American Express Transaction	32
	Discover Transaction	34
	JCB Transaction	35
	Additional CyberSource Services	37

---

<b>Appendix A</b>	<b>API Fields</b>	<b>38</b>
	Data Type Definitions	38
	Relaxed Requirements for Address Data and Expiration Date	38
	Request Fields	39
	Offer-Level Fields	45
	Reply Fields	46

# Recent Revisions to This Document

Release	Changes
May 2020	Updated information about recurring payments. See <a href="#">"Supported Processors, Card Types, and Optional Features," page 9</a> .
February 2020	Added support for the processor <i>Moneris</i> . See <a href="#">"Supported Processors, Card Types, and Optional Features," page 9</a> .
January 2020	OmniPay Direct: added support for acquirer Cardnet International. See <a href="#">"Supported Processors, Card Types, and Optional Features," page 9</a> . Updated the <b>grand_total_amount</b> field length. See <a href="#">grand_total_amount, page 43</a> .
November 2019	Changed <i>payment network tokenization</i> to <i>authorizations with payment network tokens</i> throughout. SIX: added an important note. See <a href="#">SIX, page 10</a> .
October 2019	Updated Related Documents. See <a href="#">"Related Documents," page 7</a> . Updated Business Center procedures. See <a href="#">"Enrolling in Apple Pay," page 11</a> , and <a href="#">"Generating a New CSR," page 12</a> . Updated information about reports. See <a href="#">"Transaction Reports," page 12</a> .
September 2019	This revision contains only editorial changes and no technical updates.

# 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

---

### Important Statements

---



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

---

## Text and Command Conventions

Convention	Usage
<b>Bold</b>	<ul style="list-style-type: none"> <li>Field and service names in text; for example: Include the <b>card_accountNumber</b> field.</li> <li>Items that you are instructed to act upon; for example: Click <b>Save</b>.</li> </ul>
Screen text	<ul style="list-style-type: none"> <li>XML elements.</li> <li>Code examples and samples.</li> <li>Text that you enter in an API environment; for example: Set the <b>ccAuthService_run</b> field to <code>true</code>.</li> </ul>

## Related Documents

---

CyberSource Documents:

- *Business Center User Guide* ([PDF](#) | [HTML](#))
- *Business Center Reporting User Guide* ([PDF](#) | [HTML](#))
- *Credit Card Services Using the SCMP API* ([PDF](#) | [HTML](#))
- *Authorizations with Payment Network Tokens 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](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>

# 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 9).
- You must have an *Admin* or *Team Agent* user of the [Apple Pay Developer account](#).



---

Apple Pay relies on authorizations with payment network tokens. You can sign up for Apple Pay only when both of the following statements are true:

- Your processor supports payment network tokens.
- CyberSource supports payment network tokens 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 tokens.
  - Wait until your processor supports payment network tokens.
-



# Supported Processors, Card Types, and Optional Features

Merchant-initiated transactions, multiple partial captures, and subsequent authorizations are described in *Authorizations with Payment Network Tokens Using the SCMP API* ([PDF](#) | [HTML](#)). Recurring payments and 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	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
Barclays	Visa, Mastercard, Maestro (International), Maestro (UK Domestic)	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
Chase Paymentech Solutions	Visa, Mastercard, American Express, Discover, Maestro (International)	<ul style="list-style-type: none"> <li>■ Merchant-Initiated transactions</li> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
Credit Mutuel-CIC	Visa, Mastercard, Cartes Bancaires	Recurring payments
CyberSource through VisaNet. The supported acquirers are: <ul style="list-style-type: none"> <li>■ Australia and New Zealand Banking Group Ltd. (ANZ)</li> <li>■ CitiBank Singapore Ltd.</li> <li>■ Global Payments Asia Pacific</li> <li>■ Vantiv</li> <li>■ Westpac</li> </ul>	Visa, Mastercard	<ul style="list-style-type: none"> <li>■ Merchant-Initiated transactions</li> <li>■ Recurring payments</li> <li>■ Split shipments</li> </ul>
Elavon Americas	Visa, Mastercard, American Express, JCB, Discover	<ul style="list-style-type: none"> <li>■ Merchant-Initiated transactions</li> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
FDC Compass	Visa, Mastercard, American Express	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
FDC Nashville Global	Visa, Mastercard, American Express, Discover	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> <li>■ Subsequent authorizations</li> </ul>
GPN	Visa, Mastercard, American Express	Split shipments
JCN Gateway	JCB	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Subsequent authorizations</li> </ul>

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

Processor	Card Types	Optional Features
Moneris	Visa, Mastercard, American Express	<ul style="list-style-type: none"> <li>■ Merchant-initiated transactions</li> <li>■ Recurring payments</li> </ul>
OmniPay Direct. The supported acquirers are: <ul style="list-style-type: none"> <li>■ Bank of America Merchant Services</li> <li>■ Cardnet International</li> <li>■ First Data Merchant Solutions (Europe)</li> <li>■ Global Payments International Acquiring</li> </ul>	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> </ul>
SIX <b>Important</b> SIX is supported only for card-present processing.	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	Recurring payments
Streamline	Visa, Mastercard, Maestro (UK Domestic), Maestro (International)	<ul style="list-style-type: none"> <li>■ Multiple partial captures</li> <li>■ Recurring payments</li> <li>■ Subsequent authorizations</li> </ul>
TSYS Acquiring Solutions	Visa, Mastercard, American Express	Multiple partial captures
Worldpay VAP  Worldpay VAP was previously called Litle. Litle was purchased by Vantiv, which was then purchased by Worldpay VAP. If you have any questions about this situation, contact your account manager at Worldpay VAP.	Visa, Mastercard	Recurring payments

# Enrolling in Apple Pay

---

## To enroll in Apple Pay:

---

- Step 1** Log in to the Business Center:
- Test: <https://ebctest.cybersource.com/ebc2/>
  - Live: <https://ebc2.cybersource.com/ebc2/>
- Step 2** On the left navigation pane, click the **Payment Configuration** icon.
- Step 3** Click **Digital Payment Solutions**. The Digital Payments page appears.
- Step 4** Click **Configure**. The Apple Pay Registration panel opens.
- Step 5** Enter your Apple Merchant ID.
- Step 6** Click **Generate New CSR**.
- Step 7** To download your CSR, click the **Download** icon next to the key.
- Step 8** Follow your browser's instructions to save and open the file.
- Step 9** Complete the enrollment process by submitting your CSR to Apple.
- Step 10** For information about adding certificates to your Apple Merchant ID, refer to the Apple Pay PassKit:  
<https://developer.apple.com/documentation/passkit>
- Step 11** Test your software. See "[Requesting the Authorization Service](#)," page 20.



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



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 12** Repeat Steps 1 through 10 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: <https://ebctest.cybersource.com/ebc2/>
  - Live: <https://ebc2.cybersource.com/ebc2/>
- Step 2** On the left navigation pane, click the **Payment Configuration** icon.
- Step 3** Click **Digital Payment Solutions**. The Digital Payments page appears.
- Step 4** Click **Configure**. The Apple Pay Registration panel opens.
- Step 5** To download your CSR, click the **Download** icon next to the key.
- Step 6** Follow your browser's instructions to save and open the file.
- Step 7** To edit your Apple Merchant ID, click the **Edit** icon. The Edit CSR panel opens.
- Step 8** Modify your merchant ID as necessary and click **Update**.
- 

## Transaction Reports

---

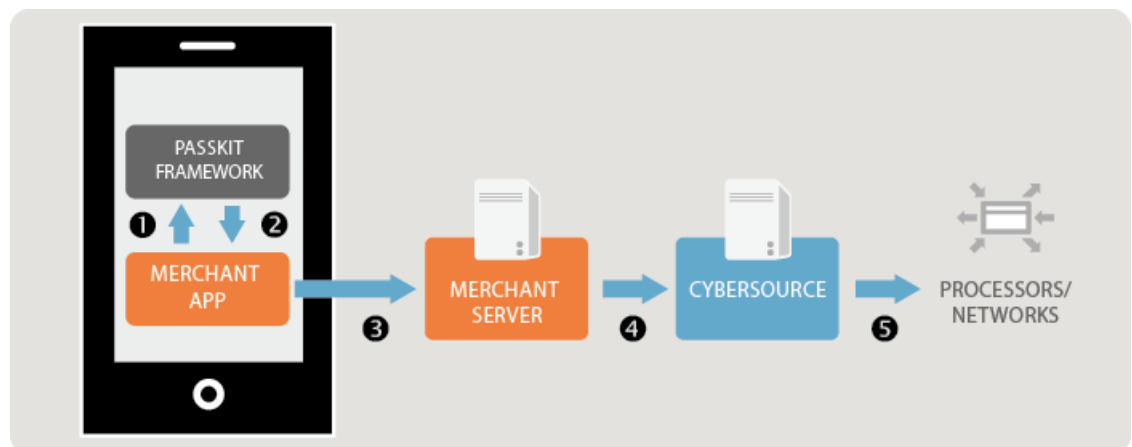
Go to the Business Center and use the Transaction Request Report to obtain information about your transactions:

- 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 Transaction Request Report, see the *Business Center Reporting User Guide* ([PDF](#) | [HTML](#))

# Apple Pay Integrations

## In-App Transactions Using the CyberSource API

### Merchant Decryption



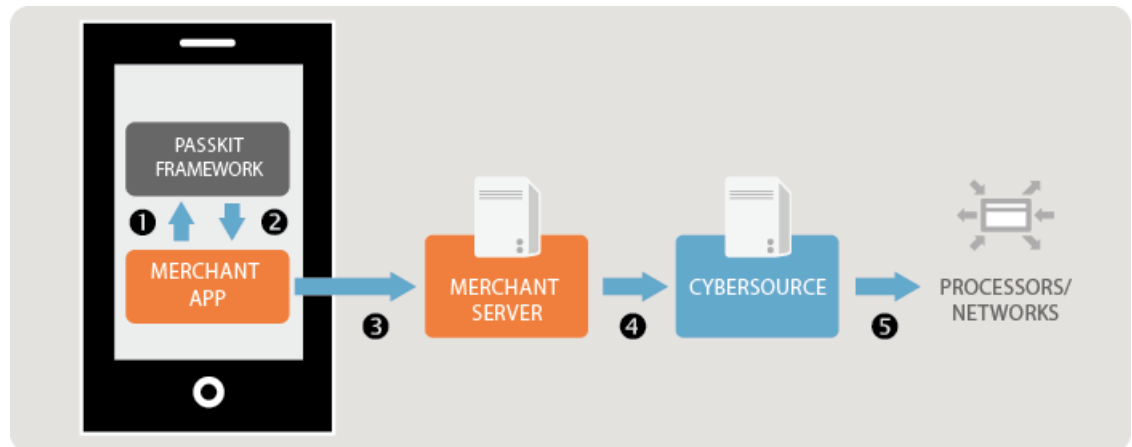
- 1 When the customer chooses to pay with Apple Pay, you use the Apple PassKit Framework to request the encrypted payment data from Apple.
- 2 Apple uses the Secure Element to create a payment token (the **PKPaymentToken** structure) and encrypt the token's payment data (the **paymentData** field of the **PKPaymentToken** structure) before it sends it your application.
- 3 You forward the encrypted payment data to your order management 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](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 20.

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



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 order management 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 29.
- 5 CyberSource decrypts the payment data and forwards the information to the payment network, including your processor and the relevant payment card company.



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 order management 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](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 29.
- 5 CyberSource forwards the information to the payment network, including your processor and the relevant payment card company.



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 order management 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 29.

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



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



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.  
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 20](#)), create a CSR.  
  
If you are using the CyberSource decryption method (see "[Option 2: CyberSource Decryption](#)," [page 29](#)), upload the CSR that you created in the Business Center (see "[Enrolling in Apple Pay](#)," [page 11](#)).  
  
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.
- 

All optional features are described in *Authorizations with Payment Network Tokens Using the SCMP API* ([PDF](#) | [HTML](#)).



## 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



When a customer clicks or taps an Apple Pay button, you must ensure that it invokes 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 17) or creating an Apple Pay session (see "[Create ApplePaySession Object](#)," page 18), 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 order management 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](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 20.

## CyberSource Decryption

Forward the encrypted payment data to your order management 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 29.

### 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);  
  
    ...  
  
}
```

---

# Requesting the Authorization Service

## Option 1: Merchant Decryption

---

### Visa Transaction

See the [Relaxed Requirements for Address Data and Expiration Date page](#) and "Request Fields," [page 39](#), for details and field descriptions.

#### To request an authorization for a Visa transaction:

---

- 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=4111111111111111
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/dhgdrhshv/4ee3Y6L/6JQAAA9xYR

```

---

## Mastercard Transaction

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

### To request an authorization for a Mastercard transaction:

---

- 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=5432543254325432
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

See the [Relaxed Requirements for Address Data and Expiration Date page](#) and "Request Fields," [page 39](#), for details and field descriptions.

### To request an authorization for an American Express transaction:

---

- 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.



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=37828224631xxxx
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/UowJcJsefb4e64b4e64756hjrds/P6lGBLhJRpbZQAAAPxNY
```

---

## Discover Transaction

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

### To request an authorization for a Discover transaction:

---

- 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=601111111111xxxx
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/P61GBLhJRpbZQAAAPxNY

```

---

## JCB Transaction

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

### To request an authorization for a JCB transaction:

- 
- 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=35661111111111xxxx
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

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

#### To request an authorization for a Visa transaction:

---

- 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 14.
- Step 2** Set the `encrypted_payment_descriptor` field to `Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U`
- 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=Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
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/wlnwRbSB04gwhQybdU2yMTRCwJyP6kjUeh08Z7iQAAA/wTV
token_expiration_month=07
token_expiration_year=2025
token_prefix=411111
token_suffix=1111

```

---

## Mastercard Transaction

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

### To request an authorization for a Mastercard transaction:

- 
- 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 14.
  - Step 2** Set the **encrypted\_payment\_descriptor** field to  
Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
  - 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=Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
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/p6oJEy1gKIiKGLNkwcsmrWHH1U5tGHST/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

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," [page 39](#), for details and field descriptions.

### To request an authorization for an American Express transaction:

---

- 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 14](#).
- Step 2** Set the `encrypted_payment_descriptor` field to `Rk1lEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U`
- 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=eyJkYXRhW5F1NWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9
encrypted_payment_descriptor=RRk1lEPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
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/wGXKw1xAKIkGLNkwcmraHH1U5tGHaT/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

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," [page 39](#), for details and field descriptions.

### To request an authorization for a Discover transaction:

---

- 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 14](#).
- Step 2** Set the `encrypted_payment_descriptor` field to  
Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
- 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=RRk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U
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/wGXKw1xAKIkGLNkwcmraHH1U5tGHaT/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**

See the [Relaxed Requirements for Address Data and Expiration Date](#) page and "Request Fields," page 39, for details and field descriptions.

**To request an authorization for a JCB transaction:**

- 
- 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 `Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1NRU5U`.
  - 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=555555555555xxxx
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/p6oJEy1gKIkJLNkwcsmrWHH1U5tGHST/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.

# API Fields

## Data Type Definitions

---

Data Type	Description
Date and time	Format is YYYY-MM-DDThhmmssZ, where: <ul style="list-style-type: none"><li>■ T separates the date and the time.</li><li>■ Z indicates Coordinated Universal Time (UTC), which equals Greenwich Mean Time (GMT).</li></ul> Example: 2020-08-11T22:47:57Z equals August 11, 2020, 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](#).

# Request Fields

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) <sup>2</sup>	CyberSource through VisaNet: String (40)  All other processors: String (60)
bill_address2	Additional address information.  <b>Example</b> 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) <sup>2</sup>	String (50)
bill_country	Country of the billing address. Use the two-character <i>ISO Standard Country Codes</i> .	ics_auth (R) <sup>2</sup>	String (2)
bill_state	State or province of the billing address. For an address in the U.S. or Canada, use the <i>State, Province, and Territory Codes for the United States and Canada</i> .	ics_auth (R) <sup>2</sup>	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 38. **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	<p>Postal code for the billing address. The postal code must consist of 5 to 9 digits.</p> <p>When the billing country is the U.S., the 9-digit postal code must follow this format: [5 digits][dash][4 digits]</p> <p><b>Example</b> 12345-6789</p> <p>When the billing country is Canada, the 6-digit postal code must follow this format: [alpha][numeric][alpha][space] [numeric][alpha][numeric]</p> <p><b>Example</b> A1B 2C3</p>	ics_auth (R) <sup>2</sup>	CyberSource through VisaNet: String (9) All other processors: String (10)
card_type	<p>Type of card to authorize. Possible values:</p> <ul style="list-style-type: none"> <li>■ 001: Visa</li> <li>■ 002: Mastercard</li> <li>■ 003: American Express</li> <li>■ 004: Discover</li> </ul>	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 38. **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	<p><b>Visa</b> Cryptogram for authorizations with payment network tokens. The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.</p> <p><b>American Express</b> For a 20-byte cryptogram, set this field to the cryptogram for authorizations with payment network tokens. For a 40-byte cryptogram, set this field to block A of the cryptogram for authorizations with payment network tokens (see "<a href="#">American Express Transaction</a>," page 24). The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.</p> <p><b>Discover</b> Cryptogram for authorizations with payment network tokens. The value for this field can be a 20 or 40-character hex binary. All cryptograms use one of these formats.</p> <p><b>CyberSource through VisaNet</b> The value for this field corresponds to the following data in the TC 33 capture file<sup>1</sup>:</p> <ul style="list-style-type: none"> <li>■ Record: CP01 TCR8</li> <li>■ Position: 77-78</li> <li>■ Field: CAVV version and authentication action.</li> </ul>	ics_auth (R)	String (40)
currency	Currency used for the order. For the possible values, see the <a href="#">ISO Standard Currency Codes</a> .	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) <sup>2</sup>	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 38. **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) <sup>2</sup>	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) <sup>2</sup>	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)
e_commerce_indicator	For an authorization with a payment network token. The values are required for the merchant decryption method (see " <a href="#">Option 1: Merchant Decryption</a> ," page 20).  Possible values: <ul style="list-style-type: none"> <li>■ aesk: American Express card type</li> <li>■ spa: Mastercard card type</li> <li>■ vbv: Visa card type mapped for Apple Pay transactions with e-commerce indicator of 5</li> <li>■ internet: Visa card type mapped for Apple Pay transactions with e-commerce indicator of 7</li> <li>■ dipb: Discover card type</li> </ul>	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 <b>paymentData</b> field of the <b>PKPaymentToken</b> structure. See the <a href="#">PassKit Framework Reference</a> .	ics_auth (R)	String (3072)
encrypted_payment_descriptor	Format of the encrypted payment data. The value for Apple Pay is: Rk1EPUNPTU1PTi5BUFBMRS5JTkFQUC5QQV1N RU5U	ics_auth (R)	String (128)
encrypted_payment_encoding	Encoding method used to encrypt the payment data:  Base64	ics_auth (R)	String (6)

- 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 38. **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)
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 (15)
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 <a href="#">Getting Started with CyberSource Advanced for the SCMP API</a> .	ics_auth (R)	String (50)
pa_specification_version	The 3D Secure version that you used for strong customer authentication (SCA); for example, 3D Secure version 1.0.2 or 2.0.0.	ics_auth (O)	String (20)
payment_network_token_assurance_level	Confidence level of the transaction. This value is assigned by the token service provider.  <b>Note</b> 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.  <b>Note</b> This field is supported only for FDC Compass.	ics_auth (O)	Integer (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 38. **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_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.  <b>Note</b> 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.  <b>Note</b> This unique ID differentiates digital solution transactions within the CyberSource platform for reporting purposes.	ics_auth (R)	String (3)
ucaf_authentication_ data	Cryptogram for authorizations with payment network tokens with Mastercard.	ics_auth (R)	String (32)
ucaf_collection_ indicator	Required field for authorizations with payment network tokens 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 38. **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	<p><b>Visa</b> Cryptogram for authorizations with payment network tokens. The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.</p> <p><b>American Express</b> For a 20-byte cryptogram, set this field to the cryptogram for authorizations with payment network tokens. For a 40-byte cryptogram, set this field to block A of the cryptogram for authorizations with payment network tokens (see "<a href="#">American Express Transaction</a>," page 24). The value for this field must be 28-character base64 or 40-character hex binary. All cryptograms use one of these formats.</p>	ics_auth (R)	String (40)
<p>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.</p> <p>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 38. <b>Important</b> It is your responsibility to determine whether a field is required for the transaction you are requesting.</p>			

## 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	<p>Identification code for the product.</p> <p>This field is required when the <b>product_code</b> value is not <code>default</code> or one of the values related to shipping and/or handling.</p>	ics_auth (See description)	String (255)
product_code	Type of product. This value is used to determine the product category: <code>electronic</code> , <code>handling</code> , <code>physical</code> , <code>service</code> , or <code>shipping</code> . The default is <code>default</code> .	ics_auth (See description)	String (255)
product_name	<p>Name of the product.</p> <p>This field is required when the <b>product_code</b> value is not <code>default</code> or one of the values related to shipping and/or handling.</p>	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 <b>product_code</b> value is not <code>default</code> 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)

## Reply Fields



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.

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 <a href="#">Credit Card Services Using the SCMP API</a> 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	<p>Mastercard service that was used for the transaction. Mastercard provides this value to CyberSource. Possible value:</p> <p>53: Mastercard card-on-file token service</p> <p>This field is returned only for CyberSource through VisaNet.</p> <p><b>CyberSource through VisaNet</b> The value for this field corresponds to the following data in the TC 33 capture file<sup>1</sup>:</p> <ul style="list-style-type: none"> <li>■ Record: CP01 TCR6</li> <li>■ Position: 133-134</li> </ul> <p>Field: Mastercard Merchant on-behalf service.</p> <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth	String (2)

<sup>1</sup> 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	<p>Result of the Mastercard card-on-file token service. Mastercard provides this value to CyberSource. Possible values:</p> <ul style="list-style-type: none"> <li>■ C: Service completed successfully.</li> <li>■ F: One of the following: <ul style="list-style-type: none"> <li>● Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 81 for an authorization or authorization reversal.</li> <li>● Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 01 for a tokenized request.</li> <li>● Token requestor ID is missing or formatted incorrectly.</li> </ul> </li> <li>■ I: One of the following: <ul style="list-style-type: none"> <li>● Invalid token requestor ID.</li> <li>● Suspended or deactivated token.</li> <li>● Invalid token (not in mapping table).</li> </ul> </li> <li>■ T: Invalid combination of token requestor ID and token.</li> <li>■ U: Expired token.</li> <li>■ W: Primary account number (PAN) listed in electronic warning bulletin. This field is returned only for CyberSource through VisaNet.</li> </ul> <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth	String (1)
auth_rcode	<p>Indicates whether the service request was successful. Possible values:</p> <ul style="list-style-type: none"> <li>■ -1: An error occurred.</li> <li>■ 0: The request was declined.</li> <li>■ 1: The request was successful.</li> </ul>	ics_auth	Integer (1)

<sup>1</sup> 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	<p>Mastercard service that was used for the transaction. Mastercard provides this value to CyberSource. Possible value:</p> <p>53: Mastercard card-on-file token service</p> <p>This field is returned only for CyberSource through VisaNet.</p> <p><b>CyberSource through VisaNet</b></p> <p>The value for this field corresponds to the following data in the TC 33 capture file<sup>1</sup>:</p> <ul style="list-style-type: none"> <li>■ Record: CP01 TCR6</li> <li>■ Position: 133-134</li> </ul> <p>Field: Mastercard Merchant on-behalf service.</p> <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth_ reversal	String (2)

<sup>1</sup> 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	<p>Result of the Mastercard card-on-file token service. Mastercard provides this value to CyberSource. Possible values:</p> <ul style="list-style-type: none"> <li>■ C: Service completed successfully.</li> <li>■ F: One of the following: <ul style="list-style-type: none"> <li>● Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 81 for an authorization or authorization reversal.</li> <li>● Incorrect Mastercard POS entry mode. The Mastercard POS entry mode should be 01 for a tokenized request.</li> <li>● Token requestor ID is missing or formatted incorrectly.</li> </ul> </li> <li>■ I: One of the following: <ul style="list-style-type: none"> <li>● Invalid token requestor ID.</li> <li>● Suspended or deactivated token.</li> <li>● Invalid token (not in mapping table).</li> </ul> </li> <li>■ T: Invalid combination of token requestor ID and token.</li> <li>■ U: Expired token.</li> <li>■ W: Primary account number (PAN) listed in electronic warning bulletin. This field is returned only for CyberSource through VisaNet.</li> </ul> <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth_ reversal	String (1)
auth_rflag	One-word description of the result of the entire request. See <a href="#">Credit Card Services Using the SCMP API</a> for a detailed list of <b>rflag</b> values.	ics_auth	String (50)
auth_rmsg	Message that explains the reply flag <b>auth_rflag</b> . 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)
<p><sup>1</sup> 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.</p>			

Table 5 Reply Fields (Continued)

Field	Description	Returned By	Data Type & Length
auth_transaction_qualification	<p>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:</p> <ul style="list-style-type: none"> <li>■ 1: Transaction qualifies for Mastercard authentication type 1.</li> <li>■ 2: Transaction qualifies for Mastercard authentication type 2.</li> </ul> <p>This field is returned only for CyberSource through VisaNet.</p> <p><b>CyberSource through VisaNet</b> The value for this field corresponds to the following data in the TC 33 capture file<sup>1</sup>:</p> <ul style="list-style-type: none"> <li>■ Record: CP01 TCR6</li> <li>■ Position: 132</li> </ul> <p>Field: Mastercard Member Defined service.</p> <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth	String (1)
card_suffix	<p>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.</p> <p><b>Note</b> This field is returned only for CyberSource through VisaNet and FDC Nashville Global.</p> <p><b>CyberSource through VisaNet</b> The value for this field corresponds to the following data in the TC 33 capture file<sup>1</sup>:</p> <ul style="list-style-type: none"> <li>■ Record: CP01 TCRB</li> <li>■ Position: 85</li> <li>■ Field: American Express last 4 PAN return indicator.</li> </ul>	ics_auth	String (4)
currency	<p>Currency used for the order. For the possible values, see the <a href="#">ISO Standard Currency Codes</a>.</p>	ics_auth	String (5)

<sup>1</sup> 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: <ul style="list-style-type: none"> <li>■ -1: An error occurred.</li> <li>■ 0: The request was declined.</li> <li>■ 1: The request was successful.</li> </ul>	ics_auth	Integer (1)
ics_rflag	One-word description of the result of the entire request. See <a href="#">Credit Card Services Using the SCMP API</a> for a detailed list of <b>rflag</b> values.	ics_auth	String (50)
ics_rmsg	Message that explains the reply flag <b>ics_rflag</b> . 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: <ul style="list-style-type: none"> <li>■ N: Nonregulated</li> <li>■ R: Regulated</li> </ul> This field is returned only for CyberSource through VisaNet. <p><b>Note</b> This field is returned only for CyberSource through VisaNet.</p>	ics_auth	String (1)
payment_network_token_assurance_level	Confidence level of the tokenization. This value is assigned by the token service provider. <p><b>Note</b> This field is returned only for CyberSource through VisaNet. and CyberSource through VisaNet and FDC Nashville Global.</p>	ics_auth	String (2)
payment_network_token_original_card_category	Mastercard product ID associated with the primary account number (PAN). For the possible values, see <a href="#">"Mastercard Product IDs"</a> in <i>Credit Card Services Using the SCMP API</i> . For the possible values, see <a href="#">"Mastercard Product IDs"</a> in <i>Credit Card Services for CyberSource through VisaNet Using the SCMP API</i> . <p><b>Note</b> This field is returned only for Mastercard transactions on CyberSource through VisaNet on CyberSource through VisaNet.</p>	ics_auth	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.

**Table 5 Reply Fields (Continued)**

Field	Description	Returned By	Data Type & Length
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.  <b>Note</b> This field is supported only for CyberSource through VisaNet and FDC Nashville Global.  <b>Note</b> 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.	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)
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 6 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 4 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.