

Paze

REST API

Integration Guide

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 Unpublished rights reserved under the copyright laws of the United States.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 and The Power of Payment are registered trademarks of Cybersource Corporation. Cybersource and Cybersource Decision Manager are trademarks and/or service marks of Cybersource Corporation. Visa, Visa International, Cybersource, the Visa logo, the Cybersource logo, and 3-D Secure are the registered trademarks of Visa International in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Version: 25.06.01

Contents

- Paze Integration Guide..... 4**
 - Recent Revisions to This Document..... 5
- Introduction to Paze..... 6**
 - Two Paze Decryption Methods.....6
 - Supported Payment Cards..... 7
- Getting Started with Paze..... 8**
 - Requirements for Integrating Paze Into Your System.....8
 - Obtain the Merchant Client ID for the Paze Service..... 8
 - Merchant Decryption Requires Generation of a Key Pair.....10
- Paze Response to a Complete API Call..... 11**
- Processing Paze Transactions with Merchant Decryption..... 14**
 - Authorize a Mastercard Payment on Paze with Merchant Decryption.....14
 - Basic Steps: Authorizing a Mastercard Payment on Paze with Merchant Decryption..... 14
 - Required Fields for Authorizing a Mastercard Payment on Paze with Merchant Decryption..... 15
 - REST Example: Authorize a Mastercard Payment on Paze with Merchant Decryption..... 16
 - Authorize a Visa Payment on Paze with Merchant Decryption..... 19
 - Basic Steps: Authorizing a Visa Payment on Paze with Merchant Decryption.....19
 - Required Fields for Authorizing a Visa Payment on Paze with Merchant Decryption..... 19
 - REST Example: Authorize a Visa Payment on Paze with Merchant Decryption.....21
- Processing Paze Transactions with Cybersource Decryption..... 24**
 - Authorize a Mastercard Payment on Paze with Cybersource Decryption.....24
 - Basic Steps: Authorizing a Mastercard Payment on Paze with Cybersource Decryption.....24
 - Required Fields for Authorizing a Mastercard Payment on Paze with Cybersource Decryption.....25
 - REST Example: Authorize a Mastercard Payment on Paze with Cybersource Decryption.....26
 - Authorize a Visa Payment on Paze with Cybersource Decryption..... 29
 - Basic Steps: Authorizing a Visa Payment on Paze with Cybersource Decryption.....29
 - Required Fields for Authorizing a Visa Payment on Paze with Cybersource Decryption.....29
 - REST Example: Authorize a Visa Payment on Paze with Cybersource Decryption.....30
- VISA Platform Connect: Specifications and Conditions for Resellers/Partners..... 33**

Paze Integration Guide

This section describes how to use this guide and where to find further information.

Audience and Purpose

The *Paze Integration Guide* is for banks and credit unions that want to offer the Paze online checkout experience and process Paze digital wallet payments through Cybersource. The guide describes how to process and search for Paze transactions. Processing is described for Paze digital wallet payment authorizations. The method you use to extract and decrypt Paze payment data depends on how you integrated Paze into your system.

Conventions



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

Related Documentation

Visit the [Cybersource documentation hub](#) to find additional technical documentation.

Customer Support

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

<http://support.visaacceptance.com>

Recent Revisions to This Document

25.06.01

Initial release.

Introduction to Paze

Paze is an online checkout option, or *digital wallet*, that enables you to offer customers a fast and secure way to make purchases online. If you integrate Paze into your ecommerce page, you can process Paze transactions in the same manner as your standard card processing and you have access to the consumers enrolled in Paze.

This guide describes how to submit transactions that originate from the Paze payment method to the Cybersource system for authorization.

Two Paze Decryption Methods

Integration hooks for two Paze decryption methods are built into the Cybersource payment management platform. The two decryption methods—merchant decryption and Cybersource decryption—handle Paze encrypted payment data differently. You will integrate the decryption method that best suits your technical development environment in terms of desired degree of exposure to, or control over, sensitive payment information.



Important: The Paze decryption method that you integrate determines how you will format your API request messages when you authorize a payment.

Merchant decryption

For merchant decryption, you (the merchant or the integrator) are responsible for the generation of the payment encryption keys, decryption of the payment response payload from Paze, and mapping the information—including the Paze payment token—to the corresponding Cybersource REST API fields for an authorization request. With merchant decryption, payment instrument details remain visible to you, and you control the technical development that decrypts this information.

Cybersource decryption

For Cybersource decryption, you integrate the Paze wallet directly on your checkout page. Cybersource creates and manages the Paze decryption keys and extracts and decrypts the sensitive payment information on your behalf. Having Cybersource process your Paze transactions reduces your PCI compliance burden. For information about the PCI Data Security Standard (DSS), see the [PCI Security Standards Council](#).



Important: The Paze decryption method that you integrate determines how you will format your API request messages when you authorize a payment.

Supported Payment Cards

Cybersource supports these cards with Paze on the Visa Platform Connect payment gateway, provided that the cards are enrolled in Paze with participating banks and credit unions:

- Mastercard
- Visa

Getting Started with Paze

This section covers information you need to know before you begin to integrate Paze into your system:

- Requirements for integrating Paze
- Steps for integrating Paze

Requirements for Integrating Paze Into Your System

- **Client ID.** You have obtained a client ID from Cybersource.
- **Acquirer.** You have an active contract with the acquirer.
- **Payment processor.** You are registered with a payment processor that connects with the acquirer.
- **Processor and acquirer relationship.** An acquirer might require you to use a payment processor that has an existing relationship with the acquirer.

For an overview of merchant financial institutions (acquirers), customer financial institutions (issuers), payment networks, and payment processors that work together to enable payment services, see the [Payments Developer Guide](#).

Obtain the Merchant Client ID for the Paze Service

Follow these steps to obtain the merchant's client ID for the Paze service:

1. In the left navigation panel, click the Payment Configuration icon.
2. Click **Digital Payment Solutions**. The Digital Payment Solutions page appears.
3. Click **Configure** next to the product name Paze. The Paze Configuration page appears.

Dotbank Services

Support
Notifications
shpandya

Dashboard
Invoicing
Virtual Terminal
Transaction Management
Fraud Management
Token Management
Pay by Link
Recurring Billing
Device Management
Tools
Portfolio Management
Reports
Analytics
Payment Configuration
Digital Payment Solutions
Key Management
Payer Authentication Configuration
Secure Acceptance Settings
Webhook Settings
Unified Checkout
Account Management

Portfolio ID partnerorg02 | Switch view

Payment Configuration / Digital Payment Solutions

Paze Configuration

Enter merchant details

Fields marked with a * are required.
Merchant legal name*

Merchant URL*

☒ By submitting the details above, I affirm that my organization: (i) has an agreement in place governing its use of Paze service; (ii) shall comply with Paze requirements and will implement Paze transaction method as per Paze specifications; and (iii) shall comply with the [Terms & Conditions available here](#).

Save
Cancel

4. Enter the merchant's legal name and the merchant URL.

5. Select the checkbox to indicate that the organization's agreements with these terms: (i) the organization has an agreement in place governing its use of the Paze service; (ii) the organization will comply with Paze requirements and will implement the Paze transaction method as per Paze specifications; and (iii) the organization will comply with the Terms and Conditions that are linked in the Paze Configuration page.
6. Click **Save**. The merchant's client ID appears on the page. The client ID can be used to enable Paze.

Dotbank Services

Support Notifications shpandya

Dashboard

Invoicing

Virtual Terminal

Transaction Management

Fraud Management

Token Management

Pay by Link

Recurring Billing

Device Management

Tools

Portfolio Management

Reports

Analytics

Payment Configuration

Digital Payment Solutions

Key Management

Payer Authentication Configuration

Secure Acceptance Settings

Webhook Settings

Unified Checkout

Account Management

Paze configuration is successful.

Payment Configuration / Digital Payment Solutions

Paze Configuration

Merchant details

Merchant legal name
Amazon.com LLC

Merchant URL
amazon.com

☒ By submitting the details above, I affirm that my organization: (i) has an agreement in place governing its use of Paze service; (ii) shall comply with Paze requirements and will implement Paze transaction method as per Paze specifications; and (iii) shall comply with the [Terms & Conditions available here](#).

Please use your Client ID below to enable Paze:

A238999

Done

Merchant Decryption Requires Generation of a Key Pair



Important: For merchant decryption only, you must generate a public/private key pair based on your merchant client ID.

yVmUxd3lOLWRzU0VnaUtQcUF5MzNzTzRSb0pKeDlxZ0U0eU5zdm50NDRJb0wycHFnlZKeDZHdDBZWGhRc
jYxMlFBLmNxY0xJcXRDMEpWekpLcWUuRlEtdlE1bUlyZmtLRWhGX2FYnkxIZzZnd2FuTlFRdmFWbFpQVGl
neUQ2WjVjd2RZb1JyazczZlFmdzduU2lRQnRoNS04MUdLcnlMVmFDLURqeThqUk04WmE2MF9md25GdENla
3NoVy03QmtOc1lSZXFGNnJERWVlVEg3TnRHYnpnenVhRmNqbUNTeUgyQlZFYjJ4RXRRV3B1TnlTWNrQ0F
KRmYxLXBwSUxJUKI5VjJCMdRvd3pNYXNjUHFsq1VPR3I3U00wWgtBOWRtelRCQlAteU92RU9qUEhFbDhGO
UlyY0QzMlZtaGRMZzJMeXdFMmNLV053alcybTJMdlF1NWVwS0NiWi1WRVU3ME1xc1VhZ1ljY3hBZnFkRnh
5N3MyWi0tQ25NbEVubjFhaUtUWUVyWGhGNk5LT1IwbjZ5eHhsUExsOWJ6ejZWX3NiRjZSazV1M3JnRDhDO
VU3UldPb2UyYmNleVlINlVVOEo4VzlrQy1peTJ3am4ySedyTkVMQ0F0MmRlUTdxeji3NlVCEff6V21TV0R
yZmJlS19fencxcEhjmKRFdlg5ZktDNmxJVUG2c2xWanBpU2N3MUFmNWlmSmhEMlFNZ2pJumFfTmoxUHgyY
09FcUxDX1NQTTjJzWDM4VFRuYUcxMV9JZVkb0b0p0bTlUS2diQ1EzSU9pdHlFYlFJcG9qSULHMHbVmtYeGx
oWENKOURCWnpqSDNxanNTTTJROHdIdlEzcjU3WDdtLXd4anphTDVMDmZwNWRnVTQ4UndjX1YxZkFhZ3pxT
1BobXNveldhbmQ4dmEYNHZoYlFiNXZxNHVfDXBLU0V6anQ3WlFLM2k2SS1lZExCWlhYT3VsOfk5cFo0MlM
2elB5S3oyTFZ4Y3JxMnlXNDdOVnpMVmhZZjRmTWducjhUVkNzWU1HVzhhNDcyOEdVaktXMMm0Z0R1adlxM
C1MVXZja1pMr244bTJYREF0dmxUUkhSeG16c3ZrN1lOTG9nV053T0NpZ0RuckdCU3lqLU5UdXVxm3BGNWp
IV1kyMTU5d2tNcHNVeFdGRkZRTnViZGZsSXBxdUpkLW81OHpWSku4T2x0NmhbQ2gtVlhuenpmOC1pY1BBM
VozeklKVGdlbF9WekpxOUpDdzd5WmEteWiWQ2RIWEVMT0s5Q1dIOGlGRWh6d2hzTW9nRDEwQUZCUG5GYUY
5bFZyWjJsc2pSWWF6dG5hbKRONERxd3J1dTR0d0pCS1lCaFF1WDNHT0VHVkJmX1hqMjh3ME95aWJyRTdnO
UQzZji2TudPb2t5bXlkrTA2NjVXclIyMj1zbV9yeURpdzNnZGEzTUp4M2wyYjBWMhraFRuSVdyLWhIU3l
jTXRxv2lQVTJ3Vzi5TXRtQlhpDENmMWZUMm13bFJYY2dmeFd3NnQyNlJSSNyTWE1VTBvdWE1a1Z3bWlod
TN2a1MwakhMY3dqUkZxeU5ESkp6OTlFdUFOaUU0Unr3MVBkZWU2cHA5TjJsXzRHSE9OMhdWVF9yZ0tvckd
2US1UU3pYNGRUTjAyZjV2c3dTZmZMdTlqVWEzZEpaB0J2MF14UG4zOE1UYlA3MG1MeU8zZWdhWlo3Zjlvd
WZuM0ZxdTlZR2dabjglazBWUWRLTHZVU05tdGRGNmJEQmYwd0t5T3p3cUNNRmdISnAtNkJUSXpXZGR5TWJ
aUkdJQlJNVG44WXN2SXptZlNHT3UxWHpkZFhbbXZ3Z2ZwSkhHdUhqRDA4dmFMZkRvZTVTNklmRV9CZDVle
XkyZUo4QnFWeEhrdlRJa0h1bEVIekNJVgx2bHdiUTl2b1BYczJUQXF4ZExoek43VElRTl9FRUpoUHg1OWd
ZZU1DY2FrMFJyZ253V01TZm5XY3VWZmI3NnNSVTZoa05xY0JUWHZBek10ZkxQ2I0ajlHaG9vNE4zNURFb
W5fNkZXduVkt0Y5cTM3YVRCRjhpWEQ0RWZva19NODV3NjhFSG01eFNPZGhzTENSUHRobkI3V1lScnJSTDV
ZOHhRUKVEcVpHMHRT2lVudl95UHZAQzItNzJUaWVPamp1ZXG3Y204aldFeWpHR0RiQ25uTEl4Ykr6VzBIN
G00YmJrTENaVVAAtVjdESVRBYk9qSeh4VWgwbmRxcC12bmFkZFc4X0FUcTVqaXBZala0YkQ0OglkeFN2WHl
VQkpJNDktTF1lZ19ZvmhnVzh1U2tjWjEtUWIybTharlpCTEFXUMf2MjZuSkt3cjFEWUdrX2xVZVZwbGpOd
19LdXl1VVBfnVUyYl1EdVNUTmtIVGVhNzZsaWlocDRuYmx1QmwwTmUzblhGR1pucVU4M1NMRDdkSF9GYlR
3UGoxZzF0YVglbTIazZlctNhcEtXWEZ5VnpNZG9BbVZqQ2lYwnltTUxnbFlUMHpFVUNKanNLNzJGV1N3X
3RNZ3NOa3p3Vm1KWVl0VVJPbUx5LXM0MmZxOTlULWRtVHFLMWJMRWJwYU9KtkZFVl1XOHczVWlVd2FWdDh
kM1R3azBfSTVqZ1lsNzNLeFJZZVhFdW9oN0d4ZFhXYXNhdKmxTVRPXzlw0VNeVl4S1lCWFJwcnBaeVBpZ
2N6aGRic24wR2pDb09CQzhVWnoxOG5zemFWaHhOUG9XdldqeHJqdmhnd0lFMTFSY0ZIQ19PMnlGS0FwRXJ
PWWF4MkxMcklWNl9qS2drS2dJazUzRG1TdW9oaG1WaWdnYU5ORHIyNm93WjFQVDQ1aFVqLU9tVERqUDNZV
XpfeEZ4QXV6cGY5M2E4SkVVeilzdGZ6ZUxRVVFQcHRTcVFKNHB2TFRlU2FGbWlPR2JWZzB2SGtIbHFmYmN
NRDB2OTJneTVFck1oaTczZndWRjFjbDIxY0djNk5RckcyQTNxSzf2SWRsMzVoSkY5ZEYwR0JacUtKeG13O
TJfQk90bHpyRUxQVHVlYXhZaEYtZUNOUHN4ZXdQUDZwZC5JcE5wQURnNndOM0VNZUJaODRxoWdnIn0.1lG
8pQGpvpGMSaUkIrxdkYB0vUbKYORna_iQc25EldokHh5naJEGnQ2sX4NWZmuntbt12adLqsoJ6H8x98Q6D
mGJ-_wWQBYm_0plU2fkV-VIe_gbOLY4ZyxMc01bf1qH8tZl-ICyALrleHmP73R-cFKijTGomguob1TFBOE
oojoKcVIMqWc46k6F1tIEQGR-cZSaPhMtiiJv4Xs6XQlvUYwCVGAJKe4jUWM43MSYG9R0R3tflUK2fZiA
ORUBA "

}

}

Depending on the Paze decryption method you integrated into your system, the payment bundle is decrypted and then used to make the authorization request to Cybersource. Cybersource then forwards the information to the payment network, which includes your processor and the relevant payment card company:

Merchant decryption

If you use the merchant decryption method, send the *decrypted payment details* to Cybersource in the **paymentInformation.tokenizedCard** fields of the authorization request that you send to Cybersource.

For detailed information, see [Processing Paze Transactions with Merchant Decryption \(on page 14\)](#).

Cybersource decryption

If you integrated the Cybersource decryption method, send the *encrypted payment object* to Cybersource in the **paymentInformation.fluidData.value** field of the authorization request that you send to Cybersource.

For detailed information, see [Processing Paze Transactions with Cybersource Decryption \(on page 24\)](#).

Processing Paze Transactions with Merchant Decryption

When you use the merchant decryption method, you are responsible for creating and managing the Paze decryption keys, extracting and decrypting payment information from the Paze payload, and mapping the required information to the Cybersource REST API fields for an authorization request.

This section of the guide shows you how to authorize Paze transactions using merchant decryption:

- How to authorize a Mastercard payment on Paze with merchant decryption
- How to authorize a Visa payment on Paze with merchant decryption

Authorize a Mastercard Payment on Paze with Merchant Decryption

The topics in this section show you how to authorize a Mastercard payment on Paze using the merchant decryption method.

Basic Steps: Authorizing a Mastercard Payment on Paze with Merchant Decryption

Follow these steps to request a Paze payment authorization with merchant decryption for Mastercard:

1. Create the request message with the required REST API fields.
 - Use the API fields listed in [Required Fields for Authorizing a Mastercard Payment on Paze with Merchant Decryption \(on page 15\)](#).
 - Refer to the example in [REST Example: Authorize a Mastercard Payment on Paze with Merchant Decryption \(on page 16\)](#).
2. Send the message to one of these endpoints:
 - Production: `POST https://api.cybersource.com/pts/v2/payments`
 - Test: `POST https://apitest.cybersource.com/pts/v2/payments`
3. Verify the response messages to make sure that the request was successful.

A 200-level HTTP response code indicates success. See the [Transaction Response Codes](#).

Required Fields for Authorizing a Mastercard Payment on Paze with Merchant Decryption

As a best practice, include these REST API fields in your request for an authorization with the merchant decryption implementation of Paze for Mastercard.



Important: Depending on your processor, your geographic location, and whether the relaxed address verification system (RAVS) is enabled for your account, some of these fields might not be required. It is your responsibility to determine whether an API field can be omitted from the transaction you are requesting.

For information about the relaxed requirements for address data and expiration dates in payment transactions, see the [Payments Developer Guide](#).

clientReferenceInformation.code

consumerAuthenticationInformation.ucafAuthenticationData

Set this field to the Token Secure cryptogram.

consumerAuthenticationInformation.ucafCollectionIndicator

Set this field to [2](#) for a Mastercard payment on Paze using merchant decryption.

orderInformation.amountDetails.currency

orderInformation.amountDetails.totalAmount

orderInformation.billTo.address1

orderInformation.billTo.administrativeArea

orderInformation.billTo.country

orderInformation.billTo.email

orderInformation.billTo.firstName

orderInformation.billTo.lastName

orderInformation.billTo.locality

orderInformation.billTo.postalCode

paymentInformation.tokenizedCard.cryptogram

Set this field to the network token cryptogram.

paymentInformation.tokenizedCard.expirationMonth

Set this field to the value from the payment network token expiration month.

paymentInformation.tokenizedCard.expirationYear

Set this field to the value from the payment network token expiration year.

paymentInformation.tokenizedCard.number

Set this field to the payment network token value.

paymentInformation.tokenizedCard.transactionType

Set this field to `1`.

paymentInformation.tokenizedCard.type

Set this field to `002` for Mastercard.

processingInformation.commerceIndicator

Set this field to `spa`.

processingInformation.paymentSolution

Set this field to `029` to specify the Paze payment solution.

REST Example: Authorize a Mastercard Payment on Paze with Merchant Decryption

Request

```
{
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "processingInformation": {
    "paymentSolution": "029",
    "commerceIndicator": "spa"
  },
  "paymentInformation": {
    "tokenizedCard": {
      "number": "5432543254325432",
      "expirationMonth": "12",
      "expirationYear": "2031",
      "cryptogram": "ABCDEFabcdefABCDEFabcdef0987654321234567",
      "transactionType": "1",
      "type": "002"
    }
  },
  "orderInformation": {
    "amountDetails": {
      "totalAmount": "100.00",
      "currency": "USD"
    }
  },
  "billTo": {
```

```

    "firstName": "Maya",
    "lastName": "Lee",
    "address1": "123MainSt",
    "locality": "SomeCity",
    "administrativeArea": "CA",
    "postalCode": "94404",
    "country": "US",
    "email": "maya.lee@email.world"
  }
},
"consumerAuthenticationInformation": {
  "ucafAuthenticationData": "ABCDEFabcdefABCDEFabcdef0987654321234567",
  "ucafCollectionIndicator": "2"
}
}

```

Response to a Successful Request

```

{
  "_links": {
    "authReversal": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/reversals"
    },
    "self": {
      "method": "GET",
      "href": "/pts/v2/payments/6234236182176225003004"
    },
    "capture": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/captures"
    }
  },
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "id": "6234236182176225003004",
  "orderInformation": {
    "amountDetails": {
      "authorizedAmount": "100.00",
      "currency": "USD"
    }
  },
  "paymentInformation": {
    "tokenizedCard": {
      "expirationYear": "2031",
      "prefix": "543254",

```

```

    "expirationMonth": "12",
    "suffix": "5432",
    "type": "002"
  },
  "card": {
    "type": "002"
  }
},
"pointOfSaleInformation": {
  "terminalId": "111111"
},
"processingInformation": {
  "paymentSolution": "029"
},
"processorInformation": {
  "approvalCode": "888888",
  "networkTransactionId": "123456789619999",
  "transactionId": "123456789619999",
  "responseCode": "100",
  "avs": {
    "code": "X",
    "codeRaw": "I1"
  }
},
"reconciliationId": "75729760OPN67ZFV",
"status": "AUTHORIZED",
"submitTimeUtc": "2021-06-11T15:00:18Z"
}

```

Authorize a Visa Payment on Paze with Merchant Decryption

The topics in this section show you how to authorize a Visa payment on Paze using the merchant decryption method.

Basic Steps: Authorizing a Visa Payment on Paze with Merchant Decryption

Follow these steps to request a Paze payment authorization with merchant decryption for Visa:

1. Create the request message with the required REST API fields.
 - Use the API fields listed in [Required Fields for Authorizing a Visa Payment on Paze with Merchant Decryption \(on page 19\)](#).
 - Refer to the example in [REST Example: Authorize a Visa Payment on Paze with Merchant Decryption \(on page 21\)](#).
2. Send the message to one of these endpoints:
 - Production: `POST https://api.cybersource.com/pts/v2/payments`
 - Test: `POST https://apitest.cybersource.com/pts/v2/payments`
3. Verify the response messages to make sure that the request was successful.

A 200-level HTTP response code indicates success. See the [Transaction Response Codes](#).

Required Fields for Authorizing a Visa Payment on Paze with Merchant Decryption

As a best practice, include these REST API fields in your request for an authorization with the merchant decryption implementation of Paze for Visa.



Important: Depending on your processor, your geographic location, and whether the relaxed address verification system (RAVS) is enabled for your account, some of these fields might not be required. It is your responsibility to determine whether an API field can be omitted from the transaction you are requesting.

For information about the relaxed requirements for address data and expiration dates in payment transactions, see the [Payments Developer Guide](#).

clientReferenceInformation.code

consumerAuthenticationInformation.ecommerceIndicator

Set this field to `vbv`.

orderInformation.amountDetails.currency

orderInformation.amountDetails.totalAmount

orderInformation.billTo.address1

orderInformation.billTo.administrativeArea

orderInformation.billTo.country

orderInformation.billTo.email

orderInformation.billTo.firstName

orderInformation.billTo.lastName

orderInformation.billTo.locality

orderInformation.billTo.postalCode

paymentInformation.tokenizedCard.cryptogram

Set this field to the network token cryptogram.

paymentInformation.tokenizedCard.expirationMonth

Set this field to the value from the payment network token expiration month.

paymentInformation.tokenizedCard.expirationYear

Set this field to the value from the payment network token expiration year.

paymentInformation.tokenizedCard.number

Set this field to the payment network token value.

paymentInformation.tokenizedCard.transactionType

Set this field to `1`.

paymentInformation.tokenizedCard.type

Set this field to `001` for Visa.

processingInformation.commerceIndicator

The mandate to use 3-D Secure for Paze transactions varies by geographic location. For Visa card transactions, 3-D Secure is called *Visa Secure*.

- If the transaction does not use 3-D Secure, set this field to the ECI value contained in the Paze response payload.
- If the transaction uses 3-D Secure, set this field to `vbv`.

processingInformation.paymentSolution

Set this field to [029](#) to specify the Paze payment solution.

REST Example: Authorize a Visa Payment on Paze with Merchant Decryption

Request

```
{
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "processingInformation": {
    "paymentSolution": "029",
    "commerceIndicator": "vbv"
  },
  "paymentInformation": {
    "tokenizedCard": {
      "number": "4111111111111111",
      "expirationMonth": "12",
      "expirationYear": "2031",
      "cryptogram": "AceY+igABPs3jdwNaDg3MAACAAA=",
      "transactionType": "1",
      "type": "001"
    }
  },
  "orderInformation": {
    "amountDetails": {
      "totalAmount": "100.00",
      "currency": "USD"
    },
    "billTo": {
      "firstName": "Maya",
      "lastName": "Lee",
      "address1": "123 Main St",
      "locality": "SomeCity",
      "administrativeArea": "CA",
      "postalCode": "94404",
      "country": "US",
      "email": "maya.lee@email.world"
    }
  },
  "consumerAuthenticationInformation": {
    "cavv": "AceY+igABPs3jdwNaDg3MAACAAA="
  }
}
```

Response to a Successful Request

```
{
  "_links": {
    "authReversal": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/reversals"
    },
    "self": {
      "method": "GET",
      "href": "/pts/v2/payments/6234236182176225003004"
    },
    "capture": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/captures"
    }
  },
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "id": "6234236182176225003004",
  "orderInformation": {
    "amountDetails": {
      "authorizedAmount": "100.00",
      "currency": "USD"
    }
  },
  "paymentInformation": {
    "tokenizedCard": {
      "expirationYear": "2031",
      "prefix": "411111",
      "expirationMonth": "12",
      "suffix": "1111",
      "type": "029"
    },
    "card": {
      "type": "029"
    }
  },
  "pointOfSaleInformation": {
    "terminalId": "111111"
  },
  "processingInformation": {
    "paymentSolution": "029"
  },
  "processorInformation": {
    "approvalCode": "888888",
    "networkTransactionId": "123456789619999",
    "transactionId": "123456789619999",
  }
}
```

```
"responseCode": "100",
"avs": {
  "code": "X",
  "codeRaw": "I1"
},
"reconciliationId": "75729760OPN67ZFV",
"status": "AUTHORIZED",
"submitTimeUtc": "2021-06-11T15:00:18Z"
}
```

Processing Paze Transactions with Cybersource Decryption

When you use the Cybersource decryption method, Cybersource creates and manages the Paze decryption keys and decrypts and extracts payment information from the Paze payload on behalf of the merchant. You are responsible for mapping the required information to the Cybersource REST API fields for an authorization request.

This section of the guide shows you how to process Paze transactions using the Cybersource decryption method:

- How to authorize a Mastercard payment on Paze with Cybersource decryption
- How to authorize a Visa payment on Paze with Cybersource decryption

Authorize a Mastercard Payment on Paze with Cybersource Decryption

The topics in this section show you how to authorize a Mastercard payment on Paze using the Cybersource decryption method.

Basic Steps: Authorizing a Mastercard Payment on Paze with Cybersource Decryption

Follow these steps to request a Paze payment authorization with Cybersource decryption for Mastercard:

1. Create the request message with the required REST API fields.
 - Use the API fields listed in [Required Fields for Authorizing a Mastercard Payment on Paze with Cybersource Decryption \(on page 25\)](#).
 - Refer to the example in [REST Example: Authorize a Mastercard Payment on Paze with Cybersource Decryption \(on page 26\)](#).
2. Send the message to one of these endpoints:

- Production: `POST https://api.cybersource.com/pts/v2/payments`

- Test: `POST https://apitest.cybersource.com/pts/v2/payments`

3. Verify the response messages to make sure that the request was successful.

A 200-level HTTP response code indicates success. See the [Transaction Response Codes](#).

Required Fields for Authorizing a Mastercard Payment on Paze with Cybersource Decryption

As a best practice, include these REST API fields in your request for an authorization with the Cybersource decryption implementation of Paze for Mastercard.



Important: Depending on your processor, your geographic location, and whether the relaxed address verification system (RAVS) is enabled for your account, some of these fields might not be required. It is your responsibility to determine whether an API field can be omitted from the transaction you are requesting.

For information about the relaxed requirements for address data and expiration dates in payment transactions, see the [Payments Developer Guide](#).

`clientReferenceInformation.code`

`orderInformation.amountDetails.currency`

`orderInformation.amountDetails.totalAmount`

`orderInformation.billTo.address1`

`orderInformation.billTo.administrativeArea`

`orderInformation.billTo.country`

`orderInformation.billTo.email`

`orderInformation.billTo.firstName`

`orderInformation.billTo.lastName`

`orderInformation.billTo.locality`

`orderInformation.billTo.postalCode`

`paymentInformation.fluidData.value`

Set this field to the Base64-encoded value returned in the Paze payload in the **`completeResponse`** property of the complete payment response object.

paymentInformation.tokenizedCard.transactionType

Set this field to **1**.

paymentInformation.tokenizedCard.type

Set this field to **002** for Mastercard.

processingInformation.paymentSolution

Set this field to **029** to identify Paze as the digital payment solution.

REST Example: Authorize a Mastercard Payment on Paze with Cybersource Decryption

Request

```
{
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "processingInformation": {
    "paymentSolution": "029"
  },
  "paymentInformation": {
    "fluidData": {
      "value": "eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9"
    },
    "tokenizedCard": {
      "type": "002",
      "transactionType": "1"
    }
  },
  "orderInformation": {
    "amountDetails": {
      "totalAmount": "100.00",
      "currency": "USD"
    },
    "billTo": {
      "firstName": "Maya",
      "lastName": "Lee",
      "address1": "123 Main St",
      "locality": "SomeCity",
      "administrativeArea": "CA",
      "postalCode": "94404",
      "country": "US",
      "email": "maya.lee@email.world"
    }
  }
}
```



```
}  
}  
}
```

Response to a Successful Request

```
{  
  "_links": {  
    "authReversal": {  
      "method": "POST",  
      "href": "/pts/v2/payments/6234236182176225003004/reversals"  
    },  
    "self": {  
      "method": "GET",  
      "href": "/pts/v2/payments/6234236182176225003004"  
    },  
    "capture": {  
      "method": "POST",  
      "href": "/pts/v2/payments/6234236182176225003004/captures"  
    }  
  },  
  "clientReferenceInformation": {  
    "code": "1234567890"  
  },  
  "id": "6234236182176225003004",  
  "orderInformation": {  
    "amountDetails": {  
      "authorizedAmount": "100.00",  
      "currency": "USD"  
    }  
  },  
  "paymentInformation": {  
    "tokenizedCard": {  
      "expirationYear": "2031",  
      "prefix": "128945",  
      "expirationMonth": "12",  
      "suffix": "2398",  
      "type": "002"  
    },  
    "card": {  
      "type": "002"  
    }  
  },  
  "pointOfSaleInformation": {  
    "terminalId": "111111"  
  },  
  "processingInformation": {
```

```
    "paymentSolution": "029"
  },
  "processorInformation": {
    "approvalCode": "888888",
    "networkTransactionId": "123456789619999",
    "transactionId": "123456789619999",
    "responseCode": "100",
    "avs": {
      "code": "X",
      "codeRaw": "I1"
    }
  },
  "reconciliationId": "75729760OPN67ZFBV",
  "status": "AUTHORIZED",
  "submitTimeUtc": "2021-06-11T15:00:18Z"
}
```

Authorize a Visa Payment on Paze with Cybersource Decryption

The topics in this section show you how to authorize a Visa payment on Paze using the Cybersource decryption method.

Basic Steps: Authorizing a Visa Payment on Paze with Cybersource Decryption

Follow these steps to request a Paze payment authorization with Cybersource decryption for Visa:

1. Create the request message with the required REST API fields.
 - Use the API fields listed in [Required Fields for Authorizing a Visa Payment on Paze with Cybersource Decryption \(on page 29\)](#).
 - Refer to the example in [REST Example: Authorize a Visa Payment on Paze with Cybersource Decryption \(on page 30\)](#).
2. Send the message to one of these endpoints:
 - Production: `POST https://api.cybersource.com/pts/v2/payments`
 - Test: `POST https://apitest.cybersource.com/pts/v2/payments`
3. Verify the response messages to make sure that the request was successful.

A 200-level HTTP response code indicates success. See the [Transaction Response Codes](#).

Required Fields for Authorizing a Visa Payment on Paze with Cybersource Decryption

As a best practice, include these REST API fields in your request for an authorization with the Cybersource decryption implementation of Paze for Visa.



Important: Depending on your processor, your geographic location, and whether the relaxed address verification system (RAVS) is enabled for your account, some of these fields might not be required. It is your responsibility to determine whether an API field can be omitted from the transaction you are requesting.

For information about the relaxed requirements for address data and expiration dates in payment transactions, see the [Payments Developer Guide](#).

clientReferenceInformation.code

orderInformation.amountDetails.currency

orderInformation.amountDetails.totalAmount

orderInformation.billTo.address1

orderInformation.billTo.administrativeArea

orderInformation.billTo.country

orderInformation.billTo.email

orderInformation.billTo.firstName

orderInformation.billTo.lastName

orderInformation.billTo.locality

orderInformation.billTo.postalCode

paymentInformation.fluidData.value

Set this field to the Base64-encoded value returned in the Paze payload in the **completeResponse** property of the complete payment response object.

paymentInformation.tokenizedCard.transactionType

Set this field to **1**.

paymentInformation.tokenizedCard.type

Set this field to **001** for Visa.

processingInformation.paymentSolution

Set this field to **029** to identify Paze as the digital payment solution.

REST Example: Authorize a Visa Payment on Paze with Cybersource Decryption

Request

```
{
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "processingInformation": {
    "paymentSolution": "001"
  },
  "paymentInformation": {
    "fluidData": {
```

```

    "value": "eyJkYXRhW5FINWZqVjfkak1NdVNSaE96dWF2ZGVyb2c9PSJ9"
  },
  "tokenizedCard": {
    "type": "001",
    "transactionType": "1"
  }
},
"orderInformation": {
  "amountDetails": {
    "totalAmount": "100.00",
    "currency": "USD"
  },
  "billTo": {
    "firstName": "Maya",
    "lastName": "Lee",
    "address1": "123 Main St",
    "locality": "SomeCity",
    "administrativeArea": "CA",
    "postalCode": "94404",
    "country": "US",
    "email": "maya.lee@email.world"
  }
}
}

```

Response to a Successful Request

```

{
  "_links": {
    "authReversal": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/reversals"
    },
    "self": {
      "method": "GET",
      "href": "/pts/v2/payments/6234236182176225003004"
    },
    "capture": {
      "method": "POST",
      "href": "/pts/v2/payments/6234236182176225003004/captures"
    }
  },
  "clientReferenceInformation": {
    "code": "1234567890"
  },
  "id": "6234236182176225003004",
  "orderInformation": {

```

```

    "amountDetails": {
      "authorizedAmount": "100.00",
      "currency": "USD"
    },
    "paymentInformation": {
      "tokenizedCard": {
        "expirationYear": "2031",
        "prefix": "411111",
        "expirationMonth": "12",
        "suffix": "1111",
        "type": "029"
      },
      "card": {
        "type": "029"
      }
    },
    "pointOfSaleInformation": {
      "terminalId": "111111"
    },
    "processingInformation": {
      "paymentSolution": "001"
    },
    "processorInformation": {
      "approvalCode": "888888",
      "networkTransactionId": "123456789619999",
      "transactionId": "123456789619999",
      "responseCode": "100",
      "avs": {
        "code": "X",
        "codeRaw": "I1"
      }
    }
  }
}

```


VISA Platform Connect: Specifications and Conditions for Resellers/Partners

The following are specifications and conditions that apply to a Reseller/Partner enabling its merchants through Cybersource for Visa Platform Connect (“VPC”) processing. Failure to meet any of the specifications and conditions below is subject to the liability provisions and indemnification obligations under Reseller/Partner’s contract with Visa/Cybersource.

1. Before boarding merchants for payment processing on a VPC acquirer’s connection, Reseller/Partner and the VPC acquirer must have a contract or other legal agreement that permits Reseller/Partner to enable its merchants to process payments with the acquirer through the dedicated VPC connection and/or traditional connection with such VPC acquirer.
2. Reseller/Partner is responsible for boarding and enabling its merchants in accordance with the terms of the contract or other legal agreement with the relevant VPC acquirer.
3. Reseller/Partner acknowledges and agrees that all considerations and fees associated with chargebacks, interchange downgrades, settlement issues, funding delays, and other processing related activities are strictly between Reseller and the relevant VPC acquirer.
4. Reseller/Partner acknowledges and agrees that the relevant VPC acquirer is responsible for payment processing issues, including but not limited to, transaction declines by network/issuer, decline rates, and interchange qualification, as may be agreed to or outlined in the contract or other legal agreement between Reseller/Partner and such VPC acquirer.

DISCLAIMER: NEITHER VISA NOR CYBERSOURCE WILL BE RESPONSIBLE OR LIABLE FOR ANY ERRORS OR OMISSIONS BY THE Visa Platform Connect ACQUIRER IN PROCESSING TRANSACTIONS. NEITHER VISA NOR CYBERSOURCE WILL BE RESPONSIBLE OR LIABLE FOR RESELLER/PARTNER BOARDING MERCHANTS OR ENABLING MERCHANT PROCESSING IN VIOLATION OF THE TERMS AND CONDITIONS IMPOSED BY THE RELEVANT Visa Platform Connect ACQUIRER.