

# Google Pay

Simple Order API  
Streamline



## Cybersource Contact Information

For general information about our company, products, and services, go to <https://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: <https://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 in 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 Unpublished rights reserved under the copyright laws of the United States.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.

## Confidentiality Notice

This document is furnished to you solely in your capacity as a client of Cybersource and as a participant in the Visa payments system.

By accepting this document, you acknowledge that the information contained herein (the "Information") is confidential and subject to the confidentiality restrictions contained in Visa's operating regulations and/or other confidentiality agreements, which limit our use of the Information. You agree to keep the Information confidential and not to use the Information for any purpose other than its intended purpose and in your capacity as a customer of Cybersource or as a participant in the Visa payments system. The Information may only be disseminated within your organization on a need-to-know basis to enable your participation in the Visa payments system. Please be advised that the Information may constitute material non-public information under U.S. federal securities laws and that purchasing or selling securities of Visa Inc. while being aware of material non-public information would constitute a violation of applicable U.S. federal securities laws.

## Revision

Version: 23.02

# Contents

|                                                       |           |
|-------------------------------------------------------|-----------|
| <b>Recent Revisions to This Document.....</b>         | <b>4</b>  |
| <b>About This Guide.....</b>                          | <b>5</b>  |
| <b>Introduction.....</b>                              | <b>6</b>  |
| Payment Network Tokens.....                           | 6         |
| Prerequisite Requirements.....                        | 6         |
| Supported Processors.....                             | 7         |
| How Google Pay Works.....                             | 7         |
| Additional Services.....                              | 8         |
| Follow-on Transactions.....                           | 9         |
| <b>Formatting Encrypted Payment Data.....</b>         | <b>10</b> |
| Configuring Google Pay.....                           | 10        |
| Formatting Payment Blobs.....                         | 11        |
| Examples of Google Pay Responses.....                 | 11        |
| <b>Google Pay Authorizations.....</b>                 | <b>13</b> |
| Required Fields for a Google Pay Authorization.....   | 13        |
| Simple Order Example: Google Pay Authorization.....   | 14        |
| <b>Follow-on Services.....</b>                        | <b>16</b> |
| Capture.....                                          | 16        |
| Required Fields for Capturing an Authorization.....   | 16        |
| Simple Order Example: Capturing an Authorization..... | 17        |
| Sales.....                                            | 17        |
| Required Fields for Processing a Sale.....            | 17        |
| Simple Order Example: Processing a Sale.....          | 18        |
| <b>Follow-on Transactions.....</b>                    | <b>20</b> |

# Recent Revisions to This Document

## 23.02

Updated the Google Pay authorizations required fields list. See [Required Fields for a Google Pay Authorization](#) on page 13.

Added follow-on transaction support to the Google Pay authorization section. See [Google Pay Authorizations](#) on page 13.

## 23.01

This revision contains only editorial changes and no technical updates.

## 22.02

This revision contains only editorial changes and no technical updates.

## 22.01

Added Diners Club and Discover as supported card types for Barclays. See [Supported Processors](#).

## 21.06

Bug update. Document erroneously said the GPN processor allows split-shipments.

## 21.05

Changed the name of the **merchantURL** field to **merchantDomainName**.

# About This Guide

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

## Audience and Purpose

This document is written for merchants who want to enable customers to use Google Pay to pay for in-app purchases. This document provides an overview of integrating the Google API and describes how to request the Cybersource API to process an authorization.

This document describes the Google Pay service and the Cybersource API. You must request the Google API to receive the customer's encrypted payment data before requesting the Cybersource API to process the transaction.

## Conventions

The following special statements are used in this document:



### Important

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

## Related Documentation

For further technical documentation, visit the Cybersource Technical Documentation Portal:

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

## Customer Support

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

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

# Introduction

You can use the Cybersource platform to process and manage Google Pay transactions.

## Google Pay Overview

Google Pay is a simple, secure in-app mobile and Web payment solution. You can choose Cybersource to process Google Pay transactions through all e-commerce channels. You can simplify your payment processing by allowing Cybersource to decrypt the payment data for you during processing.

This method integrates simply and enables you to process transactions without seeing the payment network token and transaction data.

1. Using the Google API, request the customer's encrypted payment data.
2. Using the Cybersource API, construct and submit the authorization request, and include the encrypted payment data from the Google Pay callback.
3. Cybersource decrypts the encrypted payment data to create the payment network token and processes the authorization request.

## Payment Network Tokens

Authorizations with payment network tokens enable you to securely request a payment transaction with a payment network token instead of a customer's primary account number (PAN).

The payment network token is included in the customer's encrypted payment data, which is returned by the payment processor.

For information about authorizations with payment network tokens, see the [Authorizations with Payment Network Tokens Guide](#).

## Prerequisite Requirements

Before using Google Pay, you must have:

- A Cybersource merchant evaluation account.
  - To register, go to: <https://www.cybersource.com/register/>
- A [merchant evaluation account](#) with a supported processor. See [Supported Processors](#) on page 7.
- A Google developer account.
- Google Pay APIs embedded into your application or website. For details about integrating Google Pay, see the [Google Pay API](#) documentation.

## Supported Processors

| Processor  | Card Types         | Optional Features  |
|------------|--------------------|--------------------|
| Streamline | Mastercard<br>Visa | Recurring Payments |

## How Google Pay Works

The following figure describes the Google Pay workflow:



1. The customer chooses the Google Pay button. Using the Google API, your system initiates the Google Pay request identifying Cybersource as your payment gateway, passing your Cybersource merchant ID as the gateway merchant ID.
2. The customer confirms the payment. The Google API contacts Google Pay services to retrieve the consumer's payment parameters.
3. If the customer's selected payment credentials are tokenized, or you are tokenizing new payment credentials, the Google Pay service contacts the appropriate payment network to retrieve the appropriate cryptogram.
4. The payment network returns the appropriate token and cryptogram to the Google Pay service.
5. Google creates encrypted payment data using the gateway-specific key that is supplied in the Wallet request and includes it in the Google API response.
6. The Google Pay callback returns the encrypted payment data.
7. Your system prepares the Google Pay response information for submission to the Cybersource service.
8.
  - a. Cybersource sends the authorization request to the acquirer.
  - b. The acquirer processes the request from Cybersource and creates the payment network authorization request.
  - c. The payment network processes the request from the acquirer and creates the issuer authorization request.
  - d. The issuer processes the request from the payment network. The issuer looks up the payment information and returns an approved or declined authorization message to the payment network.
  - e. The payment network returns the authorization response to the acquirer.
  - f. The acquirer returns the authorization response to Cybersource.
9. Cybersource returns the authorization response to your system.
10. Your system returns the authorization response to the payment application.
11. The payment application displays the confirmation or decline message to the customer.
  - a. The acquirer submits the settlement request to the issuer for funds.
  - b. The issuer supplies the funds to the acquirer for the authorized transactions.

## Additional Services

These additional services can be used with Google Pay.

### Follow-on Services

After the authorization is requested, you can request follow-on services to complete the transaction. For more information on these services, see [Follow-on Services](#).

#### Authorized Reversal

**An authorized reversal is a follow-on service that uses the request ID returned from the previous authorization. An authorization 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.

### **Capture**

A capture is 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.

## **Follow-on Transactions**

After the payment transaction is complete, additional follow-on transactions can be made as Merchant-Initiated Transactions (MITs).

For more information on how to process MITs, see [Merchant-Initiated Transactions](#).

MITs include:

- Delayed Authorizations
- Incremental Transactions
- Installment Payments
- No-Show Transactions
- Reauthorizations
- Recurring Transactions
- Resubmissions
- Unscheduled Transactions

# Formatting Encrypted Payment Data

This section shows you how to format encrypted payment data using these procedures:

- [Configuring Google Pay](#)
- [Formatting Payment Blobs](#)

## Configuring Google Pay

You must provide your Cybersource merchant ID to Google in order to ensure proper encryption of the Google Pay payload and authenticity of the request.

For a Google Pay tutorial, see [Google Pay for Payments](#).

Set the gateway and gateway merchant ID to the appropriate indicators. The following code examples show how to configure the **PaymentMethodTokenizationParameters** object using Cybersource as the gateway.

### Example: Java Code

```
.setPaymentMethodTokenizationType(WalletConstants.PAYMENT_METHOD_TOKENIZATION_TYPE_PAYMENT_GATEWAY)
.addParameter("gateway", "")
.addParameter("gatewayMerchantId", "[yourCybersourceMID]")
```

### Example: JavaScript Code

```
tokenizationType: 'PAYMENT_GATEWAY',
parameters: {
  gateway: 'cybersource',
  gatewayMerchantId: '[yourCybersourceMID]'
```

# Formatting Payment Blobs



## Important

This section is only applicable if you are using the Cybersource decryption method.

To prepare the google payload for submission to Cybersource, you must extract the token data element from the Google Pay payload and encode the token data element using Base64.

These samples can be used to Base64-encode payment responses:

## JavaScript

```
let token = paymentData.paymentMethodData.tokenizationData.token;
console.log(token);
var enc=window.btoa(token);
```

## Android with Java

This sample uses the Android Studio Base64 utility.

```
public static <outputString> encodeToString (byte[] <inputToken>, int DEFAULT)
```

## Apple iPhone with Swift 3

This sample requires the Foundation utility.

```
extension String {
    func base64Encoded() -> <outputString>
    {
        if let data = self.data(using: .utf8) {
            return data.base64EncodedString()
        }
        return nil
    }
}
```

## Examples of Google Pay Responses

Decrypted Google Pay Response

```
{
  "signature": "MEUCIQDhTxxHqWY8pXB9hpYxaSK5jFgsqpG2E1rX77QXssK8tAIgUBvYYAI/
bnBS8T/Tfxnm2AF981Mv5y0pHyGexM5dMJk\u003d"; "protocolVersion": "ECv1";
  "signedMessage": {
    "encryptedMessage": "\
odyUGGA7B+b1letYcJbS43AQUFQJpWEFCN4UuUEXQ5LX0V/
XcLwKELXcB95nMnmPO91M2KGp13FYsL768ccCzAjBGLYF+
fugcJTcvkrUhcNSyXr7hfw12BEsrweqJM6I7Vs51frPAukRJeLDQG4FxmTLW49QyP8vIZC+
tz2c+Z3zozzI5oB9jE8fA2do1Fa13Cu6gXqdKH\
IHRh7UniLUuTy+0G5FQV2pwST2uBSNNkZhb8WYJDHbxBjz0UebVP+
ObmT5cc8AKU5dghRdfr4GKpEZ4EBzB90 BPxLqYHpopriJ61bFgFVsQQ6V/
8HBqQ7ImIMH5y7G8p8qAFkWNb78ZcL0Fh5BjXojkxGoFp2gjAsrhhttHAFbe3WQBUPkwJu09V/
```

```
6\MyJpCSrpMHFouFV/dj0SYjQ+xI0971CHZec7jQrAhISLWZ9DZkuMvGKPWpu0CKn2XqTXQ=\
"\ephemeralPublicKey\":"\
"MFkwEwYHkoZIZj0CAQYIKoZIZj0DAQcDQgAEEnn4yYj0N6x1XO8\8j7\
4jvmLJCYAqgXLwP1FhjuTgIM9oCtPijZfI9so2QEOs2ZnVp3D0d13JYIDVe+396KkAQ==\
"\tag\":"DRp cc+YQ33RNgstcxztnJbMJnirbU5DW3dStjfhFiwc=\"}"}
```

#### Base64-Encoded Google Pay Response

```
eyJzaWduYXR1cmUiOiJNRVVDVSVFEaFR4aEhxd1k4cFhCOWhwWXhhU0s1akZnc3FwRzJ
FMXJYNzdRWHNzSzh0QUlnVUJ2WV1BSS9ibkJTOFQvVGZ4bm0yQUY5ODFNdjV5MHBIeU
dleE01ZE1Ka1x1MDAzZCIsInByb3RvY29sVmVyc2lubiI6IkVDdjEiLCJzaWduZWRRNz
XNzYWdlIjoie1wiZW5jcnlwdGVkTWVzc2FnZVwiOlwib2R5VUdHQTdCK2JsbGV0WWNK
YlM0M0FRVUZRSnBXRUZDTjRVdVVFfE1TFgwXC9YY0x3S0VsWGNCOTVvTW5tUE85bE0
yS0dwMTNGWXNMNzY4Y2NDekFqQkdMWUYrZnVnY0pUY3Zrc1VoY05TeVhyN2h3ZjEyQk
VzcndlcUpNNkk3VnM1bGZyUEF1a1JKZUxEUUC0RnhtVExXND1ReVA4dklaQyt0ejJk
1ozem96ekk1b0I5akU4ZkEyZG9sRmExM0N1NmdYcWRLSFwvSUhSaDdVbm1MVXVUeSsw
RzVGUVYycHdTVdJ1Q1NOTmtaaGI4V1lKREhieEJqejBVZWJWUCtPYm1UNWNjOEFLTV
kZ0hSZGZyNEdLcEVaNEVCekI5MEJQeExxWUhw3ByaUo2bGJGZ0ZWc1FRNlwwOEhCcV
E3SW1JTUg1eTdHOHA4cUFGa1duQjc4WmNMMEZoNUJqWG9qa3hHb0ZwMmdqQXNyaGh0d
EhBRmJlM1dRQnVQa3dKdTA5XC82XC9NeUpwQ1NycE1IRm91RlwwZGowU1lqUSt4STA5
N2xDSFp1YzdqUXJBaElTTfdaOURaa3VNdkdLUfddwTBDS24yWHFUFWE9XCIsXCJlcGh
lbWVyYXQdWJsaWNLZX1cIjpcIk1Ga3dFd1lIS29aSXpqMENBUV1JS29aSXpqMERBUW
NEUWdBRW5uNHlqeTBONnhsWE84XC84ajdcLzRqdm1MSkNZQXFnWE3UDFGaGp1VGdJT
TlvQ3RQaWpaZkk5c28yUUVPCzJab1ZwM0QwZGwzSl1JRfZ1KzM5NktrQVE9PVwiLFwi
dGFuXCI6XCJEUbnBjYytZUTMzUk5nc1RjeHp0bkpiTUuaXJiVTVEVzNkU3RqZmhGaXd
jPVwifSJ9
```

# Google Pay Authorizations

This section shows you how to make a successful authorization request. After you send the request, check the response messages to make sure that the request was successful. A value of `ACCEPT` for the **decision** field indicates success. For information about response codes, see [Reason Codes for the Simple Order API](#).

## Follow-on Transactions

After the initial transaction is complete, additional follow-on transactions can be made as Merchant-Initiated Transactions (MITs). For more information on how to process MITs, see [Merchant-Initiated Transactions](#).

## Endpoint

Set the **ccAuthService\_run** field to `true`.

Send the request to `https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`.

## Required Fields for a Google Pay Authorization

Include these required fields to request a successful authorization.

**billTo\_city**

**billTo\_country**

**billTo\_email**

**billTo\_firstName**

**billTo\_lastName**

**billTo\_postalCode**

**billTo\_state**

**billTo\_street1****ccAuthService\_run**Set this field value to `true`.**encryptedPayment\_data**

Set this field value to the string value generated from the full wallet response.

**merchantID****merchantReferenceCode****paymentSolution**Set this field value to `012`.**purchaseTotals\_currency****purchaseTotals\_grandTotalAmount**

## Related Information

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

# Simple Order Example: Google Pay Authorization

## Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.121">
  <merchantID>demomerchant</merchantID>
  <merchantReferenceCode>demorefnum</merchantReferenceCode>
  <billTo>
    <firstName>James</firstName>
    <lastName>Smith</lastName>
    <street1>1295 Charleston Road</street1>
    <city>Test City</city>
    <state>CA</state>
    <postalCode>99999</postalCode>
    <country>US</country>
    <email>demo@example.com</email>
  </billTo>
  <purchaseTotals>
    <currency>USD</currency>
    <grandTotalAmount>5.00</grandTotalAmount>
  </purchaseTotals>
  <encryptedPayment>
    <data>ABCDEFabcdefABCDEFabcdef0987654321234567</data>
  </encryptedPayment>
  <ccAuthService run="true"/>
  <paymentSolution>012</paymentSolution>
</requestMessage>
```

## Response for a Successful Request

```
<replyMessage>
```

```

<merchantReferenceCode>demorefnum</merchantReferenceCode>
<requestID>44658403407650000001541</requestID>
<decision>ACCEPT</decision>
<reasonCode>100</reasonCode>
<requestToken>Ahj/7wSR5C/4Icd2fdAKakGLadfg5535r/ghx3Z90AoBj3u</requestToken>
<purchaseTotals>
  <currency>USD</currency>
</purchaseTotals>
<ccAuthReply>
  <reasonCode>100</reasonCode>
  <amount>5.00</amount>
  <authorizationCode>888888</authorizationCode>
  <avsCode>X</avsCode>
  <avsCodeRaw>I1</avsCodeRaw>
  <authorizedDateTime>2015-11-03T20:53:54Z</authorizedDateTime>
  <processorResponse>100</processorResponse>
  <reconciliationID>11267051CGJSMQDC</reconciliationID>
</ccAuthReply>
<token>
  <prefix>294672</prefix>
  <suffix>4397</suffix>
  <expirationMonth>08</expirationMonth>
  <expirationYear>2021</expirationYear>
</token>
</replyMessage>

```

# Follow-on Services

This section provides information about and procedures for requesting these follow-on services:

- **Authorization Reversal:** A follow-on service that uses the request ID returned from the previous authorization. An authorization 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.
- **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.

## Capture

This section describes how to capture an authorized transaction.

### Endpoint

Set the **ccCaptureService\_run** field to `true`.

Send the request to <https://ics2ws.ic3.com/commerce/1.x/transactionProcessor>.

### Required Fields for Capturing an Authorization

Use these required fields for capturing an authorization.

**ccCaptureService\_authRequestID**

**ccCaptureService\_run**

**merchantID**

**merchantReferenceCode**Set to `merchant_ref_number` value used in corresponding authorization request.**purchaseTotals\_currency****purchaseTotals\_grandTotalAmount**

## Related Information

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

## Simple Order Example: Capturing an Authorization

### Request

```
ccCaptureService_authRequestID=6629978499572480812782
ccCaptureService_run=true
merchantID=npr_paymentech
merchantReferenceCode=TC42703-1
purchaseTotals_grandTotalAmount=100.00
```

### Response for a Successful Request

```
ccCaptureReply_amount=100.00
ccCaptureReply_requestDateTime=2022-09-12T173947Z
decision=ACCEPT
merchantReferenceCode=TC42703-1
purchaseTotals_currency=USD
requestID=6630043878211258349460
```

# Sales

This section shows you how to process a sale transaction.

A sale transaction combines an authorization and a capture into a single transaction.

## Endpoint

Set the **ccAuthService\_run** field to `true`, and the **ccCaptureService\_run** field to `true`.

Send the request to `https://ics2ws.ic3.com/commerce/1.x/transactionProcessor`.

## Required Fields for Processing a Sale

Use these required fields for processing a sale.

**billTo\_city****billTo\_country****billTo\_email****billTo\_firstName**

**billTo\_lastName**  
**billTo\_postalCode**  
**billTo\_state**  
**billTo\_street1**  
**card\_accountNumber**  
**card\_cardType**  
**card\_expirationMonth**  
**card\_expirationYear**  
**ccAuthService\_commerceIndicator**  
**ccAuthService\_run** Set this field to `true`.  
**ccCaptureService\_run** Set this field to `true`.  
**merchantID**  
**purchaseTotals\_currency**  
**purchaseTotals\_grandTotalAmount**

## Related Information

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

## Simple Order Example: Processing a Sale

### Request

```

ccAuthService_run=true
ccCaptureService_run=true
merchantID=Napa Valley Vacations
merchantReferenceCode=482046C3A7E94F5
billTo_firstName=John
billTo_lastName=Doe
billTo_street1=1295 Charleston Rd.
billTo_city=Mountain View
billTo_state=CA
billTo_postalCode=94043
billTo_country=US
billTo_phoneNumber=650-965-6000
billTo_email=jdoe@example.com
item_0_unitPrice=49.95
item_0_quantity=1
purchaseTotals_currency=USD
card_expirationMonth=12
card_expirationYear=2015
card_accountNumber=4111111111111111

card_cardType=001

```

## Response for a Successful Request

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

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

# Follow-on Transactions

After the payment transaction is complete, additional follow-on transactions can be made as Merchant-Initiated Transactions (MITs).

For more information on how to process MITs, see [Merchant-Initiated Transactions](#).

MITs include:

- Delayed Authorizations
- Incremental Transactions
- Installment Payments
- No-Show Transactions
- Reauthorizations
- Recurring Transactions
- Resubmissions
- Unscheduled Transactions