



# 使用手冊

---

3.1.5

## 信用卡付款模組 使用 **Payment Toolkit**

### **Java version**

Version 3.1.5

Date: 2019/12

©Copyright 2017, HiTRUST.com Inc., All rights reserved

Provided By Solution Development Dept, Regional Business Development Division.

## 目錄

<b>Chapter 1、</b>	<b>簡介.....</b>	<b>3</b>
<b>Chapter 2、</b>	<b>金流產品介紹.....</b>	<b>4</b>
<b>Chapter 3、</b>	<b>環境需求.....</b>	<b>5</b>
I、	軟、硬體需求條件.....	5
A.	硬體環境.....	5
B.	軟體環境.....	5
<b>Chapter 4、</b>	<b>系統安裝.....</b>	<b>6</b>
I、	安裝前檢查 .....	6
II、	安裝流程.....	6
III、	環境設定檔說明 .....	8
A.	HiMerchant.conf .....	8
B.	HiServer.conf .....	8
<b>Chapter 5、</b>	<b>交易流程.....</b>	<b>9</b>
<b>Chapter 6、</b>	<b>支援函式.....</b>	<b>10</b>
I、	授權（AUTHORIZATION） .....	11
II、	請款（CAPTURE）、退款（REFUND） .....	15
III、	取消請款（CAPTURE REVERSE）、取消退款（REFUND REVERSE）、 取消授權（AUTHORIZATION REVERSE） .....	16
IV、	訂單查詢（QUERY） .....	19
V、	直接授權（AUTHORIZATIONSSL） .....	21
VI、	直接授權含交易序號標記（AUTHORIZATIONSSLTRXTOKEN） ....	23
VII、	授權含交易序號標記（AUTHORIZATION WITH TRXTOKEN） .....	25
VIII、	定期定額訂單查詢（SIP_QUERY） .....	29
IX、	定期定額訂單取消（SIP_CANCEL） .....	34
X、	隨身付（FOLLOW_PAY） .....	35
XI、	APPLEPAY 加密資料付款（B2CPAYAUTHSSLAPPLEPAY） .....	36
XII、	方便付建立付款連結（B2CEASYPAY CREATE） .....	39
XIII、	方便付訂單查詢（B2CEASYPAYQUERY） .....	41
XIV、	方便付訂單狀態更改（B2CEASYPAYUPDATE） .....	43
XV、	LINE PAY 授權（AUTHORIZATION）.....	44

---

XVI、	LINE PAY 取消授權 ( AUTHORIZATION REVERSE ) .....	47
XVII、	LINE PAY 請款 ( CAPTURE )、退款 ( REFUND ) .....	49
XVIII、	LINE PAY 訂單查詢 ( QUERY ) .....	51
XIX、	LINE PAY 付款按鈕 .....	53
<b>Chapter 7、</b>	<b>參考資料.....</b>	<b>54</b>
I、	回傳訊息對照表 ( RETCODE ) .....	54
II、	訂單狀態碼 .....	55
III、	訂單狀態示意圖 .....	56
IV、	縣市群組代碼 .....	57
V、	RETURNURL、 UPDATE URL 用途與說明 .....	58
<b>Chapter 8、</b>	<b>版本更新說明.....</b>	<b>59</b>

## Chapter 1、簡介

網際網路正巨大的改變著整個世界。也因網路的無時限性、無地域性，人們有著全新生活的體驗。傳統的交易模式藉由網際網路展現著全然不同的面貌，並侵襲著每個不同產業。

網際威信公司在亞洲地區無論是面對商家或是銀行，總是以最新的技術、最熱忱的態度，來提供最好、最完善的服務。

網際威信的線上付款機制，分為 **B2C** 及 **B2B** 二大類，分別支援企業對個人及企業間之電子商務金流服務；其中 **B2C** 的 **Payment** 提供信用卡之安全線上付款服務，並支援下列的付款標準：

- SSL.
- 3D Secure (VBV / SecureCode / Jsecure)

## Chapter 2、金流產品介紹

為使商店能快速連接 HiTRUST 提供之金流服務，網際威信提供二種連接方式，商店可依安全或功能要求選擇任一方式，在技術上能保有最大的彈性，可以讓您輕易整合現有交易平台；此二種方式分別為 **WI** ( Web integration )、**TK** ( Payment Toolkit )。以下表格說明二種方式的特色及差異：

	WI	TK
支援交易種類	所有	所有
技術門檻	低	中
整合靈活性	低	高
安全等級	較低 (交易安全Https)	高 (交易安全性PKI、交易不可否認性Signature)
交易明細	無	有
支援平台或必備SDK	所有的平台	<ul style="list-style-type: none"> <li>■ Windows ASP</li> <li>■ PHP</li> <li>■ 任何支援Java及JSP的作業平台</li> </ul>

本手冊針對 TK (Payment-Toolkit) 之架構、交易流程及支援函式部分加以說明。

---

## Chapter 3、環境需求

### I、軟、硬體需求條件

#### A. 硬體環境

1. 支援 TCP/IP 通信協定網路卡。
2. 至少 200MB 磁碟空間，供 Payment Toolkits 使用。
3. 最少 128MB RAM(建議 256MB)。

#### B. 軟體環境

任何支援 Java 及 JSP 的作業平台。

## Chapter 4、系統安裝

### I、安裝前檢查

在進行安裝作業前，請先檢查您的環境，以確保安裝作業能順利執行。

#### 1. 測試系統 (強烈建議)：

在修改您的正式系統之前，請先建置一套測試的環境，並由該系統連接至網際威信的測試機台，經過測試完成後，再導入正式系統 (修改正式系統前請務必備份)。

### II、安裝流程

#### 1. 安裝：

(1) 您會拿到 toolkit.jar、範例網頁(html、jsp)及一些外部函式(lib.zip)，請將 toolkit.jar 新增至 classpath 中，並依需求將 toolkit 所使用到的外部函式一併新增至 classpath 中。

(2) 完成安裝後聯絡網際威信客服部，索取 StoreID、公私密金鑰及程式之設定檔 (HiServer.conf、HiMerchant.conf、log4j.properties)。

#### 2. 更改設定檔：

(1) 將程式設定檔置於 classpath 下。

(2) 將 HiServer.conf 用文字編輯器開啟：

1. 將 RSAName 更改成公鑰金鑰放置之位置及檔名(如：  
RSAName=C:/PUB.DER)。

2. 完成後儲存。

(3) 將 HiMerchant.conf 用文字編輯器開啟：

1. 將 StoreID 更改成 網際威信所發給的"客戶代號"。

2. 將 RSAName 更改成私密金鑰放置之位置及檔名。

3. 將 ReturnURL 及 UpdateURL 中"yourdomain" 更改成 Merchant Server 對外 IP。

4. 完成後儲存。

(4) 將 HiMerchant.conf 更改檔名為"客戶代號"+".conf"

(5) 修改記錄檔位置：

1. 將 log4j.properties 用文字編輯器開啟。

2. 將內容為"xxxxx"的地方更改成"客戶代號"，您也可以加上路徑(如：C:/log/xxxxx.HiSys)。

(6) 將所有 Jsp 檔案用文字編輯器開啟，將內容為"xxxxx"的地方更改成 網際威信所發給的"客戶代號"。

(7) 確認防火牆針對 Merchant Server 已打開 80 及 443，並允許 TrustPay Server(203.75.242.8)進入。

3. 選擇您所要開發函式：  
所有支援函式均列在第六章。您必須決定您要開發並整合哪些函式至您現有的系統。
4. HTML 網頁寫作 ( 請在測試環境中完成 )：  
您可以根據我們所附的範例網頁來編寫或改寫您的網頁。例如，若您要將授權函式整合至您的系統，您只需找到原系統的最後付款頁面，然後根據我們所要求的欄位及值，送至指定的網址即完成。.
5. 整合性測試：  
連結至測試環境測試線上交易。
6. 上線：  
修改並切換至正式系統，以啟動線上付款機制。



### III、環境設定檔說明

#### A. HiMerchant.conf

項目	使用者 自訂	說明
StoreID	否	商店代碼
RSAName	否	商家私密金鑰檔
SysLog	可	為系統訊息檔
ErrLog	可	為錯誤訊息檔
Deposit	可	<b>0</b> ：一般交易 <b>1</b> ：Sale 交易(授權成功後直接請款)
QueryFlag	可	<b>1</b> ：詳細訂單資料傳回 <b>0</b> ：簡易訂單資料傳回
Currency	否	幣別 (NTD 為台幣)
OrderDesc	可	訂單描述
ReturnURL	可	指定接續網址
UpdateURL	可	加密資料傳回網址
備註:1.若商家於設定檔設定各參數之初始值，則於呼叫函數時可不給定已設定之參數值。 2.部分參數支援網頁上動態設定，若有需要可藉由網頁設定改變其值。		

#### B. HiServer.conf

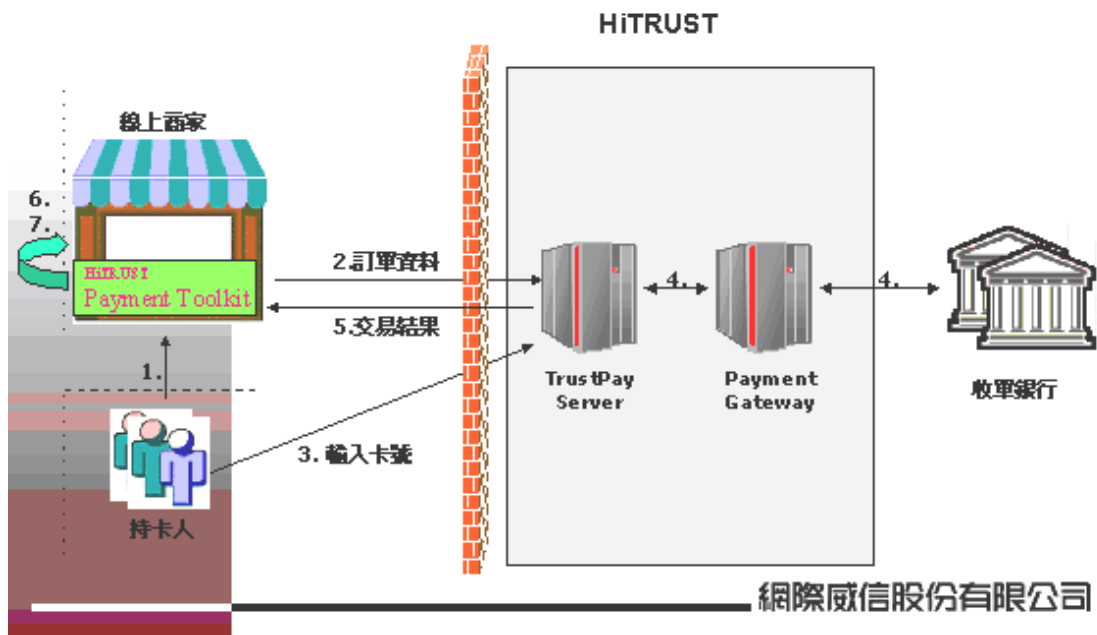
項目	使用者 自訂	說明
IP	否	網際威信 TrustPay Server IP
Port	否	80
RSAName	否	TrustPay Server 私密金鑰檔
AuthUrl	否	授權交易所在的 Servlet 路徑
OtherUrl	否	其他交易所在的 Servlet 路徑
備註：除將 RSAName 更改為公鑰放置位置及檔名外,請勿更改其他設定值。		

## Chapter 5、交易流程

以下針對授權交易作說明：



交易流程圖



1. 持卡人透過商家網站進行線上交易，確認訂單後進行線上付款程序。
2. 商家將該訂單資料，透過 Payment Toolkit 加密後，傳送至 TrustPay Server，並將持卡人導至 TrustPay Server 網頁。
3. 持卡人於 TrustPay Server 輸入信用卡資料(卡號、有效年月)，並視信用卡是否啟動 Visa 3D Secure，來決定是否必須輸入密碼。
4. 當信用卡檢核正確後，TrustPay Server 將交易資料透過 Payment Gateway 送至指定的收單銀行。
5. TrustPay Server 收到由銀行回傳的交易結果，並加密傳送給商家所安裝的 Payment Toolkit ( UpdateURL )。
6. 最後，Payment Toolkit 將根據商家設定的指定接續網址 ( ReturnURL )，將持卡人之網頁重新導向。
7. 請於 merUpdateURL 接收到我司交易結果時，回傳"**R01=00**"給我司系統，以利伺服器判斷是否有正確接收，避免重複發送交易結果。

## Chapter 6、支援函式

下列的交易函式均對應到實體的交易流程，請選擇您所需要的加以開發：

1. 授權 ( Authorization )
2. 取消授權 ( Authorization reverse )
3. 請款 ( Capture )
4. 取消請款 ( Capture reverse )
5. 退款 ( Refund )
6. 取消退款 ( Refund reverse )
7. 訂單查詢 ( Query )
8. 直接授權 ( Authorization SSL )
9. 直接授權含交易序號標記 ( AUTHORIZATIONSSLTRXTOKEN )
10. 授權含交易序號標記 ( AUTHORIZATION WITH TRXTOKEN )
11. 定期定額訂單查詢 ( SIP\_Query )

授權 ( Authorization ) 與直接授權 ( Authorization SSL ) 之差異為：授權 ( Authorization ) 會將消費者導入 TrustPay Server 網頁輸入信用卡資料，而直接授權 ( Authorization SSL ) 為消費者於商家網頁上輸入信用卡資料，商家將訂單資料及信用卡資料經由 Payment ToolKit 系統將資料加密後傳送至 TrustPay Server。

上述所有函式我們均支援，您可以透過 Payment Toolkit 將資料送至網際威信的 TrustPay Server，您透過 Payment Toolkit 所傳送的任何資料，均經過加密處理。

**※注意：(1)所有函式中的交易金額均須包含兩位小數(如交易金額輸入為 100，則為 1.00 元)。**

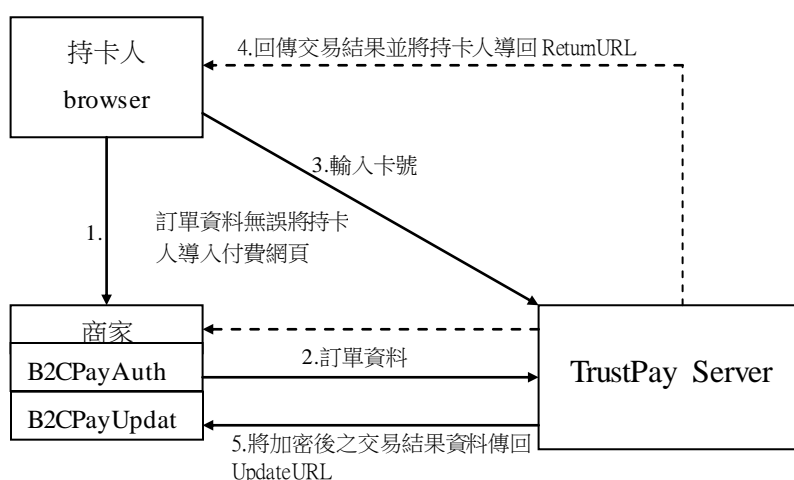
**(2)當授權交易成功之後，您必須完成請款交易，待清算之後，銀行才會撥款 ( 清算指網際威信的 TrustPay Server 與銀行間的結算 )。**

## I、授權（Authorization）

### ● 使用時機

當消費者欲進行付款行為時，商家利用此函式將消費者導入 TrustPay Server 取得銀行的授權。

### ● 交易流程



### ● 函式：B2CPayAuth, B2CPayUpdate

### ● 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100, 則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	
自動請款標記	setDepositFlag(String)	是	有	支援	'1': Sale 交易: 自動請款 '0': 一般交易

查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時, TrustPay Server 會將本筆交易的詳細資料以 POST 的方式送至 <b>UpdateURL</b> '0': 一般資料
指定接續網址	setReturnURL(String)	是	有	支援	
加密結果回傳網址	setUpdateURL(String)	是	有	支援	
機票號碼	setTicketNo(String)				
卡號	setPan(String)				
有效期限	setExpiry(String)				YYMM
CVC2/CVV2	setE01(String)				
分期期數	setE03(String)				
是否使用點點變現金功能	setE04(String)				'0': 不使用 '1': 使用 <b>*點點變現金功能與分期功能無法同時使用</b>
縣市群組代碼	setE11(String)				國旅卡交易使用 必需同時設定縣市群組代碼(見參考資料 IV)、啟程日、回程日。
啟程日	setE12(String)				國旅卡交易使用 mmddyyyy
回程日	setE13(String)				國旅卡交易使用 mmddyyyy
身分證字號	setE14(String)				英文字母大寫
交易序號標記	setE55(String)				'0': 不使用 '1': 使用
扣款期數	setE56(String)				扣款期數
扣款週期	setE57(String)				扣款週期
扣款週期種類	setE58String)				'D': 天 'W': 週 'M': 月 'Y': 年

執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

● 回應(Response)參數

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
與付費網頁溝通標記	getToken()		付費網頁之位置及溝通參數(將使用者導至付費網頁用)

● GET 回傳至 ReturnURL 的參數：

參數名 (請注意大小寫)	必要的	說明
type	是	交易類別
ordernumber	是	訂單號碼
retcode	是	交易結果 ( 詳細說明見第七章 )

● POST 回傳至 UpdateURL 的參數：

參數名 (請注意大小寫)	必要的	說明	
KEY	是	您必須呼叫 B2Cpay Update 來解開資料 (請參照使用範例)	
MAC	是		
CIPHER	是		
解密後資料			
名稱	方法	必要的	說明
交易類別	getType()	是	1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權
交易結果	getRetCode()	是	詳細說明見第七章
訂單號碼	getOrderNo()	是	
銀行授權碼	getAuthCode()	是	
銀行調單編號	getAuthRRN()	是	

交易序號標記	getTrxToken		當您啟動 E55(交易序號標記) 才會回傳此欄位
有效期限	getExpiry()		
遮罩後卡號	getPan()		取得遮罩卡號前六後四碼，需先向網際威信申請開啟回傳設定。
.			當您啟動 QueryFlag，其他欄位才會出現( 請參考訂單查詢函式 )
..			
...			
-			

- 使用範例

#### B2CPayAuth :

```

B2CPayAuth auth = new B2CPayAuth();
auth.setStoreId("XXXXX");
auth.setOrderNo((String)request.getParameter("ordernumber"));
auth.setOrderDesc((String)request.getParameter("orderdesc"));
auth.setAmount((String)request.getParameter("amount"));
auth.setTicketNo((String)request.getParameter("ticketno"));
auth.setQueryFlag("1");
auth.transaction();

```

#### B2CpayUpdate :

```

B2CPayUpdate payUpdate = new B2CPayUpdate();
payUpdate.setStoreId("XXXXX");
payUpdate.setKey((String)request.getParameter("KEY"));
payUpdate.setMac((String)request.getParameter("MAC"));
payUpdate.setCipher((String)request.getParameter("CIPHER"));
payUpdate.transaction();

```

- 範例網頁

```

B2C_Auth.html
B2C_Auth.jsp
B2C_Update.jsp

```

## II、請款 (Capture)、退款 (Refund)

- 使用時機

- (1) 商家欲對訂單向銀行申請貨款時。
- (2) 商家欲對請款已清算之訂單進行退回貨款時。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
交易類別	setType(String)	是			1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100,則代表 1 元
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時，TrustPay Server 將會回傳本筆交易的詳細資料 '0': 一般資料
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					



- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
· .. ... -			當您啟動 QueryFlag，其他欄位才會出現( 請參考訂單查詢函式 )

- 使用範例

```

B2CPayOther trx = new B2CPayOther();
trx.setStoreId("XXXXX");
trx.setType(trx.CAPTURE);
trx.setOrderNo((String)request.getParameter("ordernumber"));
trx.setAmount((String)request.getParameter("amount"));
trx.setQueryFlag((String)request.getParameter("queryflag"));
trx.transaction();

```

- 範例網頁

```

B2C_Capture.html
B2C_Capture.jsp
B2C_Refund.html
B2C_Refund.jsp

```

### III、 取消請款（ Capture Reverse ）、取消退款（ Refund Reverse ）、取消授權（ Authorization Reverse ）

- 使用時機

- (1) 商家針對已請款訂單(訂單需未清算)進行取消請款。
- (2) 當商家欲對退回貨款做取消時(訂單需未清算)。
- (3) 商家針對未請款訂單進行取消授權。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
交易類別	setType(String)	是			1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時， TrustPay Server 將會 回傳本筆交易的詳細 資料 '0': 一般資料
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
.			當您啟動 QueryFlag，其他欄位 才會出現( 請參考訂單查詢函式 )
..			
...			
-			

- 使用範例

```
B2CPayOther trx = new B2CPayOther();
```

```
trx.setStoreId("XXXXXX");  
trx.setType(trx.RE_CAPT);  
trx.setOrderNo((String)request.getParameter("ordernumber"));  
trx.setQueryFlag((String)request.getParameter("queryflag"));  
trx.transaction();
```

- 範例網頁

B2C\_CaptureRe.html  
B2C\_CaptureRe.jsp  
B2C\_RefundRe.html  
B2C\_RefundRe.jsp  
B2C\_AuthRe.html  
B2C\_AuthRe.jsp

## IV、 訂單查詢 (Query)

### ● 使用時機

商家可利用本函式來查詢訂單詳細資料，以確定訂單狀態或交易是否完成。

### ● 函式：B2CPayOther

### ● 請求(Request)參數

名稱	方法	必要的	說明
交易類別	setType(String)	是	1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是	長度限制 19 碼
執行交易	transaction()		

### ● 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
訂單狀態碼	getOrderStatus()	是	詳細說明見第七章
授權方式	getEci()	是	SSL, VISA3D
訂單日期	getOrderDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX
核准金額	getApproveAmount()		
銀行授權碼	getAuthCode()		
銀行調單編號	getAuthRRN()		
請款金額	getCaptureAmount()		
請款批次號碼	getPayBatchNum()		
請款日期	getCaptureDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX
退款金額	getRefundAmount()		
退款批次號碼	getRefundBatch()		
退款調單編號	getRefundRRN()		
退款授權碼	getRefundCode()		

退款日期	getRefundDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX
點點變現金銷帳編號	getRedemordernum()		
本次折抵點數	getRedem_discount_point()		
本次折抵金額	getRedem_discount_amount()		
本次實付金額	getRedem_purchase_amount()		
剩餘點數	getRedem_balance_point()		
遮罩後卡號	getPan()		取得遮罩卡號前六後四碼，需先向網際威信申請開啟回傳設定。
分期期數	getE06()		
首期金額	getE07()		
每期金額	getE08()		

- 使用範例

```

B2CPayOther trx = new B2CPayOther();
trx.setStoreId("XXXXX");
trx.setType(trx.QUERY);
trx.setOrderNo((String)request.getParameter("ordernumber"));
trx.transaction();

```

- 範例網頁

```

B2C_Query.html
B2C_Query.jsp

```

## V、直接授權（AuthorizationSSL）

### ● 使用時機

當消費者欲進行付款行為時，直接於商家網頁上輸入信用卡資料，商家利用此函式將交易資料及信用卡資料加密透過 TrustPay Server 取得銀行的授權。

### ● 函式：B2CPayAuthSSL

### ● 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100,則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	
自動請款標記	setDepositFlag(String)	是	有	支援	'1': Sale 交易: 自動請款 '0': 一般交易
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時， TrustPay Server 將會 回傳本筆交易的詳細 資料 '0': 一般資料
機票號碼	setTicketNo(String)				
卡號	setPan(String)	是			
有效期限	setExpiry(String)	是			YYMM
CVC2/CVV2	setE01(String)				
分期期數	setE03(String)				
是否使用點點變 現金功能	setE04(String)				'0': 不使用 '1': 使用 <b>*點點變現金功能與分期功 能無法同時使用</b>

縣市群組代碼	setE11(String)				國旅卡交易時，必需同時設定縣市群組代碼(見參考資料 IV)、啟程日、回程日。
啟程日	setE12(String)				國旅卡交易使用， mmddyyyy
回程日	setE13(String)				國旅卡交易使用， mmddyyyy
身分證字號	setE14(String)				英文字母大寫
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

● 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
.			當您啟動 QueryFlag，其他欄位才會出現(請參考訂單查詢函式)
..			
...			
-			

● 使用範例

```

B2CPayAuthSSL auth = new B2CPayAuthSSL();
auth.setStoreId("XXXXX");
auth.setOrderNo((String)request.getParameter("ordernumber"));
auth.setOrderDesc((String)request.getParameter("orderdesc"));
auth.setAmount((String)request.getParameter("amount"));
auth.setTicketNo((String)request.getParameter("ticketno"));
auth.setPan((String)request.getParameter("pan"));
auth.setExpiry((String)request.getParameter("expiry"));
auth.setE04("1");
auth.transaction();

```

● 範例網頁

```

B2C_AuthPAN.html
B2C_AuthPAN.jsp

```

## VI、直接授權含交易序號標記 ( AuthorizationSSLTrxToken )

### ● 使用時機

當消費者欲進行付款行為時，商家利用此函式將交易資料及留存在商家端的交易序號標記等資訊透過 TrustPay Server，取得銀行的授權。

### ● 函式：B2CPayAuthSSLTrxToken

### ● 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100,則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	
自動請款標記	setDepositFlag(String)	是	有	支援	'1': Sale 交易: 自動請款 '0': 一般交易
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時，TrustPay Server 將會回傳本筆交易的詳細資料 '0': 一般資料
交易序號標記	setTrxToken(String)	是			速速付交易序號
有效期限	setExpiry(String)	是			YYMM
分期期數	setE03(String)				
是否使用點點變 現金功能	setE04(String)				'0': 不使用 '1': 使用 <b>*點點變現金功能與分期功能無法同時使用</b>
縣市群組代碼	setE11(String)				國旅卡交易時，必需同時設定縣市群組代碼(見參考資料 IV)、啟程日、回程日。
啟程日	setE12(String)				國旅卡交易使用， mmddyyyy



回程日	setE13(String)				國旅卡交易使用， mmddyyyy
身分證字號	setE14(String)				英文字母大寫
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

● 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
. .. ... -			當您啟動 QueryFlag，其他欄位 才會出現( 請參考訂單查詢函式 )

● 使用範例

```

B2CPayAuthSSLTrxToken auth = new B2CPayAuthSSLTrxToken();
auth.setStoreId("XXXXX");
auth.setOrderNo((String)request.getParameter("ordernumber"));
auth.setOrderDesc((String)request.getParameter("orderdesc"));
auth.setAmount((String)request.getParameter("amount"));
auth.setTrxToken((String)request.getParameter("TrxToken"));
auth.setExpiry((String)request.getParameter("Expiry"));
auth.transaction();

```

● 範例網頁

```

B2C_AuthSSL_TrxToken.html
B2C_AuthSSL_TrxToken.jsp

```

## VII、 授權含交易序號標記 ( Authorization with trxToken )

### ● 使用時機

當消費者欲進行付款行為時，商家平台直接帶入交易序號，利用此函式將交易資料加密送至 TrustPay Server；TrustPay Server 將交易序號對應出信用卡號後進行授權作業，商家平台至 TrustPay Server 不會傳遞卡號資料。

### ● 函式：B2CPayAuthTrxToken

此函示支援 3DS 驗證

### ● 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100, 則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	
自動請款標記	setDepositFlag(String)	是	有	支援	'1': Sale 交易: 自動請款 '0': 一般交易
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時， TrustPay Server 將會 回傳本筆交易的詳細 資料 '0': 一般資料
指定接續網址	setReturnURL(String)	是	有	支援	
加密結果回傳網址	setUpdateURL(String)	是	有	支援	
機票號碼	setTicketNo(String)				
卡號	setPan(String)				
有效期限	setExpiry(String)				YYMM
交易序號	setTrxToken(String)	是			
CVC2/CVV2	setE01(String)				
分期期數	setE03(String)				

是否使用點點變現金功能	setE04(String)				'0':不使用 '1':使用 <b>*點點變現金功能與分期功能無法同時使用</b>
縣市群組代碼	setE11(String)				國旅卡交易時，必需同時設定縣市群組代碼(見參考資料 IV)、啟程日、回程日。
啟程日	setE12(String)				國旅卡交易使用， mmddyyyy
回程日	setE13(String)				國旅卡交易使用， mmddyyyy
身分證字號	setE14(String)				英文字母大寫
交易序號標記	setE55(String)				'0':不使用 '1':使用
<b>執行交易</b>	<b>transaction()</b>				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

● 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
與付費網頁溝通標記	getToken()		付費網頁之位置及溝通參數(將使用者導至付費網頁用)

● 回傳至 returnUrl 的參數：

參數名 (請注意大小寫)	必要的	說明
type	是	交易類別
ordernumber	是	訂單號碼
retcode	是	交易結果 ( 詳細說明見第七章 )

● 回傳至 UpdateURL 的參數：

參數名 (請注意大小寫)		必要的	說明
KEY		是	您必須呼叫 B2Cpay Update 來解開資料 (請參照使用範例)
MAC		是	
CIPHER		是	
解密後資料			
名稱	方法	必要的	說明
交易類別	getType()	是	1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權
交易結果	getRetCode()	是	詳細說明見第七章
訂單號碼	getOrderNo()	是	
銀行授權碼	getAuthCode()	是	
銀行調單編號	getAuthRRN()	是	
交易序號標記	getTrxToken		當您啟動 E55(交易序號標記) 才會回傳此欄位
有效期限	getExpiry()		
遮罩後卡號	getPan()		取得遮罩卡號前六後四碼，需先 向網際威信申請開啟回傳設定。
.			當您啟動 QueryFlag，其他欄位 才會出現( 請參考訂單查詢函式 )
..			
...			
-			

- 使用範例

**B2CPayAuthTrxToken :**

```
B2CPayAuthTrxToken auth = new B2CPayAuthTrxToken ();  
auth.setStoreId("XXXXX");  
auth.setOrderNo((String)request.getParameter("ordernumber"));  
auth.setOrderDesc((String)request.getParameter("orderdesc"));  
auth.setAmount((String)request.getParameter("amount"));  
auth.setTrxToken((String) request.getParameter("trxToken"));  
auth.setQueryFlag("1");  
auth.transaction();
```

- 範例網頁

B2C\_Auth\_TrxToken.html

B2C\_Auth\_TrxToken.jsp

## VIII、定期定額訂單查詢 (SIP\_Query)

- 使用時機

商家可利用本函式來查詢定期定額首期及各期訂單資料，以確定訂單狀態或交易是否完成。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	說明
交易類別	setType(String)	是	1:授權 3:請款 4:取消請款 5:退款 6:取消退款 7:查詢 8:取消授權 9:定期定額查詢
訂單號碼	setOrderNo(String)	是	長度限制 19 碼
執行交易	transaction()		

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
訂單資訊	getSIPData()	是	回傳(JSON Object)：SIPData

● **JSON Object : SIPData 說明**

SIPInfo			
名稱	參數名稱	必要的	說明
訂單建立日期	createTime	是	YYYY-MM-DD.HH.MM.SS
週期種類	periodType	是	- Y 年 - M 月 - D 日
扣款頻率	deductFreq	是	
扣款總期數	deductTotalNum	是	
已扣款期數	deductChargedNum	是	
遮罩後卡號	maskPan	是	
效期	expire	是	

SIPDetail			
名稱	參數名稱	必要的	說明
期數	periodNumber	是	
訂單編號	orderNumber	是	
扣款狀態碼	status	是	- 00 已扣款 - 01 待扣款 - 02 扣款失敗 - 03 已終止定期定額
訂單狀態碼	orderStatus		詳細說明見第七章
預計扣款日期	estimatedDeductDate	是	YYYY-MM-DD

- **SIPData** 回傳資訊範例：

**回傳資訊範例 1.首期訂單已扣款、產生次期待扣款訂單**

```
{
  "SIPInfo":
  {
    "createTime":"2017-07-24-19.19.18",
    "periodType":"D",
    "deductFreq":"1",
    "deductTotalNum":"5",
    "deductChargedNum":"1",
    "maskPan":"444433*****1111",
    "expire":"1912"
  }
,
  "SIPDetail":[
    {
      "periodNumber":"01",
      "orderNumber":"ORDER_01",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-24"
    },
    {
      "periodNumber":"02",
      "orderNumber":"AUTO_CREATE_ORDER_01",
      "status":"01",
      "orderStatus":"",
      "estimatedDeductDate":"2017-07-25"
    }
  ]
}
```



回傳資訊範例 2.首期至第二期訂單 **已扣款**、第三期因超額 **扣款失敗**、第四期 **已扣款**

```
{
  "SIPInfo":
  {
    "createTime":"2017-07-24-19.19.18",
    "periodType":"D",
    "deductFreq":"1",
    "deductTotalNum":"5",
    "deductChargedNum":"4",
    "maskPan":"444433*****1111",
    "expire":"1912"
  }
,
  "SIPDetail":[
    {
      "periodNumber":"01",
      "orderNumber":"ORDER_01",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-24"
    },
    {
      "periodNumber":"02",
      "orderNumber":"AUTO_CREATE_ORDER_01",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-25"
    },
    {
      "periodNumber":"03",
      "orderNumber":"AUTO_CREATE_ORDER_02",
      "status":"02",
      "orderStatus":"",
      "estimatedDeductDate":"2017-07-26"
    },
    {
      "periodNumber":"04",
      "orderNumber":"AUTO_CREATE_ORDER_03",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-27"
    },
    {
      "periodNumber":"05",
      "orderNumber":"AUTO_CREATE_ORDER_04",
      "status":"01",
      "orderStatus":"",
      "estimatedDeductDate":"2017-07-28"
    }
  ]
}
```

回傳資訊範例 3. 首期至第二期訂單 **已扣款**、第三期因超額 **扣款失敗**，  
第四期 **已扣款**、第五期 **終止定期定額**

```
{
  "SIPInfo":
  {
    "createTime":"2017-07-24-19.19.18",
    "periodType":"D",
    "deductFreq":"1",
    "deductTotalNum":"5",
    "deductChargedNum":"4",
    "maskPan":"444433*****1111",
    "expire":"1912"
  }
},
  "SIPDetail":[
    {
      "periodNumber":"01",
      "orderNumber":"ORDER_01",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-24"
    },
    {
      "periodNumber":"02",
      "orderNumber":"AUTO_CREATE_ORDER_01",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-25"
    },
    {
      "periodNumber":"03",
      "orderNumber":"AUTO_CREATE_ORDER_02",
      "status":"02",
      "orderStatus":"",
      "estimatedDeductDate":"2017-07-26"
    },
    {
      "periodNumber":"04",
      "orderNumber":"AUTO_CREATE_ORDER_03",
      "status":"00",
      "orderStatus":"04",
      "estimatedDeductDate":"2017-07-27"
    },
    {
      "periodNumber":"05",
      "orderNumber":"AUTO_CREATE_ORDER_04",
      "status":"03",
      "orderStatus":"",
      "estimatedDeductDate":"2017-07-28"
    }
  ]
}
```

- 使用範例

```
B2CPayOther trx = new B2CPayOther();
trx.setStoreId("XXXXX");
trx.setType(trx.SIP_QUERY);
trx.setOrderNo((String)request.getParameter("ordernumber"));
trx.transaction();
```

- 範例網頁

```
B2C_SIP_Query.html
B2C_SIP_Query.jsp
```

## IX、定期定額訂單取消 (SIP\_CANCEL)

- 使用時機

商家可利用本函式來取消定期定額的訂單。

- 函式：B2CPayOther

- 請求 ( Request ) 參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
交易類別	setType(String)	是			33:取消定期定額
訂單號碼	setOrderNo(String)	是			主訂單，長度限制 19 碼
執行交易	transaction()				
備註：取消後就不會再固定時間扣款，且取消定期定額訂單必須為主訂單。					

- 回應 ( Response ) 參數

名稱	方法	必要的	說明
交易結果	getRetCode()	是	

- 使用範例

```
B2CPayOther trx = new B2CPayOther();  
trx.setType("33");  
trx.setOrderNo((String)request.getParameter("ordernumber"));  
trx.transaction();
```

## X、 隨身付 (FOLLOW\_PAY)

- 使用時機

當消費者欲進行付款行為時，商家利用此函式取得銀行的授權。

- 使用範例說明

1. 頁面需引用 js 資源
  - 1.1. <https://trustpay.hitrust.com.tw/TRUSTPAY/js/hitrust-fp.js>
  - 1.2. <https://code.jquery.com/jquery-1.10.2.js>
2. 您需定義 iframe 名稱 (hitrust\_iframe)讓調用的 js 自動嵌入，並定義付款按鈕名稱(btn-hitrustpay) 註：欄位設定名稱請參考以下範例  
頁面範例程式碼：

```
<html>  
<head>  
<!--引用以下JS檔案-->  
<script src="https://code.jquery.com/jquery-1.10.2.js"> </script>  
<script  
src="https://trustpay.hitrust.com.tw/TRUSTPAY/js/hitrust-fp.js"> </script>  
<title>Sample</title>  
<script>  
    <!--確保資源都有完整載入-->  
    window.onload = function() {  
        <!--設定交易相關資訊-->  
        var fObj = {};  
        fObj['Type'] = "FOLLOW_PAY";  
        fObj['storeid'] = "88046";  
        fObj['ordernumber'] = "test123456";  
        fObj['amount'] = "100";  
        fObj['orderdesc'] = "follow pay sample";  
        fObj['merUpdateURL'] = "https://merupdateurl.com";  
        fObj['returnURL'] = "https://returnurl.com";
```

```

        fPObj['e03'] = "0";
        <!--執行doFLP()系統會寫入iframe的src-->
        doFLP(fPObj);

    };
</script>
</head>

<body>
    <!--可自定義iframe格式-->
    <iframe id="hitrust_iframe" name="hitrust_iframe" frameborder="0"
        scrolling="no" ></iframe>
    <!--設定followPayToken input和btn-hitrustpay-->
    <input type="hidden" id="followPayToken">
    <button type="button" id="btn-hitrustpay"
disabled="none"> <p>Pay</p> </button>
</body>

</html>

```

## XI、 ApplePay 加密資料付款（B2CPayAuthSSLApplePay）

### ● 使用時機

商家自行與 Apple 進行串接後，取得加密資訊後，商家無須自行解密，即可透過此函式進行 HiTRUST Apple Pay 付款作業。

### ● 函式：B2CPayAuthSSLApplePay

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
商家編號	setStoreId(String)	是			
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex.若輸入值為 100,則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	

自動請款標記	setDepositFlag(String)	是	有	支援	'1' : Sale 交易: 自動請款 '0' : 一般交易 <b>*商家欲使用自動請款交易時,需先經過銀行同意。</b>
查詢標記	setQueryFlag(String)	是	有	支援	'1' : 詳細資料: 當您啟動本參數時, TrustPay Server 會將本筆交易的詳細資料以 POST 的方式送至 UpdateURL '0' : 一般資料

啟用 ApplePay 功能	setE60(String)	是			'0' : 不啟用 '1' : 啟用 Applepay '2' : 啟用 pay with google
ApplePayData	setE61(String)	是			填入 Apple/google 回傳的 data json 資料。
憑證註記	setE66(String)				'Y' : 使用 Hitrust 憑證
執行交易	transaction()				
備註: 支援動態設定之參數, 可給定新的值取代設定檔之預設值, 若不另行給定值則送出預設值。					

### ● 回應(Response)參數

名稱	方法	必要的	說明
交易結果	getRetCode()	是	

### ● 程式範例：

```
B2CPayAuthSSLApplePay auth = new B2CPayAuthSSLApplePay();
auth.setStoreId = "XXXXX";
auth.setOrderNo = Request.Params.Get("ordernumber");
auth.setOrderDesc = Request.Params.Get("orderdesc");
auth.setAmount = Request.Params.Get("amount");
```

```
//E60 : 為 1 表示 Apple pay 為 2 表示 google pay
auth.setE60 = Request.Params.Get("E60");
```

```
auth.setE61 = Request.Params.Get("E61");  
auth.transaction();
```

## XII、 方便付建立付款連結 (B2CEasyPay Create)

- 使用時機

商家可利用本函式來取得付款的連結，供消費者直接點擊連結進行付款動作。

- 函式：B2CEasyPay Create

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
付款人 Email	setEmailAddress(String)	是			多組帳號請用「,」分開
收款名稱	setOrderName(String)	是			
金額	setAmount (String)	是			須包含兩位小數 Ex. 若輸入值為 100,則代表 1 元
自動請款標記	setDepositFlag(String)	是	有		'1': Sale 交易: 自動請款 '0': 一般交易
有效日期起迄	setAvailableDate(Date, Date)	是			格式: yyyy/MM/dd
可分期期數	setInstallmentPlans(String)	是			請用「,」分開 EX: 可使用 3 期及 6 期, 則為 0,3,6
付款完成通知 Email	setNoticeMail(String)	是			
付款完成通知 UpdateURL	setUpdateURL(String)				
訂單描述	setOrderDesc(String)				

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
付款連結	getEasyPayLink()		付款連結須加上 token 值，才會是正確的付款連結網址。
付款清單	getEasyPayDetailList()		以下明細資訊請透過此清單取出
商家代碼	getStoreId()		



收款名稱	getOrderName()		
付款起始日	getStartDate()		
付款截止日	getEndDate()		
付款建立日期	getCreateDate()		
付款金額	getAmount()		包含兩位小數 Ex. 若輸入值為100,則代表 1 元
付款人 Email	getMail()		
訂單序號	getOrderId()		
付款狀態	getStatus()		0:未付款 1:付款中 2:已付款 5:暫停付款
Token	getToken()		
付款日期	getPayDate()		
可分期期數	getInstallm_plan()		
付款完成通知 Email	getNotice_mail()		
付款類型	getPaytype()		

### ● 使用範例

```

B2CEasyPayCreator creator = new B2CEasyPayCreator(storeId);
creator.setEmailAddress(email);
creator.setOrderName(orderName);
creator.setAmount(amount);
creator.setInstallmentPlans(installmentPlans);
creator.setUpdateURL(updateURL);
creator.setNoticeMail(noticeMail);
creator.setAvailableDate(sdf.parse(startDate),sdf.parse(endDate));
creator.setDepositFlag(depositflag);
creator.setOrderDesc(orderdesc);
creator.transaction();

```

### ● 範例網頁

B2C\_EasyPayCreate.html  
B2C\_EasyPayCreate.jsp

### XIII、 方便付訂單查詢 (B2CEasyPayQuery)

- 使用時機

可利用本函式查詢方便付訂單資訊。

- 函式：B2CEasyPay Query

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
付款人 Email	setEmail (String)				只能查詢一組 Email
有效日期起迄	setAvailableDate(Date, Date)				格式：yyyy/MM/dd
建立日期起迄	setCreateDate(Date, Date)				格式：yyyy/MM/dd

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
取得訂單列表	getOrders()	是	以下明細資訊請透過此清單取出
商家代碼	getStoreId()		
收款名稱	getOrderName()		
付款起始日	getStartDate()		
付款截止日	getEndDate()		
付款建立日期	getCreateDate()		
付款金額	getAmount()		包含兩位小數 Ex. 若輸入值為100,則代表 1 元
付款人 Email	getMail()		
訂單序號	getOrderId()		
付款狀態	getStatus()		0:未付款 1:付款中 2:已付款 5:暫停付款
Token	getToken()		
付款日期	getPayDate()		
可分期期數	getInstallm_plan()		

付款完成通知 Email	getNotice_mail()		
付款類型	getPaytype()		

- 使用範例

```
B2CEasyPayQuery qry = new B2CEasyPayQuery("XXXXX");
qry.setCreateDate((sdf.parse(createStartDate+" 00:00:00")),
(sdf.parse(createEndDate+" 23:59:59")));
qry.setAvailableDate(
(sdf.parse(startDate+" 00:00:00")),
(sdf.parse(endDate+" 23:59:59")));
qry.setEmail(email);
qry.transaction();
```

- 範例網頁

```
B2C_EasyPayQuery.html
B2C_EasyPayQuery.jsp
```

## XIV、 方便付訂單狀態更改 (B2CEasyPayUpdate)

- 使用時機

可利用本函式更改方便付訂單狀態。

- 函式：B2CEasyPayUpdate

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單序號	setOrderId(String)	是			
Token	setToken(String)	是			
付款人 Email	setEmail(String)	是			
修正的狀態值	changeStatusTo(int)	是			0:未付款 1:付款中 2:已付款 5:暫停付款

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章

- 使用範例

```
B2CEasyPayUpdate update = new B2CEasyPayUpdate("XXXXX");
update.setOrderId(orderId);
update.setToken(token);
update.setEmail(email);
update.changeStatusTo(Integer.parseInt(toStatus));
update.transaction();
```

- 範例網頁

```
B2C_EasyPayUpdate.html
B2C_EasyPayUpdate.jsp
```

## XV、 LINE Pay 授權 (Authorization)

### ● 使用時機

當消費者欲進行付款行為時，商家利用此函式將消費者導入 LINE Pay 付款頁面。

### ● 函式：B2CPayAuth

### ● 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex.若輸入值為 100,則代表 1 元
訂單說明	setOrderDesc(String)	是	有	支援	
自動請款標記	setDepositFlag(String)	是	有	支援	'1': Sale 交易: 自動請款 ※如與 LINEPAY 申請自動請款模式，此欄位請帶入"1" '0': 一般交易: 手動請款 ※如與 LINEPAY 申請手動請款模式，此欄位請帶入"0"，特別注意五天內未請款該筆交易會被取消。
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時，TrustPay Server 會將本筆交易的詳細資料以 POST 的方式送至 UpdateURL '0': 一般資料
指定接續網址	setReturnURL(String)	是	有	支援	
加密結果回傳網址	setUpdateURL(String)	是	有	支援	
執行交易	transaction()				

備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。

### ● 回應(Response)參數

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
與付費網頁溝通標記	getToken()		付費網頁之位置及溝通參數(將使用者導至付費網頁用)

### ● GET 回傳至 ReturnURL 的參數：

參數名 (請注意大小寫)	必要的	說明
type	是	交易類別
ordernumber	是	訂單號碼
retcode	是	交易結果 ( 詳細說明見第七章 )

### ● POST 回傳至 UpdateURL 的參數：

參數名 (請注意大小寫)	必要的	說明	
KEY	是	您必須呼叫 B2CpayUpdate 來解開資料 (請參照使用範例)	
MAC	是		
CIPHER	是		
解密後資料			
名稱	方法	必要的	說明
交易類別	getType()	是	1:授權 3:請款 5:退款 7:查詢 8:取消授權
交易結果	getRetCode()	是	詳細說明見第七章
訂單號碼	getOrderNo()	是	
. .. ... -			當您啟動 QueryFlag，其他欄位 才會出現( 請參考訂單查詢函式 )

- 使用範例

**B2CPayAuth :**

```
B2CPayAuth auth = new B2CPayAuth();  
auth.setStoreId("XXXXX");  
auth.setOrderNo((String)request.getParameter("ordernumber"));  
auth.setOrderDesc((String)request.getParameter("orderdesc"));  
auth.setAmount((String)request.getParameter("amount"));  
auth.setQueryFlag("1");  
auth.transaction();
```

**B2CpayUpdate :**

```
B2CPayUpdate payUpdate = new B2CPayUpdate();  
payUpdate.setStoreId("XXXXX");  
payUpdate.setKey((String)request.getParameter("KEY"));  
payUpdate.setMac((String)request.getParameter("MAC"));  
payUpdate.setCipher((String)request.getParameter("CIPHER"));  
payUpdate.transaction();
```

- 範例網頁

```
B2C_Auth.html  
B2C_Auth.jsp  
B2C_Update.jsp
```

## XVI、 LINE Pay 取消授權 (Authorization Reverse)

- 使用時機

(1) 商家針對未請款訂單進行取消授權。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
交易類別	setType(String)	是			1:授權 3:請款 5:退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時， TrustPay Server 將會 回傳本筆交易的詳細 資料 '0': 一般資料
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
.			當您啟動 QueryFlag，其他欄位 才會出現(請參考訂單查詢函式)
..			
...			
-			



- 使用範例

```
B2CPayOther trx = new B2CPayOther();  
trx.setStoreId("XXXXXX");  
trx.setType(trx.RE_AUTH);  
trx.setOrderNo(request.getParameter("ordernumber"));  
trx.setQueryFlag(request.getParameter("queryflag"));  
trx.transaction();
```

- 範例網頁

B2C\_AuthRe.html  
B2C\_AuthRe.jsp

## XVII、 LINE Pay 請款 (Capture)、退款 (Refund)

- 使用時機

- (1) 商家欲對已授權訂單向 LINE Pay 申請請款時。
- (2) 商家欲對已請款訂單進行退回款項時。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	預設值	動態設定 (網頁設定)	說明
交易類別	setType(String)	是			1:授權 3:請款 5:退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是			長度限制 19 碼
交易金額	setAmount(String)	是			須包含兩位小數 Ex. 若輸入值為 100,則代表 1 元
查詢標記	setQueryFlag(String)	是	有	支援	'1': 詳細資料: 當您啟動本參數時， TrustPay Server 將會 回傳本筆交易的詳細 資料 '0': 一般資料
執行交易	transaction()				
備註：支援動態設定之參數，可給定新的值取代設定檔之預設值，若不另行給定值則送出預設值。					

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
· .. ... -			當您啟動 QueryFlag，其他欄位 才會出現( 請參考訂單查詢函式 )

- 使用範例

```
B2CPayOther trx = new B2CPayOther();  
trx.setStoreId("XXXXX");  
trx.setType(trx.CAPTURE);  
trx.setOrderNo(request.getParameter("ordernumber"));  
trx.setAmount(request.getParameter("amount"));  
trx.setQueryFlag(request.getParameter("queryflag"));  
trx.transaction();
```

- 範例網頁

```
B2C_Capture.html  
B2C_Capture.jsp  
B2C_Refund.html  
B2C_Refund.jsp
```

## XVIII、 LINE Pay 訂單查詢 (Query)

- 使用時機

商家可利用本函式來查詢訂單詳細資料，以確定訂單狀態或交易是否完成。

- 函式：B2CPayOther

- 請求(Request)參數

名稱	方法	必要的	說明
交易類別	setType(String)	是	1:授權 3:請款 5:退款 7:查詢 8:取消授權
訂單號碼	setOrderNo(String)	是	長度限制 19 碼
執行交易	transaction()		

- 回應(Response)參數：

名稱	方法	必要的	說明
交易結果	getRetCode()	是	詳細說明見第七章
訂單狀態碼	getOrderStatus()	是	詳細說明見第七章
訂單日期	getOrderDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX
核准金額	getApproveAmount()		
請款金額	getCaptureAmount()		
請款日期	getCaptureDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX
退款金額	getRefundAmount()		
退款日期	getRefundDate()		YYYY-MM-DD.HH.MM.SS.XXXXXX

- 使用範例

```
B2CPayOther trx = new B2CPayOther();  
trx.setStoreId("XXXXX");  
trx.setType(trx.QUERY);  
trx.setOrderNo(request.getParameter("ordernumber"));  
trx.transaction();
```

- 範例網頁

```
B2C_Query.html  
B2C_Query.jsp
```

## XIX、 LINE Pay 付款按鈕

- 下載連結

<https://drive.google.com/file/d/0B8KC8-cHKIMGNVd5cHBnQWZoQUU/view>

### LINE Pay

mix payment select button for online

2018.05 / ver02

TYPE-A

● LINE Pay × P (可用LINE Points折抵) 18PX Minimum size

TYPE-B

● LINE Pay × P  
(可用LINE Points折抵) 36PX Minimum size

## Chapter 7、參考資料

### I、回傳訊息對照表 ( retcode )

回傳訊息對照表已上傳至敝司雲端，再請下載，網址如下：

<https://docs.google.com/spreadsheets/d/180yyxtbDUgINVwcgxnWK66GslerNp9MPkGk5WfT4FgY/edit#gid=0>

此為敝公司內部整理的代碼，僅供參考使用。

詳細代碼原因，請依銀行端為主。

附件資料共分兩種代碼，正向及負向代碼。

正向代碼是由銀行端回復的訊息，

而負向代碼則是網際威信端系統回覆的訊息。

## II、 訂單狀態碼

- 訂單狀態總表

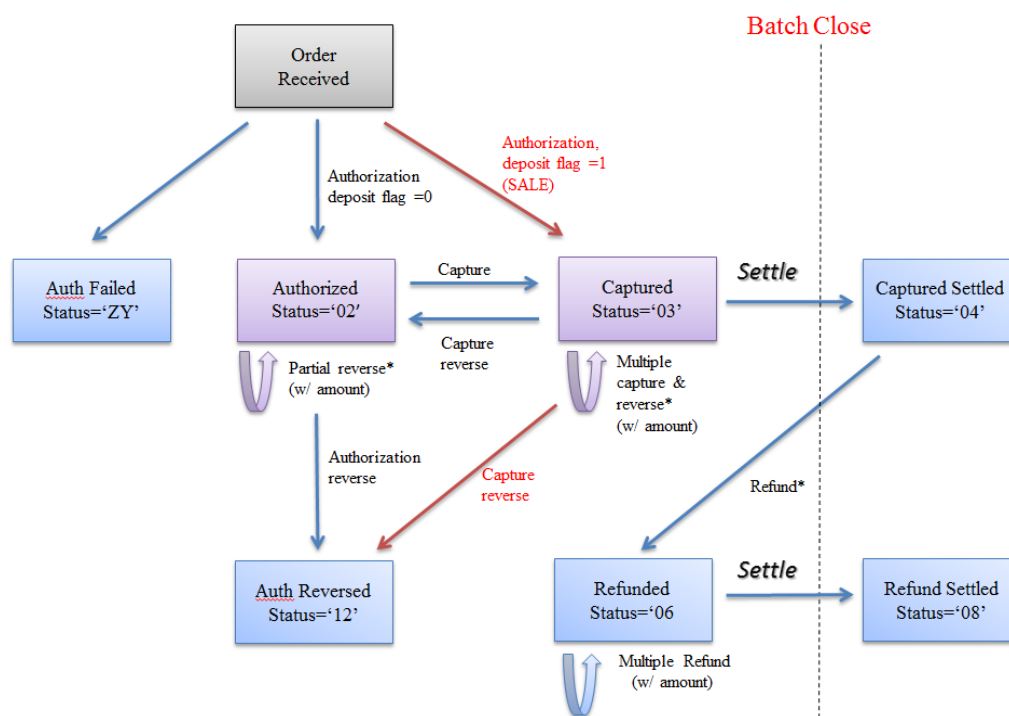
Code	說明	可執行函式
02	已授權	AuthRe
03	已請款	CaptureRe
04	請款已清算	Refund
06	已退款	RefundRe
08	退款已清算	X
12	訂單已取消	X
ZY	授權失敗	X



### III、 訂單狀態示意圖

下圖說明 Chapter 6 中各函式產生訂單狀態變化

**Status Transition Diagram**



注意：

\*各項函式提供與否，將視各收單行之業務支援而定，請洽詢銀行或網際威信客服人員。

#### IV、縣市群組代碼

縣市名稱	縣市群組代碼
台北市	001
基隆市	002
台北縣	003
宜蘭縣	004
新竹市	005
新竹縣	006
桃園縣	007
苗栗縣	008
台中市	009
台中縣	010
彰化縣	011
南投縣	012
嘉義市	013
嘉義縣	014
雲林縣	015
台南市	016
台南縣	017
高雄市	018
高雄縣	019
屏東縣	020
台東縣	021
花蓮縣	022
澎湖縣	023
金門縣	024
連江縣	025
南海諸島	026
釣魚台列嶼	027
屏東縣琉球鄉離島	028
台東縣綠島、蘭嶼	029

## V、 ReturnURL、 UpdateURL 用途與說明

### ReturnURL：

用於商家進行授權交易，交易完成時欲將消費者瀏覽器轉導之最終結果頁面，因此頁面為前端頁面，故此頁面請僅用於顯示交易結果予消費者，切勿使用此頁面接收之回傳資料進行更新資料庫等作業，以避免因使用者瀏覽器操作流程錯誤或接收錯誤、竄改資料導致資料更新失敗。

### UpdateURL：

用於商家進行授權交易，交易完成時我司將會以HTTP POST方式回傳交易資訊至此URL，商家需於此頁面起始處回傳 "R01=00" 程式片段，我司將會以此訊息判斷是否有回傳資訊成功，若我司系統未正確接收此段訊息，將會重新回傳 3 次交易資訊，若皆失敗則會寄發通知信件至商家技術聯絡人信箱，以通知商家進行反向查詢作業 (Query 函式) 將訂單資訊重新更新回商家端。

同時請商家以此頁面收到之交易資訊更新商店訂單狀態，建議商家以交易回傳代碼Retcode是否等於"00" 判斷交易是否成功再進行後續作業。

## Chapter 8、版本更新說明

### Version 3.1.5

---

新增定期定額訂單取消

### Version 3.1.3

---

修正方便付規格

### Version 3.1.2

---

新增隨身付 ( Follow Pay ) 說明

### Version 3.1.1

---

回傳訊息對照表

### Version 3.1.0

---

LINE Pay

### Version 3.0.8

---

定期定額終止扣款

### Version 3.0.7

---

新增支援 TLS 1.2

### Version 3.0.6

---

新增定期定額查詢

### Version 3.0.5

---

新增速速付支援 3DS 驗證

### Version 3.0.4

---

新增速速付功能說明

### Version 2.0.1

---

修改技術文件說明.

### Version 2.0

---

整合交易元件.

#### Version 1.1

---

新增國民旅遊卡相關欄位.

#### Version 1.0

---

初版.