



\* Sólo Emisores adheridos.

# Manual de Integración

*KIT DE CONEXIÓN DE COMERCIO KCC*  
*VERSIÓN 6.0.2*  
V 2.2

# Contenido

NUEVO EN ESTA VERSIÓN .....	4
1. INTRODUCCIÓN .....	5
2. ¿QUÉ ES EL KCC? .....	7
2.1 PLATAFORMAS SOPORTADAS .....	7
2.2 COMPATIBILIDAD NAVEGADORES Y SMARTPHONES .....	7
3. AMBIENTES DE INCORPORACIÓN A WEBPAY .....	10
3.1 AMBIENTE DE INTEGRACIÓN .....	10
3.2 AMBIENTE DE PRODUCCIÓN .....	10
4. ATRIBUTOS DE WEBPAY .....	11
5. CLASIFICACIONES WEBPAY .....	11
5.1 TIPOS DE TRANSACCIONES SOPORTADA EN ESTA VERSIÓN .....	11
5.2 TIPOS DE MONEDA .....	11
5.3 TIPOS DE CAPTURA .....	12
5.4 TIPOS DE PAGO .....	13
5.5 MEDIOS DE CONEXIÓN .....	14
6. ARQUITECTURA DEL KCC .....	15
6.1 DEFINICIONES GENERALES .....	15
6.1.1 Subprogramas .....	15
6.1.2 Flujo de Información entre Servidor de Comercio y Servidor de Pago .....	17
6.1.3 Condiciones de Excepción .....	19
7. INTEGRACIÓN DEL KCC CON EL SOFTWARE DE LA TIENDA VIRTUAL .....	20
7.1 INSTALACIÓN DEL KCC .....	20
7.1.1 Creación de estructura de Directorios .....	20
7.1.2 Definición de Permisos .....	21
7.1.3 Copia de Archivos .....	22
7.2 CONFIGURACIÓN DEL KCC .....	25
7.2.1 Configuración del archivo <i>tbk_config.dat</i> .....	26
7.2.2 Configuración del archivo <i>tbk_trace.dat</i> .....	28
7.2.3 Habilitación de las formas de pago en Webpay .....	29
7.3 CREACIÓN DE PÁGINAS DE COMERCIO .....	31
7.3.1 Página de Pago .....	31
7.3.2 Página de Cierre para confirmar las transacciones .....	32
7.3.3 Página de Éxito .....	33
7.3.4 Página de Fracaso .....	33
7.4 VALIDACIONES OBLIGATORIAS EN PÁGINA DE CIERRE .....	34
7.4.1 Validación de Monto .....	34
7.4.2 Validación de MAC .....	34
7.4.3 Validación de Duplicidad de Orden de Compra .....	35
7.5 GENERACIÓN DE LLAVES ASIMÉTRICAS PRODUCTIVAS .....	37
8. DESCRIPCIÓN DE TIPOS DE TRANSACCIONES .....	38
8.1 TRANSACCIÓN NORMAL .....	39
8.2 TRANSACCIÓN MALL VIRTUAL .....	41
8.3 TRANSACCIÓN NORMAL COMPLETA .....	43

<b>8.5 TRANSACCIÓN CONSULTA DE TASA.....</b>	<b>47</b>
<b>9. ARCHIVOS GENERADOS POR EL KCC .....</b>	<b>48</b>
9.1 ARCHIVO DE LOG DE EVENTOS.....	48
9.2 ARCHIVO DE LOG DE TRANSACCIONES (BITÁCORA) .....	50
9.3 CÓDIGOS DE RESPUESTA .....	51
<b>10. SOPORTE OPERATIVO .....</b>	<b>52</b>
<b>ANEXO A.....</b>	<b>53</b>
DATOS AMBIENTE DE INTEGRACIÓN DE WEBPAY.....	53
<b>ANEXO B.....</b>	<b>56</b>
RECOMENDACIONES A LA SEGURIDAD DEL KCC. ....	56
<b>ANEXO C.....</b>	<b>57</b>
ALCANCE DE LA VALIDACIÓN DE INTEGRACIÓN ENTRE EL COMERCIO Y TRANSBANK. ....	57
<b>ANEXO D.....</b>	<b>59</b>
TIPIFICACIÓN DE ERRORES KCC.....	59
<b>ANEXO E.....</b>	<b>69</b>
CÁLCULO DEL VALOR CUOTA .....	69
<b>ANEXO F.....</b>	<b>70</b>
EJEMPLO INTEGRACIÓN DE TIENDA TRANSACCIÓN NORMAL ASP .....	70
<b>ANEXO G .....</b>	<b>79</b>
EJEMPLO INTEGRACIÓN DE TIENDA TRANSACCIÓN NORMAL PHP.....	79
<b>ANEXO H .....</b>	<b>88</b>
EJEMPLO INTEGRACIÓN DE TIENDA TRANSACCIÓN NORMAL COMPLETA PHP .....	88
<b>ANEXO I.....</b>	<b>99</b>
EJEMPLO INTEGRACIÓN DE TIENDA TRANSACCIÓN MALL COMPLETA PHP .....	99

## NUEVO EN ESTA VERSIÓN

La nueva versión incluye lo siguiente:

- Se modifica la URL del ambiente de integración  
De:

- ✓ SERVERTRA : <http://certificacion.webpay.cl>
- ✓ PORTTRA : 6443

A:

- ✓ SERVERTRA : <https://webpay3gint.transbank.cl>
- ✓ PORTTRA : 443

- Se modifican los códigos de comercio de integración.
- Se modifica la llave pública declarada en el presente manual
- Se actualizan los ejemplos de integración con los nuevos códigos de comercio.
- Se actualizan los MD5 correspondiente a las piezas de software entregadas en el kit de conexión de comercios.

# 1. INTRODUCCIÓN

Webpay es un servicio de pago desarrollado por Transbank para realizar transacciones vía Internet con tarjetas bancarias de crédito y débito. Hoy en día constituye una herramienta clave para el desarrollo de un comercio electrónico eficaz y seguro en Chile.

Este servicio se sustenta a través de un software de aplicación, que se separa en dos componentes independientes que interactúan entre sí. Una de ellas está ubicada físicamente en la tienda virtual, dentro del servidor de comercio en donde se genera una posible compra por intermedio del browser de un cliente y la otra en el denominado Servidor de Pago de Transbank a través del cual se realiza la transacción.

Secuencia Webpay:

1. Esta comienza cuando el potencial cliente de una tienda virtual presiona el botón pagar.
2. En ese instante la tienda se comunica con Transbank, por medio del KCC, para solicitar autorización para transar la compra.
3. En este punto, la aplicación ubicada en el Servidor de Pago toma el control de la transacción comunicándose directamente con el cliente, solicitando los datos de su tarjeta.
4. Una vez realizado el envío de dicha información directamente a Transbank, se verifica la disponibilidad de cursar la transacción.
5. En caso de que la tienda opere con Webpay Plus, se solicita la autenticación del tarjetahabiente en su emisor.
6. El emisor despliega página de autenticación para el tarjetahabiente.
7. El tarjetahabiente ingresa los datos solicitados para autenticación, estos dependen de cada emisor.
8. El emisor envía a Transbank el resultado de la autenticación, si esta es positiva continua con los pasos siguientes, en caso contrario, Webpay rechaza la transacción.
9. Webpay solicita la autorización de la transacción al autorizador del emisor de la tarjeta de crédito.
10. Autorizador responde (aprobado, rechazado) a Transbank.
11. Transbank responde a Webpay el resultado de la autorización de la transacción.
12. Luego se le envía una respuesta positiva o negativa a la tienda virtual, así esta puede cerrar la transacción ya sea en forma exitosa o no.

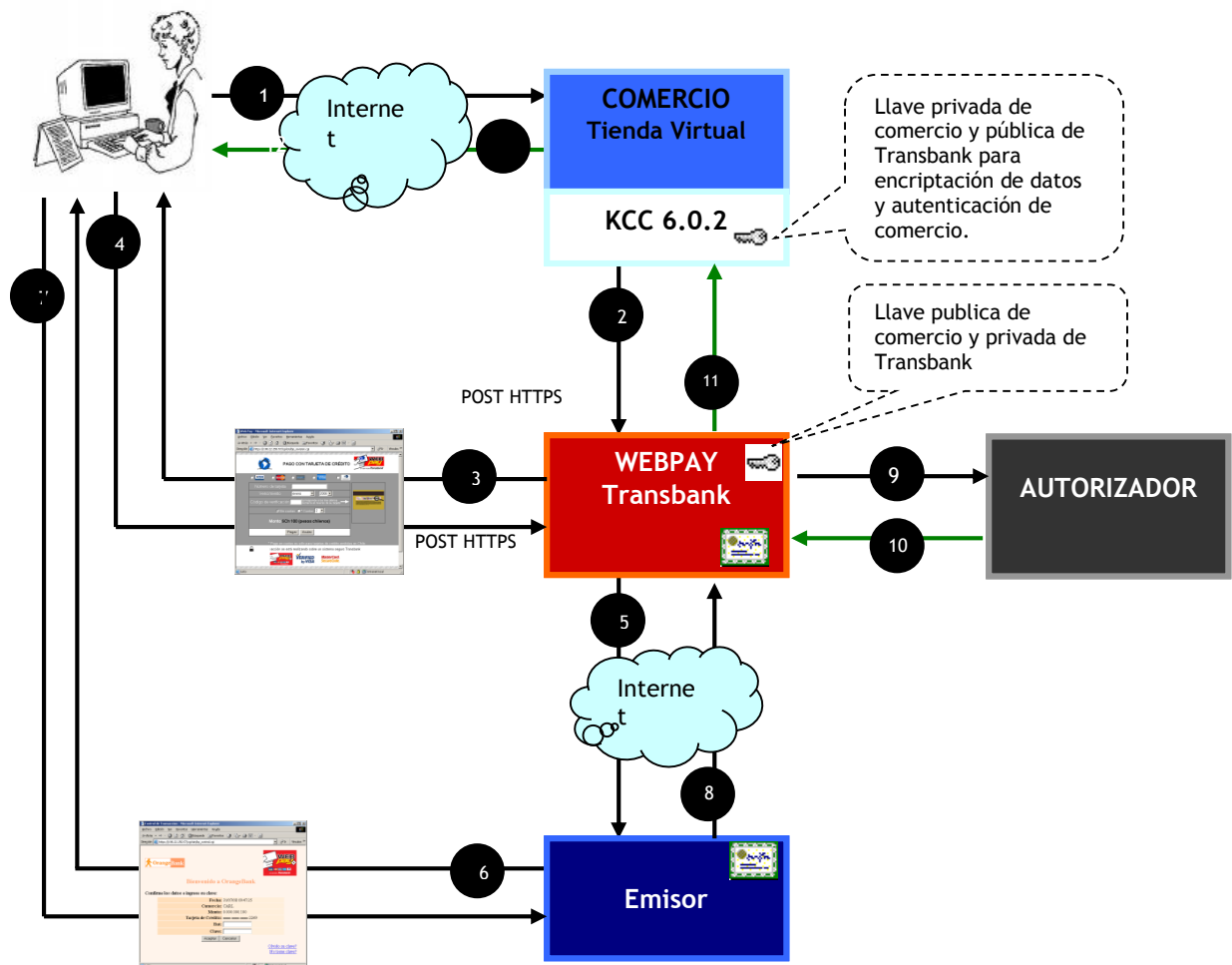


Figura 1. Secuencia de Transacción Electrónica con Webpay Plus

## 2. ¿QUÉ ES EL KCC?

Es la aplicación que se instala en el Servidor de Comercio que permite conectar el software utilizado por la tienda para vender sus productos y/o servicios, con el Servidor de Pago de Transbank. Es un Kit De Conexión al Comercio, identificado con la sigla KCC.

Para que el KCC se conecte y logre comunicarse sin problemas con el Servidor de Pago es necesario su correcta instalación y configuración por parte del comercio afiliado al servicio Webpay de Transbank. Así la finalidad de este manual es explicar en detalle cómo realizar la integración entre el KCC y el software de venta de la tienda virtual.

La integración del KCC, consta de 2 etapas:

- Etapa de Validación: Corresponde a la etapa en donde se valida el correcto funcionamiento de la integración del KCC en la tienda del comercio.
- Etapa de Producción: Corresponde al funcionamiento productivo de la solución en régimen de operación normal.

### 2.1 PLATAFORMAS SOPORTADAS

Las siguientes son las plataformas soportadas por KCC

- Windows 2008 server o superior, 32 bits y 64 bits
- Linux, 32 bits y 64 bits en x86 (kernel 2.6.x o superior)
- Solaris 10 o superior para SPARC

### 2.2 COMPATIBILIDAD NAVEGADORES Y SMARTPHONES

#### Navegadores

Transbank garantiza el correcto funcionamiento de KCC y Webpay bajo los siguientes navegadores Web:

- Internet Explorer 6, 7, 8 y 9
- Mozilla Firefox 10
- Chrome 17
- Safari 5

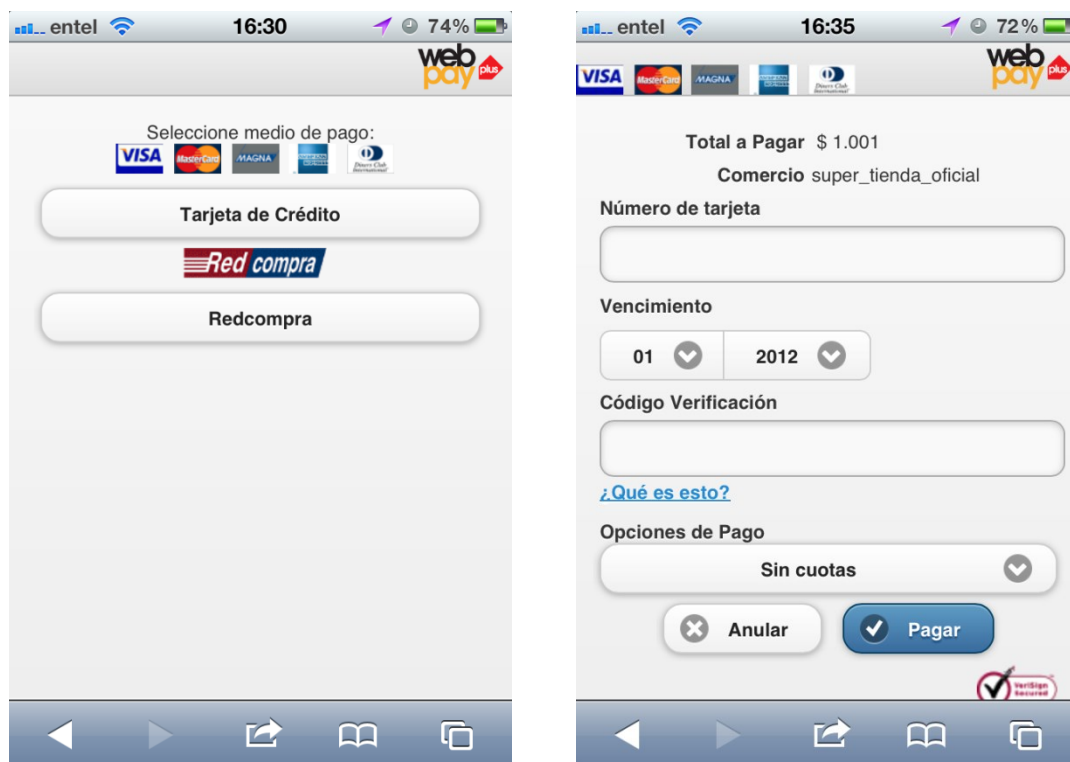
**IMPORTANTE:** Es de responsabilidad del comercio garantizar la compatibilidad de su sitio Web con navegadores Web, como también tomar todas las medidas necesarias de seguridad para garantizar una compra segura. Se sugiere uso de certificados digitales y evitar uso de pop-up emergentes para evitar problemas de bloqueo con últimas versiones de navegadores Web.

## Smartphones

Transbank garantiza el correcto funcionamiento de Webpay bajo los siguientes dispositivos móviles:

- BlackBerry 9900 y 9700
- iPhone, en cualquiera de sus versiones
- Dispositivos Android, en cualquiera de sus versiones

Desplegando la siguiente interfaz:



Los dispositivos **excluidos** en WebPay son los siguientes:

- Blackberry 8900 y 8520
- SymbianOS

Que sean excluidos quiere decir que se despliega un mensaje indicando “dispositivo no soportado” y no permite realizar transacciones. En cualquier otro dispositivo que ingrese a WebPay, se visualizará la versión normal.

**IMPORTANTE:** Es de responsabilidad del comercio garantizar la compatibilidad de su sitio Web con dispositivos móviles, como también tomar todas las medidas necesarias de seguridad para garantizar una compra segura.

El despliegue en dispositivos móviles de la página de autenticación bancaria, es de exclusiva responsabilidad del banco emisor.



## 2.3 VALIDACIÓN DE INTEGRIDAD Y AUTENTICIDAD DEL KCC

Con el objetivo de asegurar la integridad y autenticidad del Kit de Conexión de Comercio, se ha incorporado en este manual el hash MD5 correspondiente a los principales componentes del kit para cada plataforma.

Plataforma	Componente	Hash md5
Linux 32 bits	tbk_bp_pago.cgi	7CBF9D97256E4534D97C90DF3A0DAD1A
	tbk_bp_resultado.cgi	54EAE0E24D7F31D6C029EA5A59161AD8
	tbk_check_mac.cgi	216293459114F163B9AEF2BAA4D746E8
Linux 64 bits	tbk_bp_pago.cgi	7AFA22A2C9C27558A15CBCDC92DDF66D
	tbk_bp_resultado.cgi	29C00C15D81135DB10797372344125D1
	tbk_check_mac.cgi	F70B6D704FD3A66F6E2B686F20E314C5
Sun solaris 6.0.2	tbk_bp_pago.cgi	2A8CD1D64A13B4E6A322EA5D813F092F
	tbk_bp_resultado.cgi	B5476997CA0EF37A4939F05B6715A685
	tbk_check_mac.cgi	D230BE2C8AE6D81098181A1AFC48CDE5
Windows	tbk_bp_pago.cgi	6EA355871363D7A2817273EFC4F19618
	tbk_bp_resultado.cgi	15F673786D81FD727935EC086D047AB2
	tbk_check_mac.exe	C68765339AC050315F36377C9B0B4E0E

Información:

md5sum es un comando de los sistemas Unix que permite identificar la integridad de un fichero mediante de la suma de comprobación del hash MD5 de un archivo.

Es muy utilizado por su sencillez de uso, potencia y popularidad, siendo relativamente sencillo el comprobar si un determinado archivo se ha descargado correctamente o por el contrario ha ocurrido algún problema.

Ejemplo de uso:

```
#md5sum tbk_bp_pago.cgi
```

Resultado esperado para KCC Linux kernel 32: 7CBF9D97256E4534D97C90DF3A0DAD1A

### 3. AMBIENTES DE INCORPORACIÓN A WEBPAY

Para el proceso de incorporación de Webpay a una tienda virtual existen dos ambientes de conexión:

- Ambiente de Integración.
- Ambiente de Producción.

#### 3.1 Ambiente de Integración.

El ambiente de integración es un ambiente especialmente creado para efectuar, en modalidad de pruebas, transacciones Webpay. En este ambiente, las transacciones son simuladas, por lo cual no se requieren datos reales, sino de pruebas.

Durante el proceso de integración se le entrega al cliente (comercio) el KCC versión **6.0.2** para que lo integre a su tienda virtual y realice transacciones hacia el ambiente de integración. **Para mayor información consultar el Anexo A.**

Una vez realizada la integración es necesario validarla, para esto se realizan transacciones de prueba desde la tienda virtual hacia el ambiente de integración, y se verifica el funcionamiento global del sistema. Referirse al Anexo C para ver detalles exigidos en este proceso.

Para fines de configuración, la URL y puerto de comunicación con Webpay de integración de Transbank son:

- SERVERTRA : https://webpay3gint.transbank.cl
- PORTTRA : 443

#### 3.2 Ambiente de Producción

Una vez finalizada la integración y validada con éxito, el comercio pasa a la última fase denominada producción en donde la tienda ya está habilitada para trabajar con Webpay sin problemas.

Para fines de configuración, la URL y puerto de comunicación con Webpay productivo de Transbank son:

- ✓ SERVERTRA : https://webpay.transbank.cl
- ✓ PORTTRA : 443

## 4. ATRIBUTOS DE WEBPAY

- Permite realizar transacciones seguras y en línea a través de Internet.
- La información de la Tarjeta de Crédito del cliente sólo es manejada por Transbank, y no es de conocimiento en ningún momento por el comercio.
- Transacciones con Webpay Plus se solicita al tarjetahabiente autenticarse con su emisor, protegiendo de esta forma al comercio por eventuales fraudes o desconocimientos de compra.
- La seguridad es reforzada por medio de la utilización de servidores seguros, protegidos con SSL (Secure Socket Layer).
- Encriptación de mensajería y firma digital.

## 5. CLASIFICACIONES WEBPAY

Tomando en cuenta que cada comercio es en si una entidad única y con requerimientos propios, el sistema Webpay soporta dos tipos de transacciones, las cuales puede brindar una solución tecnológica a una amplia gama de modelos de operación de tiendas virtuales.

### 5.1 Tipos de Transacciones soportada en esta versión

Tipo Transacción	Descripción
Normal	Permite la compra de productos y/o servicios a través de Internet. Esta modalidad opera de forma que uno, o varios productos y/o servicios agrupados en un carro de compra entregan un total final, de venta a pagar. Está directamente asociado a una sola tienda virtual.
Mall Virtual	Permite que un grupo de tiendas virtuales realicen ventas en conjunto generando sólo un total de venta final.

### 5.2 Tipos de Moneda

Además de las clasificaciones de empleo de Webpay mencionadas anteriormente, se permite elegir el tipo de moneda de la transacción, es decir, peso o dólar. Esto debido a que se pueden realizar compras con tarjetas emitidas en el extranjero con cupo en dólares y validas internacionalmente.

### 5.3 Tipos de Captura

Se permiten dos formas de captura de la transacción electrónica estas son:

Tipo de Captura	Descripción
Simultanea	Es la utilizada cuando la transacción es validada en línea por Transbank. El cargo del pago se hace simultáneamente en la tarjeta de crédito del cliente.
Captura Diferida	Es utilizada cuando el comercio por diversas razones, ejemplo: verificación de stock, no puede realizar la transacción en tiempo real, aquí lo que se hace es retener el valor de la compra del saldo de la tarjeta del cliente sin validar la transacción hasta que el comercio apruebe la compra y lo comunique a Transbank. Existe un tiempo máximo estipulado para realizar esto último, de no ser así la retención de la tarjeta de crédito del cliente será reversada.

Observación: A pesar que el KCC ha sido desarrollado para ser utilizado por cualquier tienda virtual, este debe ser integrado a cada comercio en particular de acuerdo al tipo de plataforma, transacción y software de comercio utilizado.

## 5.4 Tipos de Pago

Actualmente existen cuatro tipos de pago posibles. Estas se detallan a continuación:

Tipo de Pago			Descripción
CREDITO	VENTA NORMAL	SIN GRACIA	Corresponde al pago total del precio establecido por el Comercio para el producto en cuestión.
		CON GRACIA	Corresponde al pago total del precio establecido por el Comercio para el producto en cuestión, con un mes de desfase para el TH y para el comercio. Requiere adherencia adquirente.
	VENTA CUOTAS	CUOTAS SIN INTERÉS	Corresponde al pago de un producto en cuotas iguales sin intereses para el TH. Puede ir de 2 hasta "N" cuotas. Con "N" máximo 12, y definido por el Comercio. El abono al comercio también es en cuotas.
		CUOTAS SIN INTERÉS DEL EMISOR	Cuotas sin intereses para el TH entregadas por una promoción del emisor de la tarjeta. Puede ir de 2 hasta 48. El Emisor escoge el rango en promoción.
		CUOTAS SIN INTERÉS DEL ESTABLECIMIENTO	Cuotas sin intereses para el TH entregada por una promoción del Comercio. Puede ir de 4 hasta 24. El Comercio escoge el rango en promoción.
		CUOTA CON TASA DE INTERÉS ÚNICA	Corresponde a cuotas con intereses para el TH. Donde el Comercio puede publicar el valor de la cuota dado que conoce previamente la tasa de interés entregada por Transbank. Va desde 4 a 24 cuotas.
		CUOTAS NORMALES	Cuotas normales o cuotas fijas van de 2 hasta 48 y se les aplica el interés definido por cada Emisor. Si éste se encuentra adherido a diferimiento puede ofrecer desde 1 hasta 3 meses de diferido (sin impacto para el Comercio).
DEBITO	REDCOMPRA		Pago a través de tarjeta de débito Redcompra

Observación:

En la forma de pago “Cuotas Comercio”, la única condición que Transbank impone a los comercios, es que en el cálculo del monto de la cuota no supere el límite correspondiente a la tasa de interés máxima legal. Para velar por que tal condición se cumpla, se realiza una validación de esta tasa para todas aquellas transacciones que utilicen esta forma de pago. Se deben considerar una cantidad de cuotas entre 6 y 48. (Ver Anexo E).

## 5.5 Medios de Conexión

El medio de conexión indica a KCC la forma en la que se comunicará con TBK. Esto se define a través del parámetro MEDCOM que se encuentra ubicado en el archivo tbk\_config.dat.

El valor de este parámetro es 1.

Este parámetro permite que se abra un canal SSL (canal de comunicación seguro) entre el comercio y Transbank, por el cual se envían los datos encriptados y en segundo plano generando un token identificador de la transacción que se le envía al tarjeta habiente para su redirección al formulario de pago Webpay.

Observaciones:

- El parámetro MEDCOM = 1, como se indicó anteriormente, requiere de SSL para su funcionamiento, lo que implica que el comercio pueda establecer una comunicación directa con las IP's de Transbank al puerto 443 (la url de conexión es <https://webpay.transbank.cl>)
- La url de Transbank (Integración) a las que el comercio **debe permitir el tráfico y filtrar por url (comercio -> Transbank)**, es la siguiente:  
<https://webpay3gint.transbank.cl> (url de desarrollo)
- Se recomienda filtrar por la url listadas anteriormente el acceso a la cgi tbk\_bp\_resultado (ver sección 7.3.2) para asegurar que sólo Transbank pueda ejecutarla.

## 6. ARQUITECTURA DEL KCC

### 6.1 Definiciones generales

El KCC es una aplicación que básicamente está compuesta por:

- Subprogramas
- Archivos de configuración
- Archivos de seguridad
- Archivos de registro
- Páginas Web
- Llave pública de Transbank

#### 6.1.1 Subprogramas

Subprograma	Nombre del Archivo
Pago	tbk_bp_pago
Resultado de la Transacción	tbk_bp_resultado
Verificación de Firma Digital	tbk_check_mac

Tabla 1. Subprogramas utilizados en el KCC.

La extensión de tbk\_check\_mac puede variar de .CGI a .EXE de acuerdo a la plataforma utilizada. (.CGI para Linux y .EXE para Windows).

#### TBK\_BP\_PAGO

Es el encargado de hacer el requerimiento al Servidor de Pagos, derivándole a Transbank, el control del browser del tarjetahabiente.

Funciones que realiza:

- Firma el mensaje, agregando el MAC (código de encriptación).
- Conecta con Servidor de Pagos para envío de parámetros encriptados, recibiendo de dicha conexión el Token o número único de la transacción.
- Redirecciona el Tarjetahabiente a Transbank con el Token dado.
- Registra en el log de eventos:
- Resultado de las operaciones.
- Errores ocurridos.

## TBK\_BP\_RESULTADO

---

Se encarga de recibir la respuesta desde el Servidor de Pagos, transferirla al software del comercio y solicitar la entrega del control del browser del tarjetahabiente a una página definida por la tienda.

Funciones que realiza:

- Valida el MAC.
- Recibe resultado cifrado de la transacción.
- Descifra parámetros recibidos.
- Confirma con el comercio el cierre de las compras pasando todos los parámetros recibidos.
- Envía la confirmación o rechazo del cierre de la compra con el comercio (aceptación o rechazo por parte del comercio).

## TBK\_CHECK\_MAC

---

Encargado de validar, la firma digital en las respuestas que entrega Transbank al comercio. Es una herramienta que se utiliza para asegurar que la respuesta provenga exactamente desde Transbank.



### 6.1.2 Flujo de Información entre Servidor de Comercio y Servidor de Pago

A continuación se explica el flujo de información para el tipo de transacción normal entre el Servidor de Comercio y el Servidor de Pago.

1. El tarjetahabiente realiza una compra con tarjeta de crédito en una tienda virtual a través del servicio Webpay de Transbank.
2. El software de la tienda ejecuta `tbk_bp_pago` entregándole los siguientes parámetros vía POST:
  - Monto de la compra.
  - Número de Orden de Compra<sup>1</sup>.
  - URL de Éxito.
  - URL de Fracaso.
3. `tbk_bp_pago` se conecta con `bp_validacion` ubicado en el servidor de Transbank.
4. `bp_validacion` desencripta la información, valida al comercio y le asigna una sesión transaccional llamada Token.
5. `bp_validacion` retorna token a `tbk_bp_pago` del comercio, `tbk_bp_pago` redirecciona al tarjetahabiente a `bp_revision` de Transbank.
6. `bp_revision` busca información de la transacción según Token y despliega la ventana de solicitud de datos al tarjetahabiente.
7. El tarjetahabiente digita los datos requeridos de su tarjeta y presiona el botón “Pagar”.
8. En caso de operar con Webpay Plus, se solicita al tarjetahabiente su autenticación con el emisor. Una vez autenticada correctamente, se solicita la autorización de la transacción.
9. La transacción es autorizada o rechazada, y su resultado es comunicado por Webpay a través de parámetros encriptados `bp_transaccion` (Servidor de Pago), a `tbk_bp_resultado` en el Servidor de Comercio.
10. **En caso de aprobación**, `tbk_bp_resultado` ejecuta una página (página de cierre), en la tienda virtual, comunicándole el número de la orden de compra y el código de autorización asignado por Transbank. La tienda virtual busca en su base de datos el número de la orden de compra, verifica que no está ya pagada y procede a cerrar la transacción registrando el código de autorización de Transbank. Una vez realizado esto se devuelve un string de “ACEPTADO” que indica que la operación fue realizada con éxito.

**En caso de rechazo**, `tbk_bp_resultado` registra la transacción en su archivo log y confirma el resultado a Transbank. Los rechazos comúnmente ocurren por alguna de las siguientes razones:

- Tarjeta bloqueada por robo

---

<sup>1</sup> La orden de compra es generada por el comercio, esta debe tener un largo máximo de 26 caracteres. Transbank no valida la orden de compra informada por el comercio.

- CVV invalido
- Fecha de expiración inválida
- Falta de fondo
- Tarjeta expirada
- Autenticación inválida
- Time out
- Transacción abortada. Tarjeta habiente cierra browser de la página de autenticación.
- Autenticación inválida.

En caso de rechazo se devuelve un string de “ACEPTADO” que indica que confirma el rechazo.

No confundir los rechazos con los errores de comunicación que pudiesen ocurrir entre el comercio y Transbank.

11. tbk\_bp\_resultado recibe “ACEPTADO” y confirma la transacción hacia Transbank.
12. Finalmente se redirecciona la página del tarjetahabiente a la página de éxito o fracaso según corresponda.

Observación:

El string “ACEPTADO” debe ser devuelto en mayúsculas para que este sea reconocido por la aplicación.

El flujo de información se muestra gráficamente en la Figura 2.



Figura 2. Flujo de Información Webpay Modalidad Normal

### 6.1.3 Condiciones de Excepción

Las condiciones de excepción son ciertas instancias del proceso que hacen que la transacción sea reversada cuando no se cumplen algunas condiciones básicas, estas son:

- **Timeout:** El timeout se produce cuando el tiempo de espera, por parte de Transbank, de la comprobación (ACK), de que fue realizado el cierre de la compra por parte del comercio es elevado. Se asume por lo tanto que la compra nunca fue realizada. Tiempo límite 30 segundos.
- **Error en Cierre de la Compra:** Este ocurre cuando el String de confirmación del cierre de la compra es distinto de “ACEPTADO”.

## 7. INTEGRACIÓN DEL KCC CON EL SOFTWARE DE LA TIENDA VIRTUAL

En la instalación del KCC en el sitio del comercio hay aspectos que varían entre un comercio y otro. Esto debido a que no todos los comercios tienen los mismos software instalados (Servidor Web y Software de Comercio Electrónico principalmente), así hay algunas decisiones en la instalación que se dejan a criterio del comercio.

Los pasos necesarios para la integración del KCC genérico con un software de comercio electrónico son los siguientes:

Pasos	Detalle
Instalación del KCC.	<ul style="list-style-type: none"><li>- Creación de estructura de directorios.</li><li>- Definición de permisos.</li><li>- Copia de archivos.</li></ul>
Configuración del KCC	<ul style="list-style-type: none"><li>- Archivo de configuración <i>tbk_config.dat</i>.</li><li>- Archivo de configuración <i>tbk_trace.dat</i></li></ul>
Creación de páginas de comercio.	<ul style="list-style-type: none"><li>- Páginas de llamada al KCC.</li><li>- Páginas para confirmar las transacciones.</li></ul>

### 7.1 Instalación del KCC

#### 7.1.1 Creación de estructura de Directorios

##### Directorio Base

---

La aplicación se debe alojar en un directorio base para facilitar su ubicación y posterior configuración, así se debe crear un directorio base denominado, por ejemplo, KCC.

##### Directorio CGI-BIN

---





El segundo paso para la instalación del KCC en el sitio del comercio es crear un directorio de CGIs dentro del directorio base. Para ello hay que ejecutar las siguientes acciones:

1. Crear un directorio vacío con el nombre `$(PATH_DISCO_DIRCGI)`.
2. Configurar el servidor Web para agregar un directorio de CGIs en la ubicación `$(PATH_DISCO_DIRCGI)` cuya ubicación lógica sea `PATH_WEB_DIRCGI`.
3. Asignar permisos de ejecución para el usuario Web.

##### Sub-Directorios

---

El KCC posee una estructura de directorios determinada para alojar sus archivos, así dentro del directorio \$(PATH\_DISCO\_DIRCGI), se deben crear los siguientes subdirectorios:

-  datos
-  log
-  maestros
-  template

### 7.1.2 Definición de Permisos

#### Sistema Operativo

---

Para el usuario web se le deben otorgar los siguientes permisos a los directorios creados.

Directorio	Lectura	Escritura	Ejecución
cgi-bin			x
datos	x		
log	x	x	
maestros	x		
Template	x		

Tabla 2. Asignación de Permisos.

Observación: En el caso de Unix, se debe agregar permiso de ejecución a todos los directorios para poder acceder a los archivos ubicados dentro de ellos.

#### Servidor Web

---

Se deben habilitar sólo los siguientes permisos en el Servidor Web a los directorios virtuales:

Directorio Virtual	Atributos
/ cgi-bin	Permitir ejecución de cgi utilizando el método POST.
/ cgi-bin / log	Ningún permiso
/ cgi-bin / datos	Ningún permiso
/ cgi-bin / maestros	Ningún permiso

### 7.1.3 Copia de Archivos

Dentro de la estructura de directorios definida anteriormente se deben copiar los siguientes archivos.

<b>tbk_bp_pago.cgi</b>	
Descripción	Programa CGI, módulo ejecutable. Permite: <ul style="list-style-type: none"><li>• Traspaso de datos de la transacción y del comercio a Transbank.</li><li>• Firmar el mensaje, agregando el MAC.</li><li>• Conectar a Servidor de Pago para toma de parámetros.</li><li>• Redireccionar al Tarjetahabiente al Servidor de Pago en Transbank.</li></ul>
Ubicación	\$(PATH_DISCO_DIRCGI)
Permiso	Lectura y Ejecución

<b>tbk_bp_resultado.cgi</b>	
Descripción	Programa CGI, módulo ejecutable. Permite: <ul style="list-style-type: none"><li>• Recibir resultado de la transacción.</li><li>• Validar el MAC.</li><li>• Confirmar con el comercio el cierre de la compra, a través de la página de cierre (ver punto 8.3.2).</li><li>• Realizar la confirmación, ya sea exitosa o no, del cierre de la compra, enlazándose con el CGI bp_transaccion, ubicado en el Servidor de Pago de Transbank encargado de realizar el redireccionamiento del cliente a la página de éxito o fracaso de la transacción del comercio.</li></ul>
Ubicación	\$(PATH_DISCO_DIRCGI)
Permiso	Lectura y Ejecución

<b>tbk_check_mac.exe (Windows) - tbk_check_mac.cgi (otras plataformas)</b>	
Descripción	Programa que permite asegurar que la página de confirmación del comercio es llamada desde el programa tbk_bp_resultado.cgi. Se utiliza para brindar una mayor seguridad a la transacción.
Ubicación	\$(PATH_DISCO_DIRCGI)
Permiso	Lectura y Ejecución

<b>tbk_check_mac.bat (para Windows)</b>	
Descripción	Archivo de uso opcional, que centraliza la llamada a tbk_check_mac.exe y dirige la salida hacia un archivo para su posterior lectura.
Ubicación	(PATH_DISCO_DIRCGI)
Permiso	Lectura y Ejecución.

<b>tbk_config.dat</b>	
Descripción	Archivo de configuración del KCC. Permite almacenar datos de configuración que definirán la forma en que operará la tienda en cada transacción. Puede ser editado con un editor de texto.
Ubicación	\$(PATH_DISCO_DIRCGI)/datos
Permiso	Lectura.

<b>tbk_trace.dat</b>	
Descripción	Archivo de configuración que define el nivel de detalle del registro de eventos del KCC (Log), puede ser editado con un editor de texto.
Ubicación	\$(PATH_DISCO_DIRCGI)/datos
Permiso	Lectura.

<b>Tbk_bitacora_tipo_de_transaccion_fecha.dat</b>	
Descripción	<p>Archivo que permite almacenar el resultado de las operaciones realizadas durante el mes de acuerdo al tipo de transacción habilitado. Se genera por fecha contable.</p> <p>Donde:</p> <p>tipo_de_transacción:</p> <ul style="list-style-type: none"> <li>• Normal: tr_normal.</li> <li>• Mall Virtual: tr_mall</li> </ul> <p>fecha:</p> <ul style="list-style-type: none"> <li>• mm: dos dígitos de mes.</li> <li>• dd: dos dígitos de día.</li> </ul> <p>Puede ser visualizado con un editor de texto.</p>
Ejemplo	tbk_bitacora_tr_normal_0407.log
Ubicación	\$(PATH_DISCO_DIRCGI)/log
Permiso	Lectura y escritura.

<b>tbk_evn_fecha.dat</b>	
Descripción	<p>Archivo que permite almacenar el resultado de las operaciones. Se genera por día.</p> <p>Donde Fecha:</p> <ul style="list-style-type: none"> <li>• aaaa: cuatro dígitos de año.</li> <li>• mm: dos dígitos de mes.</li> <li>• dd: dos dígitos de día.</li> </ul> <p>Puede ser visualizado con un editor de texto.</p>
Ejemplo	tbk_evn20000410.log
Ubicación	\$(PATH_DISCO_DIRCGI)/log
Permiso	Lectura y escritura.

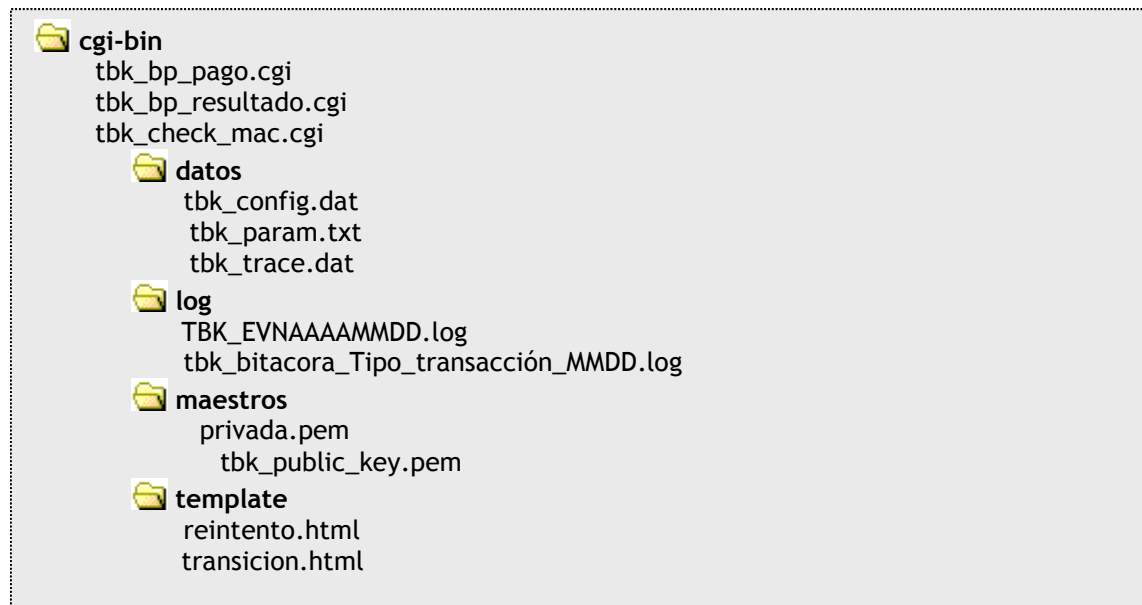
<b>privada.pem</b>	
Descripción	Llave de encriptación de información enviada a través de Internet a Transbank.
Ubicación	\$(PATH_DISCO_DIRCGI)/maestros.
Permiso	Lectura.

<b>Reintento.htm</b>	
Descripción	Página que se despliega en caso de ocurrir un error de CGI.
Ubicación	\$(PATH_DISCO_DIRCGI)/templates.
Permiso	Lectura.

<b>tbk_param.txt</b>	
Descripción	Archivo de texto que contiene los parámetros que deben recibir los CGIs bp_pago y bp_resultado dependiendo del tipo de transacción.
Ubicación	\$(PATH_DISCO_DIRCGI)/datos
Permiso	Lectura.

<b>tbk_public_key.pem</b>	
Descripción	Llave pública Transbank, utilizada por KCC para tareas de encriptación / firma de mensajes
Ubicación	\$(PATH_DISCO_DIRCGI)/maestros
Permiso	Lectura.

## Estructura de directorios



Observación: Al momento de subir los archivos al servidor de comercio se debe establecer en la aplicación utilizada el modo de envío como sigue:

Tipo	Extensión
ASCII	*.log; *.dat; *.html; *.asp; *.pem, *.txt
BINARIO	*.cgi; *.exe; *.pem



## 7.2 Configuración del KCC

Los archivos de configuración del KCC son:

- tbk\_config.dat
- tbk\_trace.dat

Y los parámetros que en ellos se configuran tienen la forma:

`$(NOMBRE_PARAMETRO) = $(VALOR_PARAMETRO)`

Donde:

- `$(NOMBRE_PARAMETRO)`: Corresponde al nombre del parámetro a configurar.
- `$(VALOR_PARAMETRO)`: Valor asignado a dicho parámetro.

Ejemplo:

`WHITELISTCOM=ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789./:=&?_`

Observación: Solamente puede ir un parámetro por línea.

### 7.2.1 Configuración del archivo *tbk\_config.dat*

Este archivo configura el comportamiento general del KCC, tiene los siguientes parámetros.

IDCOMERCIO	
Valor	\$(CODIGO_COMERCIO)
Descripción	Código numérico de 12 dígitos asignado por Transbank al comercio de acuerdo a la modalidad adoptada.
Ejemplo	IDCOMERCIO = 597020000501

MEDCOM	
Valor	\$(Medio Conexión)
Descripción	Posibles valores: 1 1: Conexión segura (SSL) a bp_validacion y posterior redireccionamiento a bp_revision
Ejemplo	1

PARAMVERIFCOM	
Valor	0 o 1
Descripción	PARAMVERIFCOM es 0, no se realiza el filtro y revisión de parámetros, si es 1, si realiza.
Ejemplo	1

URLCGICOM	
Valor	\$(PATH_WEB_DIRCGI)/tbk_bp_resultado.cgi
Descripción	Path web donde se encuentra el cgi tbk_bp_resultado
Ejemplo	URLCGICOM = /cgi-bin/tbk_bp_resultado.cgi

SERVERCOM	
Valor	\$(SERVIDOR_COMERCIO)
Descripción	Numero IP del servidor Web del comercio donde está el cgi tbk_bp_resultado Largo máximo 12 dígitos.
Ejemplo	SERVERCOM = 198.22.250.29

PORTCOM	
Valor	\$(PUERTO_COMERCIO)
Descripción	Puerto asignado al servidor Web del comercio donde está el CGI tbk_bp_resultado.
Ejemplo	PORT = 80

WHITELISTCOM	
Valor	\$(Lista Blanca)
Descripción	Listado de caracteres permitidos en los parámetros enviados por POST o GET
Ejemplo	ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789./: =&?_

HOST	
Valor	\$(SERVIDOR_CONFIRMACION_COMERCIO)
Descripción	IP del servidor Web del comercio donde están las páginas de confirmación de las transacciones por parte del comercio.
Ejemplo	HOST = 198.22.250.29

WPORT	
Valor	\$(PUERTO_CONFIRMACIÓN_COMERCIO).
Descripción	Puerto asignado al servidor Web del comercio donde están las páginas de confirmación de las transacciones por parte del comercio.
Ejemplo	WPORT = 80

URLCGITRA	
Valor	/cgi-bin/bp_revision.cgi
Descripción	Path Web del cgi bp_revision en Transbank.
Ejemplo	URLCGITRA = /cgi-bin/bp_revision.cgi

URLCGIMEDTRA	
Valor	/cgi-bin/bp_validacion.cgi
Descripción	Path Web del cgi bp_validacion en Transbank.
Ejemplo	URLCGIMEDTRA = /cgi-bin/bp_validacion.cgi

SERVERTRA	
Valor	\$(SERVIDOR_TRANSBANK)
Descripción	Nombre o IP del servidor de Transbank donde se ejecutan las transacciones de WebPay. Incluye el prefijo https.
Ejemplo	SERVERTRA = https://webpay3gint.transbank.cl

PORTTRA	
Valor	\$(PUERTO_TRANSBANK)
Descripción	Puerto asignado al servidor Web de Transbank donde se ejecutan las transacciones Webpay.
Ejemplo	PORTTRA = 443

PREFIJO_CONF_TR	
Valor	\$(PREFIJO)
Descripción	Prefijo que contiene el nombre del parámetro que indica cual es la página que se ejecuta en el comercio para confirmar la transacción.
Ejemplo	PREFIJO_CONF_TR=HTML_

HTML_TR_NORMAL	
Valor	\$(PAGINA_CONFIRMACION)
Descripción	URL absoluta que debe indicar la dirección de la página confirmar la transacción NORMAL.
Ejemplo	http://www.comercio.cl/ xt_compra.asp

HTML_TR_MALL	
Valor	\$( PAGINA_CONFIRMACION)
Descripción	URL absoluta que debe indicar la dirección de la página confirmar la transacción MALL.
Ejemplo	http://www.comercio.cl/ xt_compra.asp

TBK_REFERER	
Valor	\$(REFERER)
Descripción	URL absoluta que debe indicar la dirección de la página encargada de inicializar el proceso de pago, es decir, desde donde se ejecuta el cgi tbk_bp_pago.
Ejemplo	http://www.comercio.cl/pago.asp

TBK_KEY_ID	
Valor	101 o el valor indicado por Transbank
Descripción	Indica la llave publica que tiene cargada el comercio, este valor solo debe ser modificado a solicitud de Transbank.
Ejemplo	101

### 7.2.2 Configuración del archivo *tbk\_trace.dat*

Configura el nivel de logs que reporta el KCC, este tiene los siguientes parámetros.

GLEVEL	
Valor	\$(NIVEL_LOG)
Descripción	Nivel de logs que genera el KCC. Ej.: GLEVEL = 7

Los niveles posibles de log son:

Nivel	Valor	Descripción
LOGINFO	1	El KCC genera logs informativos.
LOGERROR	2	El KCC genera logs de errores.
LOG_FATAL	4	El KCC genera logs Fatales

También existen las combinaciones de nivel de logs. De esta manera los valores posibles del parámetro GLEVEL son:

Valor	Nivel de Log
0	No hay logs.
1	LOGINFO
2	LOGERROR
3	LOGINFO+LOGERROR
4	LOGFATAL
5	LOGFATAL+LOGINFO
6	LOGFATAL+LOGERROR
7	LOGINFO+LOGERROR+LOGFATAL

### 7.2.3 Habilitación de las formas de pago en Webpay

Las formas de pago que acepta cada tienda en particular son una definición comercial. Estas son habilitadas automáticamente por Transbank desde el momento en que la tienda comienza a operar con Webpay.

Las formas de pago venta normal, venta cuotas y tres cuotas sin interés se habilitan en el formulario de pago estándar de Transbank y le permiten al tarjetahabiente seleccionar la opción a utilizar.

**darcon tarjeta.cl**  
El portal de donaciones de Chile

**web pay plus**

**Tarjeta de Crédito**  
VISA MasterCard MAGNA

**Red compra**

**Total a Pagar \$ 5.000**  
Cinco mil pesos chilenos

**Comercio :** INTERNET DARCONTARJE

**Número de tarjeta :**

**Vencimiento :** 01 / 2010  
Ej. 08/2009

**Código Verificación :**   
[¿Qué es esto?](#)

**Detalle**

F.CH.ADOPC.DARCONTAR	\$ 5.000	Sin cuotas
----------------------	----------	------------

**Anular** **Pagar**

**VISA** **MasterCard** **MAGNA** **Red compra**

Esta transacción se está realizando sobre un sistema seguro.

**VERIFIED by VISA** **MasterCard SecureCode**

La forma de pago cuotas comercio se despliega en un formulario de pago especial que no le permite al tarjetahabiente cambiar su opción de pago.

The screenshot shows the 'darcon tarjeta.cl' website header with the 'web pay plus' logo. Below the header, the 'Tarjeta de Crédito' section is active, displaying logos for VISA, MasterCard, MAGNA, and others. The total amount to be paid is \$10,000 (Diez mil pesos chilenos). The merchant is 'INTERNET DARCONTARJE'. The card number field is empty. The expiration date is set to 01 / 2010, with an example of 08/2009. The verification code field is empty, with a link '¿Qué es esto?'. The payment options are 'Cuotas Comercio' (10 cuotas de \$1,001, total \$10,000). At the bottom, there are 'Anular' and 'Pagar' buttons. A security notice at the bottom states: 'Esta transacción se está realizando sobre un sistema seguro.'

Para pago a través de Redcompra, el usuario debe seleccionar el emisor adherido e ingresar los datos solicitados: número de tarjeta y/o RUT.

The screenshot shows the 'darcon tarjeta.cl' website header with the 'web pay plus' logo. Below the header, the 'Red compra' section is active, displaying the 'Red compra' logo. The total amount to be paid is \$5,000 (Cinco mil pesos chilenos). The merchant is 'INTERNET DARCONTARJE'. The issuer is set to 'BANCO SANTANDER'. The card number field is empty. The details section shows 'F.CH.ADOPC,DARCONTAR' for \$5,000 with a green checkmark. At the bottom, there are 'Anular' and 'Pagar' buttons. A security notice at the bottom states: 'Esta transacción se está realizando sobre un sistema seguro.' Below the security notice, there are logos for 'VERIFIED by VISA' and 'MasterCard SecureCode'.

## 7.3 Creación de Páginas de Comercio

El último paso en la integración del KCC con el software del comercio, es la creación de cuatro páginas dinámicas por parte de la tienda para conectarse con el KCC.

### 7.3.1 *Página de Pago*

Por cada transacción Webpay que el comercio quiera ejecutar deberá crear una página dinámica de pago, que haga la llamada al CGI `tbk_bp_pago.cgi` con los parámetros adecuados de acuerdo a la transacción y que se especifican más adelante. Así, esta es la página que tiene el botón pagar y llama al CGI `tbk_bp_pago.cgi` con los parámetros adecuados.

Ejemplo:

Default.asp	
Ubicación	Directorio base
Permiso	Lectura

### 7.3.2 Página de Cierre para confirmar las transacciones



**IMPORTANTE:** La página de cierre, explicada en esta sección, contiene la lógica más relevante de la transacción. De no estar bien implementada, ocasionará errores e inconsistencias entre la información manejada por Transbank y el comercio. Específicamente, se podría producir un descuadre frente a transacciones autorizadas y reversadas por no contar con la respuesta adecuada por parte de la tienda del comercio. El efecto final serán diferencias en abono al comercio.

La página de cierre, es ejecutada por el CGI `tbk_bp_resultado`<sup>2</sup> una vez que Webpay ha resuelto el resultado de la autorización de la transacción. Tiene la finalidad de confirmar la transacción por parte del comercio.

La página de cierre recibirá por método POST, los parámetros que se indican en el punto 7.4.2. Con ellos, esta página deberá ser capaz de realizar las operaciones necesarias (validación ver punto 7.4 y almacenaje de parámetros), desde el punto de vista del comercio, para confirmar la transacción.

La transacción puede tener 2 resultados: autorizada por el emisor o rechazada por el emisor.

Para el caso de transacciones autorizadas por el emisor (`TBK_RESPUESTA = 0`), se debe considerar lo siguiente:

1. Si el comercio acepta la respuesta de la transacción, la salida estándar de esta página debe devolver el String (HTML), “**ACEPTADO**”, en tal caso la CGI `tbk_bp_resultado` se encarga de realizar la confirmación (ACK) a Transbank.
2. Por otro lado si el comercio no acepta la respuesta de la transacción, el String de salida deberá ser distinto a ACEPTADO, por ejemplo “**RECHAZADO**”, en tal caso la transacción es reversada.
3. Se debe tener claro también, que frente a fallas en la página de cierre, inexistencia o no acceso a ella, la transacción se entenderá como no aceptada por el comercio, generándose también una reversa de la autorización.

Para el caso de transacciones rechazadas por el emisor (`TBK_RESPUESTA ≠ 0`), se debe considerar lo siguiente:

1. No existirá impacto financiero, sin embargo para este caso la respuesta por parte de la tienda debe ser siempre “**ACEPTADO**”, ver página 33 de este manual, de esta forma se desplegará la página de fracaso del comercio, de otra forma se desplegará al cliente la página de error de Webpay.

Es importante indicar, que esta página debe registrar en un medio persistente el resultado entregado a Transbank, asociándolo a la Orden de Compra respectiva. Esto ayudará a la tienda a conocer y tener un respaldo de la respuesta entregada, como también detectar problemas propios, por ejemplo múltiples rechazos o no aceptaciones.

---

<sup>2</sup> Dado que sólo Transbank puede ejecutar la cgi `tbk_bp_resultado`, se recomienda que se filtre la ejecución de dicha cgi por las IP's informadas en la sección 5.5



Dado que esta página debe ser capaz de recibir datos por método POST y además ejecutar acciones, en realidad no debe ser una página HTML pura, sino más bien una página dinámica (cgi, php, asp, jsp, etc.), capaz de generar HTML como salida.

Ejemplo:

xt_compra.asp	
Ubicación	Directorio base
Permiso	Lectura y Escritura



**Importante:** Si el sitio Web trabaja con algún tipo de librerías que se incorporen en la página de cierre, las líneas de importación deben ser eliminadas de esta página, ya que la respuesta esperada por la cgi, en caso de éxito, es solo “ACEPTADO”. En caso de incluir las librerías, estas agregarían código adicional a la página, como funciones, estilos, etc. con lo cual se podría no tener el comportamiento esperado de la

tienda.

---

### 7.3.3 Página de Éxito

Se define como la página que indica al cliente del comercio que su transacción fue realizada con éxito, esta página debe ser programada por el comercio con el fin de desplegar información de la transacción, por ejemplo: orden de compra, productos/servicios, montos, etc. Ejemplo:

Exito.asp	
Ubicación	Directorio base
Permiso	Lectura y Escritura

### 7.3.4 Página de Fracaso

Se define cómo la página que indica al cliente del comercio que su transacción no pudo ser realizada, ésta debe ser programada por el comercio. Ejemplo:

Fracaso.asp	
Ubicación	Directorio base
Permiso	Lectura

Observaciones:

- Para Tipo Transacción Mall Virtual la página de éxito se reemplaza por una página de resultado en donde se encuentra una lista con todas las transacciones aceptadas y rechazadas.
- Las páginas de éxito, fracaso y resultado son invocadas mediante el método POST.

## 7.4 VALIDACIONES OBLIGATORIAS EN PÁGINA DE CIERRE

### 7.4.1 Validación de Monto

Para aumentar la seguridad de la tienda virtual, Transbank exige la realización de una validación del monto de cada transacción. Para tal efecto, se entrega a la página de cierre el parámetro TBK\_MONTO ó TBK\_MONTO\_TIENDA\_MXXX de acuerdo a la transacción usada. Así la tienda virtual debe validar el monto que pide autorizar con el monto autorizado por Transbank. En la página de pago se guarda la información.

### 7.4.2 Validación de MAC

Debido a que la página de cierre es la que define finalmente la ejecución de la transacción, es importante asegurar que esta sólo sea invocada desde la CGI tbk\_bp\_resultado instalado en el comercio. Para lo anterior es necesario realizar una validación de la firma digital.

La CGI tbk\_bp\_resultado entrega a la página de cierre un parámetro con el nombre TBK\_MAC que contiene la firma digital de los datos entregados. Para la validación de dicha firma se entrega el programa tbk\_check\_mac el cuál se encarga de verificar esta firma retornando la palabra “CORRECTO” en el caso que sea válida o la palabra “INVALIDO” en caso contrario. En el caso de error, además escribe en la salida estándar la razón del error.

Para chequear los parámetros de respuesta utilizando el programa tbk\_check\_mac, la página de cierre debe generar un archivo de texto con los parámetros recibidos desde tbk\_bp\_resultado, en el mismo formato y orden en que llegan a la tienda, separados por el carácter “&” y entregar al programa como parámetro la ubicación y nombre del archivo. Por ejemplo, en el caso de páginas programadas en ASP, la totalidad de los parámetros se pueden obtener llamando al método “Form” del objeto “Request” sin parámetros (data = Request.Form ()).

El orden de los parámetros recibidos es<sup>3</sup>:

- |                             |                          |
|-----------------------------|--------------------------|
| 1) TBK_ORDEN_COMPRA         | 9) TBK_HORA_TRANSACCION  |
| 2) TBK_TIPO_TRANSACCION     | 10) TBK_ID_SESION        |
| 3) TBK_RESPUESTA            | 11) TBK_ID_TRANSACCION   |
| 4) TBK_MONTO                | 12) TBK_TIPO_PAGO        |
| 5) TBK_CODIGO_AUTORIZACION  | 13) TBK_NUMERO_CUOTAS    |
| 6) TBK_FINAL_NUMERO_TARJETA | 14) TBK_TASA_INTERES_MAX |
| 7) TBK_FECHA_CONTABLE       | 15) TBK_VCI              |
| 8) TBK_FECHA_TRANSACCION    | 16) TBK_MAC              |

---

<sup>3</sup> En caso de que el comercio se encuentre habilitado para recibir el número de la tarjeta, y que además haya solicitado el envío de la fecha de vencimiento Ver Anexo G

Observaciones:

- Para generar el nombre del archivo temporal se requiere usar un nombre distinto para cada transacción ejecutada. Para lo anterior se recomienda utilizar, como parte del nombre del archivo, el valor que viene en el parámetro “TBK\_ID\_TRANSACCIÓN” o “TBK\_ORDEN\_COMPRA”
- Lenguajes como Java no aseguran que el orden de recuperación de los parámetros sea el mismo con el que fueron enviados. Se debe tener en cuenta el orden de recuperación parámetros, en caso de que estos sean recuperados uno a uno por su nombre, ejemplo: `request.getParameter("TBK_ORDEN_COMPRA")` . Si el orden de los parámetros no es el original, la respuesta de `check_mac` siempre será INVALIDO.
- Al guardar los parámetros en archivo temporal, estos deben ser separados por el carácter “&”.

Ejemplo:

`TBK_ORDEN_COMPRA=99221&TBK_TIPO_TRANSACCION=TR_NORMAL&...`

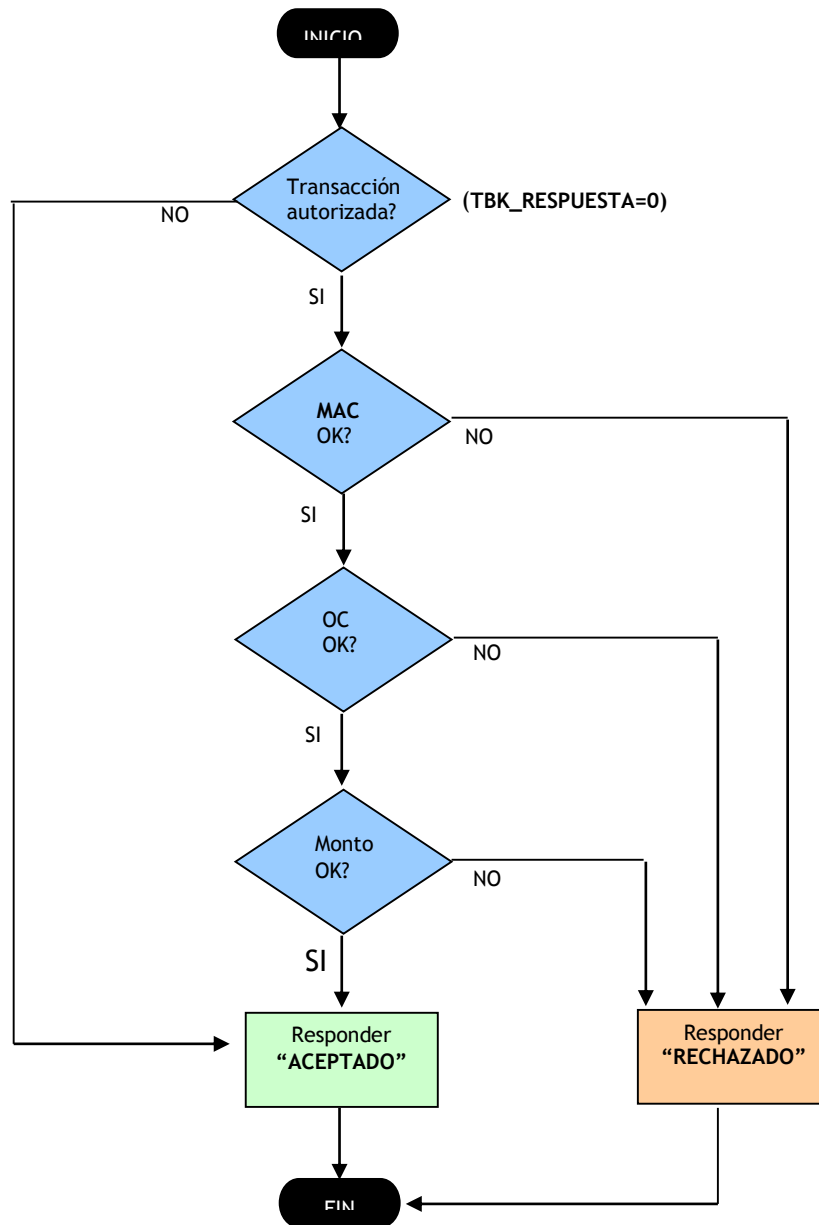
#### **7.4.3 Validación de Duplicidad de Orden de Compra**

Si la tienda permite pagar dos órdenes de compra idénticas esto significaría un doble cargo al tarjetahabiente de una misma compra. De acuerdo a lo anterior se debe contemplar en la tienda un manejo de la duplicidad de órdenes, para ello se debe contemplar los siguientes pasos:

Grabar Orden de Compra en la Base de datos antes de realizar el pago.

Verificar, de acuerdo a la respuesta que envía Transbank a la tienda virtual, que para dicha orden de compra, en la base de datos de la tienda no exista un código de autorización asociado, si existe (significa que ya fue pagada), se envía el String “RECHAZADO” y si no existe se graba el código de autorización a dicha orden de compra y se envía un String “ACEPTADO”.

La lógica implementada en la página de cierre debe ser la siguiente:



## 7.5 GENERACIÓN DE LLAVES ASIMÉTRICAS PRODUCTIVAS

Para generar las llaves asimétricas productivas, se puede utilizar la aplicación proporcionada por Transbank (genkey.exe) o utilizar las librerías OpenSSL:

Generación de llaves con OpenSSL:

```
# openssl genrsa -out privada.pem 4096  
# openssl rsa -in privada.pem -pubout -out publica.pem
```

Generación de llaves con herramienta genkey (sólo plataforma Windows):

Se debe utilizar **#genkey <parámetro llave>**

El parámetro **<parámetro llave>** tiene el siguiente valor:

4 = 4096

Si no se agrega el parámetro, genera por default una llave de 4096.

```
#genkey ↵  
Generacion de LLaves para KCC 6.0.2 Webpay Transbank  
Ingrese su codigo de comercio:59701234556  
Tamano Llave: 4096  
generando llaves....  
Clave privada RSA guardada en archivo privada.pem  
  
El archivo privada.pem contiene su llave privada, este es secreto y  
debe ser cargado en el KCC.
```

Las llaves que se encuentran por defecto en el kit son las llaves de ambiente de desarrollo. Para realizar la puesta en producción, debes cargar a la carpeta “MAESTROS” la llave “privada.pem” y modificar el nombre “tbk\_public\_key101prod.pem”, por el de “tbk\_public\_key.pem”

Las llaves debes ser cargadas al servidor web de manera ASCII, para evitar que sea adulterado.

## 8. DESCRIPCIÓN DE TIPOS DE TRANSACCIONES

Para el buen funcionamiento del sistema de pago Webpay, se deben definir para cada una de las transacciones de pago, parámetros de entrada y salida. Los parámetros de entrada son los que se deben generar en las páginas de llamada al KCC por parte del comercio, básicamente a través de algún formulario que envíe dichos parámetros vía método POST al CGI tbk\_bp\_pago.

Los parámetros de salida son aquellos que entrega Transbank a través del CGI tbk\_bp\_resultado al comercio por medio de la página de cierre para finalizar la transacción.

La definición de parámetros se encuentra establecida para cada tipo de transacción. Esta definición se encuentra en el archivo tbk\_param.txt, la cual no es necesario crear o modificar, ya se entrega en el KCC.

Ejemplo para Transacción normal:

```
<TR_NORMAL>
  TBK_TIPO_TRANSACCION#A#50#1
  TBK_MONTO#M#10#1
  TBK_ORDEN_COMPRA#A#26#1
  TBK_ID_SESION#A#61#0
  TBK_URL_FRACASO#A#256#1
  TBK_URL_EXITO#A#256#1
  TBK_MONTO_CUOTA#M#9#0
  TBK_NUMERO_CUOTAS#N#2#0
</TR_NORMAL>
```

Lectura:

TBK\_TIPO\_TRANSACCION#A#50#1 indica que el parámetro TBK\_TIPO\_TRANSACCION es de tipo alfanumérico, de tamaño máximo 50 y que es obligatorio.

#A : Alfanumérico  
#N : Numérico entero  
#M: Numérico con decimales  
#0 : Opcional  
#1 : Obligatorio

Observación: Los tipos de transacciones soportadas por las tiendas están definidos implícitamente con cada código de comercio, lo cual se establece en el contrato comercial.

## 8.1 TRANSACCIÓN NORMAL

La Transacción Normal consiste en realizar una transacción virtual llamando a la CGI `tbk_bp_pago` con los parámetros de entrada definidos en la Tabla 3. Una vez ejecutado este cgi el control de la transacción es tomado por Transbank quien le exige al tarjetahabiente que entregue una serie de datos de su tarjeta para terminar la transacción, el resultado de ésta es comunicado al comercio por medio del cgi `tbk_bp_resultado`, que entrega los parámetros a la página de cierre, estos parámetros de salida son mostrados en la Tabla 4. Una vez confirmada la transacción a Transbank por parte del comercio, se le devuelve a éste el control del browser del cliente. Esta Transacción está orientada a una tienda en donde por cada pago el cliente debe digitar los datos de la tarjeta de crédito.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_NORMAL)	50	A
TBK_MONTO	Monto de la transacción. Sin separadores de miles ni de decimales. Se asume los dos últimos dígitos como valores decimales.	10	N
TBK_ORDEN_COMPRA	Orden de compra, generado por el comercio.	26	A
TBK_ID_SESION	Identificador de sesión. (opcional)	61	A
TBK_URL_EXITO	Página a llamar en caso de éxito en la transacción. Debe ser una dirección absoluta, para lograr el redireccionamiento del <i>browser</i> del cliente.	256	A
TBK_URL_FRACASO	Página a llamar en caso de fracaso en la transacción. Debe ser una dirección absoluta, para lograr el redireccionamiento del <i>browser</i> del cliente.	256	A
TBK_MONTO_CUOTA	Se envía para caso de pago SI, S2, CI y NC, Monto correspondiente de cada cuota.	9	N
TBK_NUMERO_CUOTAS	Sólo se envía para caso de pago CI. (opcional) Número de cuotas a pagar.	2	N

Tabla 3. Parámetros de entrada Transacción Normal

Parámetro	Descripción	Largo	Tipo
TBK_ACCION	Acción (ACK, ERR)	3	A
TBK_ORDEN_COMPRA	Orden de compra.	26	A
TBK_CODIGO_COMERCIO	Código de comercio	12	N
TBK_CODIGO_COMERCIO_ENC	Código de comercio encriptado		A
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_NORMAL en este caso).	50	A
TBK_RESPUESTA	Respuesta a la solicitud de la transacción.	2	N
TBK_MONTO	Monto de la transacción. Sin separadores de miles ni de decimales. Se asume los dos últimos dígitos como valores decimales.	10	N
TBK_CODIGO_AUTORIZACION	Código de autorización de la transacción de compra, Es mínimo 6 y un máximo de 8 dígitos.	8	A
TBK_FINAL_NUMERO_TARJETA	Cuatro últimos números de la Tarjeta de Crédito con que se realizó la compra. (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_FECHA_CONTABLE	Fecha contable de la transacción. (mmdd).	4	N
TBK_FECHA_TRANSACCION	Fecha en que se realizó la transacción. (mmdd).	4	N
TBK_FECHA_EXPIRACION	Fecha de expiración de Tarjeta de Crédito con que se realizó la compra (aamm). (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_HORA_TRANSACCION	Hora en la que se realizó la transacción. (hhmmss).	6	N
TBK_ID_SESION	Identificador de sesión. (Si fue habilitado).	61	A
TBK_ID_TRANSACCION	Identificador de transacción	20	N
TBK_TIPO_PAGO	Forma de Pago escogida por el tarjetahabiente.	2	A
TBK_NUMERO_CUOTAS	En el caso de corresponder a un tipo de pago CI, tiene el mismo valor que el parámetro de entrada. Para los otros tipos de pago (VC y SI), contiene el número de Cuotas escogida por el tarjetahabiente.	2	N
TBK_VCI	Resultado de transacción	3	A
TBK_MAC	Firma digital. (llave de 4096 id = 101)	4096	A

Tabla 4. Parámetros de salida Transacción Normal.

En el Anexo F se describe un ejemplo paso a paso de integración de KCC 6.0.2 con transacción Normal.

La correspondencia entre TBK\_TIPO\_PAGO y TBK\_NUMERO\_CUOTAS (el máximo de cuotas depende de cada emisor) el detalle es el siguiente :

DESCRIPCIÓN TIPO PAGO	TBK_TIPO_PAGO	TBK_NUMERO_CUOTAS
Venta Normal	VN	0
Venta Cuotas (Sólo cuotas normales)	VC	2 - 48
Venta Cuotas (Con 3 cuotas sin interés activado)	VC	4 - 48
Tres Cuotas Sin Interés	SI	3
Dos Cuotas Sin Interés	S2	2
Cuotas Comercio	CI	6 - 48
Cuotas sin interés (N cuotas)	NC	2 - 10
Redcompra	VD	0



## 8.2 TRANSACCIÓN MALL VIRTUAL

La Transacción Mall Virtual permite agrupar a varias tiendas virtuales en un Mall Virtual, el cual posee un código de comercio propio y único que corresponde al parámetro ID\_COMERCIO que se configura en el archivo tbk\_config.dat. Por otro lado, cada tienda virtual asociada al Mall también posee un código de comercio (código de tienda), que la identifica como única, este código se envía como parámetro al momento de realizar la compra.

Esta transacción actúa básicamente igual que la Transacción Normal con la salvedad que al momento de la confirmación por parte de Transbank, la derivación del browser es hacia una página de resultado en la cual se lleva un catastro de las transacciones por comercio aprobadas y rechazadas. En caso de existir un error en la confirmación por parte de la tienda (Mall), a Transbank en cuanto a la realización del cierre de la compra el browser del cliente será derivado a la página de fracaso definida por el comercio.

Los parámetros de entrada al cgi tbk\_bp\_pago se definen en la Tabla 5 y los de salida en la Tabla 6.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_MALL).	50	A
TBK_MONTO	Monto total de la transacción Mall.	10	N
TBK_ORDEN_COMPRA	Orden de compra de la transacción Mall.	26	A
TBK_ID_SESION	Identificador de sesión (opcional).	61	A
TBK_URL_RESULTADO	Página a llamar con los resultados de las transacciones.	256	A
TBK_URL_FRACASO	Página a llamar en caso de fracaso en la transacción de Mall.	256	A
TBK_NUM_TRX	Número de transacciones de compra.	4	N
TBK_CODIGO_TIENDA_MXXX	Código de Comercio de la tienda número XXX en la transacción Mall.	12	N
TBK_ORDEN_TIENDA_MXXX	Orden de compra de la tienda XXX en la transacción Mall.	26	A
TBK_MONTO_TIENDA_MXXX	Monto de compra de la tienda XXX en la transacción Mall.	10	N
TBK_MONTO_CUOTA_MXXX	Se envía para caso de pago SI, S2, CI y NC, Monto correspondiente de cada cuota para tienda XXX.	9	N
TBK_NUMERO_CUOTAS_MXXX	Sólo se envía para caso de pago CI. Número de cuotas a pagar para tienda XXX.	2	N

Tabla 5 Parámetros de entrada de la Transacción Mall Virtual.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_MALL).	50	A
TBK_RESPUESTA	Siempre viene en 0.	2	N
TBK_ORDEN_COMPRA	Orden de compra Mall.	26	A
TBK_ID_SESION	Identificador de sesión (si fue habilitado).	61	A
TBK_MONTO	Monto de la transacción Mall.	10	N
TBK_FINAL_NUMERO_TARJETA	Cuatro últimos números de la Tarjeta de Crédito con que se realizó la compra. (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_FECHA_EXPIRACION	Fecha de expiración de Tarjeta de Crédito con que se realizó la compra (aamm). (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_FECHA_CONTABLE	Fecha contable de la transacción.	4	N
TBK_FECHA_TRANSACCION	Fecha en que se realizó la transacción.	8	N
TBK_HORA_TRANSACCION	Hora en la que se realizó la transacción.	6	N
TBK_ID_TRANSACCION	Identificador de transacción.	20	N
TBK_MAC	Firma digital. (llave de 4096 id = 101)	4096	A
TBK_TASA_INTERES_MAX	Tasa de interés máxima permitida para el período vigente.	4	N
<b>Parámetros de salida por cada tienda del Mall</b>			
TBK_COD_AUT_MXXX	Código de Autorización número XXX.	6	N
TBK_TIPO_PAGO_MXXX	Forma de pago número XXX.	2	A
TBK_NUMERO_CUOTAS_MXXX	Número de cuotas de la compra en tienda número XXX.	2	N
TBK_MONTO_CUOTA_MXXX	Se envía para caso de pago SI, S2, CI y NC, Monto correspondiente de cada cuota para tienda XXX.	9	N
TBK_COD_RESP_MXXX	Respuesta a la transacción número XXX.	2	N
TBK_MONTO_TIENDA_MXXX	Monto en la tienda número XXX.	26	A
TBK_ORDEN_TIENDA_MXXX	Orden de compra para la tienda XXX.	10	N

Tabla 6 Parámetros de salida Transacción Mall Virtual.

**Observación:**

XXX varía de 001 a 020, es decir cómo máximo se pueden enviar 20 pagos distintos en una única transacción. Los pagos pueden pertenecer a la misma tienda o a tiendas diferentes, identificando cada pago con su propia orden de compra.



**IMPORTANTE:**

Para validar el pago exitoso en la página de cierre y en la página de resultado del mall se debe considerar el parámetro TBK\_COD\_RESP\_MXXX de cada tienda y por ningún motivo utilizar el parámetro TBK\_RESPUESTA del mall ya que este siempre viene con valor 0.

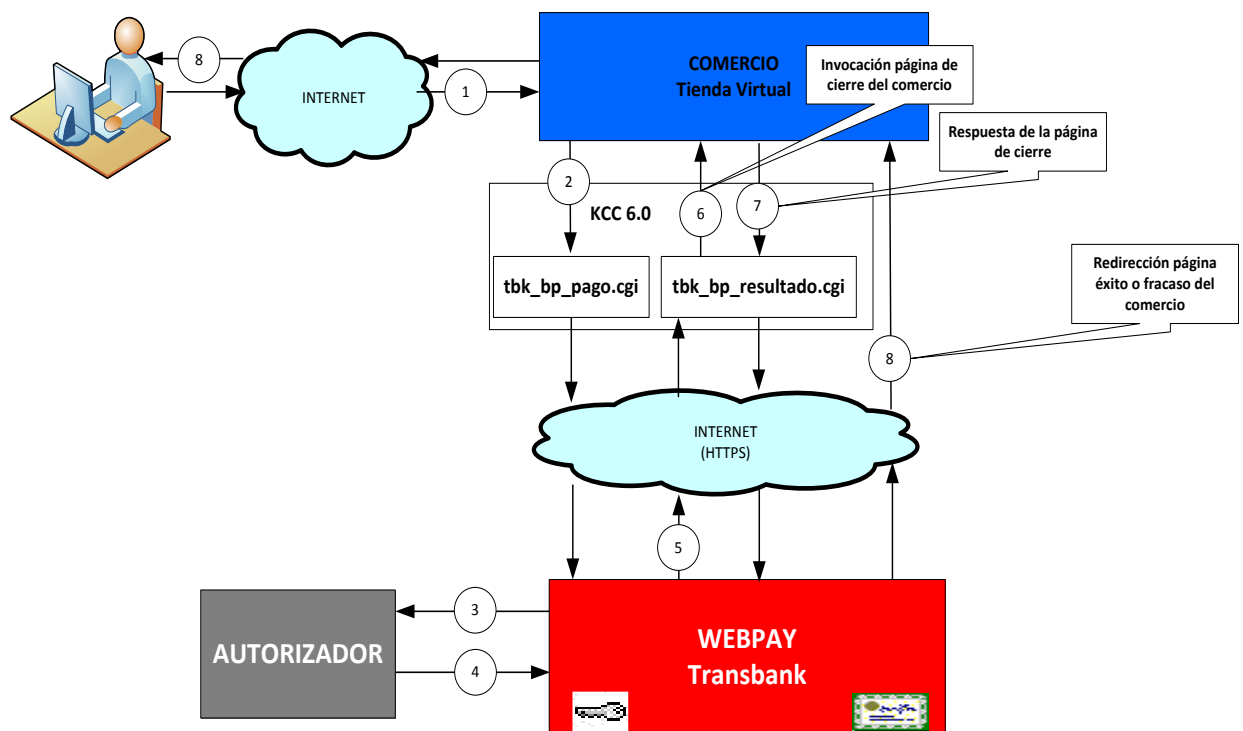
### 8.3 TRANSACCIÓN NORMAL COMPLETA

La Transacción Normal Completa consiste en realizar una transacción virtual llamando a la CGI `tbk_bp_pago` con los parámetros de entrada definidos en la Tabla 7. Una vez ejecutado este cgi, el control de la transacción por Webpay, quien autoriza la transacción para comunicarle el resultado al comercio por medio del cgi `tbk_bp_resultado`, la cual envía a la página de cierre (comercio) los parámetros indicados en la Tabla 8. Una vez que ésta página confirma la transacción, la respuesta es recibida por Transbank quien registra el resultado y devuelve el control al navegador del cliente. Esta Transacción está orientada a una tienda que tiene en su poder los datos de la tarjeta de crédito del tarjetahabiente o tiene al tarjetahabiente a su lado para que ingrese sus datos en el momento.

**Es importante señalar que en esta transacción no existe el proceso de autenticación con el banco emisor.**

**El comercio para poder contar con este producto debe contar con la certificación PCI (Security Standards Council).**

A continuación se muestra un esquema de operación:



Flujo transaccional Transacción Completa

**Paso 1:**

Tarjeta habiente en sitio web del comercio inicia un pago con Webpay. (Por ejemplo presionando sobre el botón “pagar con Webpay”).

**Paso 2:**

Sitio web del comercio invoca al cgi `tbk_bp_pago.cgi` mediante método POST enviando los parámetros de entrada de la tabla número 7. Webpay recibe los datos de la transacción.

**Paso 3:**

Webpay con los datos de la transacción la envía a autorizar.

**Paso 4:**

Autorizador envía respuesta a Webpay del proceso de autorización.

**Paso 5:**

Webpay invoca a CGI `tbk_bp_resultado.ci` alojada en el servidor Web del comercio enviado la respuesta encriptada.

**Paso 6:**

La CGI `tbk_bp_resultado.cgi` desencripta la respuesta enviada por Webpay y envía los datos de respuesta (Tabla 8) a la página de cierre del comercio. En dicha página el comercio debe realizar las validaciones exigidas en el punto 7.4 del presente documento.

**Paso 7:**

Sitio web del comercio contesta “ACEPTADO” o “RECHAZADO” desde la página de cierre. La CGI `tbk_bp_resultado.cgi` envía dicha respuesta a Webpay.

**Paso 8:**

Según corresponda Webpay redirecciona al Tarjeta Habiente a la página de éxito o de fracaso del sitio web del comercio.

Parámetros de entrada de la Transacción Normal Completa.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_COMPLETA)	50	A
TBK_MONTO	Monto total de la transacción Normal Completa.	10	N
TBK_ORDEN_COMPRA	Orden de compra de la transacción Normal Completa.	26	A
TBK_ID_SESION	Identificador de sesión (opcional).	61	A
TBK_URL_EXITO	Página a llamar en caso de éxito en la transacción. Debe ser una dirección absoluta, para lograr el redireccionamiento del <i>browser</i> del cliente.	256	A
TBK_URL_FRACASO	Página a llamar en caso de fracaso en la transacción. Debe ser una dirección absoluta, para lograr el redireccionamiento del <i>browser</i> del cliente.	256	A
TBK_NUMERO_TARJETA	Número completo de la tarjeta de crédito con el que se realizó la compra.	19	N
TBK_FECHA_EXPIRACION	Fecha de expiración de la tarjeta de crédito con que se realizó la compra (aamm).	4	N
TBK_CVV	Código de verificación de la tarjeta de crédito con que se realizó la compra.	4	N
TBK_TIPO_PAGO	Forma de Pago escogida por el tarjetahabiente.	2	A
TBK_NUMERO_CUOTAS	Sólo se envía para caso de pago CI. Número de cuotas a pagar (opcional).	2	N
TBK_MONTO_CUOTA	Se envía para caso de pago SI, S2, CI y NC, Monto correspondiente de cada cuota.	9	M

Tabla 7 Parámetros de entrada de la Transacción Normal Completa.

Parámetros de salida de la Transacción Normal Completa.

Parámetro	Descripción	Largo	Tipo
TBK_ORDEN_COMPRA	Orden de compra.	26	A
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_COMPLETA en este caso).	50	A
TBK_RESPUESTA	Respuesta a la solicitud de la transacción.	2	N
TBK_MONTO	Monto de la transacción. Sin separadores de miles ni de decimales. Se asume los dos últimos dígitos como valores decimales.	10	N
TBK_CODIGO_AUTORIZACION	Código de autorización de la transacción de compra, Es mínimo 6 y un máximo de 8 dígitos	6	A
TBK_FINAL_NUMERO_TARJETA	Cuatro últimos números de la Tarjeta de Crédito con que se realizó la compra. (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_FECHA_CONTABLE	Fecha contable de la transacción. (mmdd).	4	N
TBK_FECHA_TRANSACCIÓN	Fecha en que se realizó la transacción. (mmdd).	4	N
TBK_FECHA_EXPIRACION	Fecha de expiración de Tarjeta de Crédito con que se realizó la compra (aamm). (Sólo para los comercios que Transbank habilite esta opción).	4	N
TBK_HORA_TRANSACCIÓN	Hora en la que se realizó la transacción. (hhmmss).	6	N
TBK_ID_SESION	Identificador de sesión. (Si fue habilitado).	61	A
TBK_ID_TRANSACCIÓN	Identificador de transacción	20	N
TBK_TIPO_PAGO	Forma de Pago escogida por el tarjetahabiente.	2	A
TBK_NUMERO_CUOTAS	En el caso de corresponder a un tipo de pago CI, tiene el mismo valor que el parámetro de entrada. Para los otros tipos de pago (VC y SI), contiene el número de Cuotas escogida por el tarjetahabiente.	2	N
TBK_MAC	Firma digital. (llave de 4096 id = 101)	1024	A

Tabla 8 Parámetros de salida de la Transacción Normal Completa.

## 8.5 TRANSACCIÓN CONSULTA DE TASA

La totalidad de los parámetros de entrada a la CGI tbk\_bp\_pago se muestran en la Tabla 11.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_TASA_INTERES_MAX).	50	A
TBK_URL_EXITO	Página a llamar en caso de éxito en la transacción	256	A
TBK_URL_FRACASO	Página a llamar en caso de fracaso en la transacción.	256	A

Tabla 11 Parámetros de entrada de la Transacción Consulta de Tasa.

Los parámetros de salida de la Transacción Consulta De Tasa se muestran la Tabla 12.

Parámetro	Descripción	Largo	Tipo
TBK_TIPO_TRANSACCION	Tipo de transacción (TR_TASA_INTERES_MAX).	50	A
TBK_RESPUESTA	Respuesta a la solicitud de la transacción.	2	N
TBK_FECHA_TRANSACCIÓN	Fecha en que se realizó la transacción.	8	N
TBK_HORA_TRANSACCIÓN	Hora en la que se realizó la transacción	6	N
TBK_ID_TRANSACCIÓN	Identificador de transacción.	20	N
TBK_TASA_INTERES_MAX	Tasa de interés máxima permitida del periodo.	4	N
TBK_MAC	Firma digital. (llave de 4096 id = 101)	4096	A

Tabla 12 Parámetros de salida de la Transacción Consulta de Tasa.

## 9. ARCHIVOS GENERADOS POR EL KCC

El KCC genera dos tipos de archivos logs, un log de eventos y un log de transacciones, ambos archivos de logs se generan en el directorio \$(PATH\_DIRCGI\_WEB)/log con formato texto plano.

### 9.1 Archivo de Log de Eventos

En este archivo se escriben todos los logs informativos, de error o fatales según lo configurado en el archivo tbk\_trace.dat. Se genera un archivo distinto por cada día y el nombre del archivo es de la forma TBK\_EVN\$(FECHA).log.

Donde:

\$(FECHA): corresponde a la fecha en formato aaaammdd.

El formato de este archivo es el que se muestra a continuación:

Variable	Descripción	Largo	Tipo
ID_TRANSACCION	Identificación de la transacción	10	A
PARAMETROS	Datos relevantes, Ejemplo: acción realizada, estado de acción o sobre quién aplica, etc.	40	A
FECHA	Formato ddmmaaaa.	14	N
HORA	Formato hhmmss.	6	N
IP	IP desde donde se realiza la compra.	15	A
RESULTADO	Resultado operación, donde : OK = evento correcto KO = evento incorrecto 0 ERR	3	A
DESCRIPCIÓN	Descripción evento o error.	50	A

Tabla 13 Formato log de eventos.



## Ejemplo:

	68 Filtro	Inicio	29082008	102150	200.10.12.227	OK	Inicio de filtrado
	68 Filtro	tbk_param.txt	29082008	102150	200.10.12.227	OK	Archivo parseado
	68 Filtro	Terminado	29082008	102150	200.10.12.227	OK	Datos Filtrados con exito
	68 sanitiza	inicio	29082008	102150	200.10.12.227	OK	Entrando a sanitizacion
	68 sanitiza	Array	29082008	102150	200.10.12.227	OK	Arreglo de parametros OK.
	68 sanitiza	Fin	29082008	102150	200.10.12.227	OK	Sanitizacion exitosa
19710175	68 pago	inicio	29082008	102150	200.10.12.227	OK	5,97028E+11 Parseo realizado
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Datos en datos/tbk_config.dat
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Mac generado
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Construcccion TBK_PARAM
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Codigo comercio encriptado
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 TBK_PARAM encriptado
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Datos listos para ser enviados
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Medio 2: Por redireccion
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Redireccion web
19710175	68 pago	https://172.24.15.14	29082008	102150	200.10.12.227	OK	5,97028E+11 Todo OK
	228 resultado	Inicio	29082008	102257	172.24.15.14	OK	correctamente recibidos
	228 resultado	Desencriptando	29082008	102257	172.24.15.14	OK	Codigo comercio desencriptado
	228 resultado	Desencriptando	29082008	102257	172.24.15.14	OK	TBK_PARAM desencriptado
	228 resultado	Validacion	29082008	102257	172.24.15.14	OK	Entidad emisora de los datos validada
	228 resultado	OC_20088292148	29082008	102257	172.24.15.14	OK	Parseo de los datos
	228 resultado	OC_20088292148	29082008	102257	172.24.15.14	OK	http://198.22.250.29/qa/kcc51_qa/xt_compra.asp
19710175	228 transacc	19710175	29082008	102257	172.24.15.14	OK	5,97028E+11 conectandose al port :(80)
19710175	228 resultado	logro abrir_conexion	29082008	102257	172.24.15.14	0	5,97028E+11 Abrio socket para conex-com
19710175	228 transacc	19710175	29082008	102257	172.24.15.14	OK	5,97028E+11 POST a url http://198.22.250.29/qa/kcc51_qa/xt_compra.asp
19710175	228 transacc	19710175	29082008	102257	172.24.15.14	OK	5,97028E+11 mensaje enviado
19710175	228 transacc	19710175	29082008	102257	172.24.15.14	OK	5,97028E+11 Llego ACK del Comercio
19710175	228 resultado	OC_20088292148	29082008	102257	172.24.15.14	OK	5,97028E+11 tienda acepto transaccion
19710175	228 resultado	OC_20088292148	29082008	102257	172.24.15.14	OK	5,97028E+11 respuesta enviada a TBK (ACK)
19710175	228 resultado	OC_20088292148	29082008	102257	172.24.15.14	OK	5,97028E+11 Todo OK

Observación: Todas las variables están ordenadas en una sola línea, así se genera una línea por actividad de transacción.

## 9.2 Archivo de Log de Transacciones (bitácora)

Este archivo contiene una línea con cada transacción ejecutada y almacena la data que entrega la CGI tbk\_bp\_resultado a la página de confirmación de la transacción.

El nombre de este archivo es TBK\_\$(TIPO\_TRANSACCION)\_\$(FECHA).log.

Donde:

\$(TIPO\_TRANSACCION) = TR\_NORMAL, TR\_MALL, etc.

\$(FECHA) = mmdd

El formato de este archivo es el que se muestra a continuación en la Tabla 14.

Variable	Descripción	Largo	Tipo
TBK_ACCION *	Acción (ACK, ERR)	3	A
TBK_ORDEN_COMPRA	Número de la orden de compra generada por el comercio (número único por transacción).	26	A
TBK_CODIGO_COMERCIO	Código de comercio, número que identifica en forma unívoca al comercio.	12	N
TBK_TIPO_TRANSACCION	Indica el tipo de transacción utilizada.	50	A
TBK_RESPUESTA	Respuesta entregada por Transbank.	2	N
TBK_MONTO	Monto de la transacción entregado por el comercio.	10	N
TBK_CODIGO_AUTORIZACION	Código de autorización entregado por Transbank. Solo tienen código las transacciones aprobadas "R=0".	8	A
TBK_FINAL_NUMERO_TARJETA	Campos correspondientes a datos de la tarjeta que no son entregados al comercio Internet "sin envío de tarjeta 00000".	17	N
TBK_FECHA_CONTABLE	Fecha (mmdd).	4	N
TBK_FECHA_TRANSACCION	Fecha (mmdd), de la transacción.	4	N
TBK_HORA_TRANSACCION	Hora (hhmmss), de la transacción.	6	N
TBK_ID_SESION	Identificador de la sesión.	61	A
TBK_ID_TRANSACCION	Identificador de la transacción.	20	N
TBK_TIPO_PAGO	Tipo de pago utilizado.	2	A
TBK_NUMERO_CUOTAS	Número de cuotas.	1	N
TBK_TASA_INTERES_MAX **	Sólo CI. Tasa de interés máxima permitida del periodo.	4	N
TBK_MONTO_CUOTA **	Se envía para caso de pago SI, S2, CI y NC, Monto correspondiente de cada cuota.	9	N
TBK_MAC	Firma digital del mensaje (conjunto de parámetros) enviados por Transbank.	4096	A

Tabla 14 Formato log de transacciones

\* Se imprime solo el valor.

\*\* Solo si compra es en cuotas.

Ejemplo de una transacción en bitácora:

```
ACK; TBK_ORDEN_COMPRA=221353; TBK_CODIGO_COMERCIO=597027342714;  
TBK_TIPO_TRANSACCION=TR_NORMAL; TBK_RESPUESTA=-1; TBK_MONTO=565200;  
TBK_CODIGO_AUTORIZACION=000000; TBK_FINAL_NUMERO_TARJETA=7276;  
TBK_FECHA_CONTABLE=0520; TBK_FECHA_TRANSACCION=0520;  
TBK_HORA_TRANSACCION=124144; TBK_ID_SESION=238831;  
TBK_ID_TRANSACCION=129613424593; TBK_TIPO_PAGO=VN; TBK_NUMERO_CUOTAS=0;  
TBK_MAC=76d37677633e4a095669d512475ad6473e43d1039b9b599ea1583cd8ab52017e3cab320  
5d1b7b889b8a0fc2a0b9495764473c9d13f54e4ef54044b296ccd8534e19bf5f0332a0db3c8217f3a3  
c685c871590985585b14e58c45d68b6be56231b48425844c20da8f105b6e79d1db2b4ee86c68d588f  
0479c30e5f46e6634957347482f899ae57c3259d84c1827dd58e051dabcb8bbd6be915c40f6dbe8a7  
d01ae9f293e05b0db073eaa039796c540e38a0918f9e78a0633af18d9953b6ce96f4cd54f7e776bc1a  
79ae987fd34873c8fdb98a29d5d39d0f74eca41d73524e709414714bde16ca1f09e7f9e15ab36cbeb  
6347de8723593833059041558169fc71db9801f6e3732611ad275d14cdcab2837808d0757d17b21ab  
609d2f52d63a10dfd7d257c6f833a020918d06da38c47d5424bebf93d352c3ae4f5cd4e67afe24de9  
32bb5485ab57e986f3e3cbe2f6b7e8ed2281f7eed52df7ec75f65dd237e71ce1da96b7b85c1860df  
8f57c053f8481400a252754f3579a1c7f50b2c8d494810e1671cb7618f9ecb04e426a74c3c68dbcc04  
64f2434dd9f636df770a66298001205539633d746691dd12be7e61d27ddc922442e1c52377e48349e
```

Observación: Todas las variables están ordenadas en una sola línea, así se genera una línea por actividad de transacción.

### 9.3 Códigos De Respuesta

Los códigos de respuesta que entrega Transbank a la tienda virtual se definen en la Tabla 11.

TBK_RESPUESTA	Definición
0	Transacción aprobada.
-1	Rechazo de transacción.
-2	Transacción debe reintentarse.
-3	Error en transacción.
-4	Rechazo de transacción.
-5	Rechazo por error de tasa.
-6	Excede cupo máximo mensual.
-7	Excede límite diario por transacción.
-8	Rubro no autorizado.

Tabla 15. Definición de Códigos de Respuesta.

## 10. SOPORTE OPERATIVO

En caso de requerir soporte durante el proceso de incorporación a Webpay Plus o por problemas que eventualmente se pudieran presentar en régimen productivo, deberá enviar un correo a [soporte@transbank.cl](mailto:soporte@transbank.cl), entregando la siguiente información sobre el problema:

- Ambiente en que ocurrió el problema (Integración - producción)
- Código del comercio (5970...)
- Fecha y hora de ocurrencia
- Anexar al correo:
  - Log de eventos y de bitácora correspondientes a la fecha del suceso.
  - Archivo tbk\_config.dat
  - Navegador y sistema operativo del usuario al que se le presentó el problema.
  - Identificar, en lo posible, el emisor de la tarjeta del cliente (aplica sólo a ambiente productivo)
  - Hardware utilizado por el usuario, ejemplo: Pc, Smartphone, Tablet,
  - Informar, en lo posible, la estructura utilizada, ejemplo: Cantidad de servidores, Balanceador de carga, nodos, versión del IOS, nombre del hosting y saber si es a través de enlace nacional o internacional.

Para soporte técnico en línea comunicarse vía telefónica al 800441144, este servicio está disponible los 7 días de la semana las 24 hrs del día.

## ANEXO A

### Datos Ambiente de Integración de Webpay

Datos del Servidor de Prueba:

URL Servidor de Pago	<a href="https://webpay3gint.transbank.cl">https://webpay3gint.transbank.cl</a>
Puerto Servidor de Pago	443

Comercio normales disponibles

Tipo Tienda	Código
Normal Pesos	597020000501
Normal Dólar	597020000510
Normal Pesos Cuotas	597020000503
Normal completa pesos	597020000507
Normal Pesos sin autenticación	597020000541

Comercios mall disponibles

Tipo tienda	Código	Comercio asociado	Código
Mall Pesos	597020000504	Tienda 1 Mall Pesos	597020000505
		Tienda 2 Mall Pesos	597020000506
Mall Dólar	597020000512	Tienda 1 Mall dólar	597020000513
		Tienda 2 Mall dólar	597020000514
Mall Pesos Cuotas	597020000542	Tienda 1 Mall Pesos Cuotas	597020000543
		Tienda 2 Mall Pesos Cuotas	597020000544
Comercios Cert 20 Tiendas	597020000520	20Tiendas - T01	597020000521
		20Tiendas - T02	597020000522
		20Tiendas - T03	597020000523
		20Tiendas - T04	597020000524
		20Tiendas - T05	597020000525
		20Tiendas - T06	597020000526
		20Tiendas - T07	597020000527
		20Tiendas - T08	597020000528
		20Tiendas - T09	597020000529
		20Tiendas - T10	597020000530
		20Tiendas - T11	597020000531
		20Tiendas - T12	597020000532
		20Tiendas - T13	597020000533
		20Tiendas - T14	597020000534
		20Tiendas - T15	597020000535
		20Tiendas - T16	597020000536
		20Tiendas - T17	597020000537
		20Tiendas - T18	597020000538
		20Tiendas - T19	597020000539
		20Tiendas - T20	597020000540

Observación: Cada comercio debe tener asociada un par de llaves asimétricas para encriptación de la información transmitida desde la tienda (KCC) y Transbank. Para efectos del ambiente de integración se indica sólo la llave privada para configuración de las pruebas (La llave indicada es la misma para todos los comercios).



## Llave privada de comercios de prueba (privada.pem)

```
-----BEGIN RSA PRIVATE KEY-----
MIIeowIBAAKCAQEA3HzPC1ZBzC03edUCf/XJiWj3bzJpjjTi/zB090+DDzZCaMp
14aspxQryvJhv8644E19Q+NHfxtz1cxd2wnSYKvay1gJx30ZlTOAkzUj4QMimR16
vomLlQ3T2MAz1znt/PVPVU7T/JOg9R+EbiHNVKa/hUjwJEFVXLQNME97nHoLjb3v
V5yV2aVhmox7b54n6F3UVPVHCsHKbJpXpE+vnLpVmdETbNpFVrDyGxyG+mnEvyiO
BLIwEY3XTMrgXvS069groLi5Gg8C5LDaYOWjE9084T4fiWGrHhn2781R1rykunTu
77wiWPuQHMS0+YC7mhnsk8Z/ilD+aWz/vhsgHwIDAQABAoIBAQCm+Nrt4cpNKQmn
+Ne8348CGRS9ACXp6WRg60CQX04zM7lRZAminVgZgSQXE6aJR+T9rIwMeG7GwydX
aJGzEEQJZ0jV0MkUr+7mk9qiTOGkGHmGlyHnRQU8jDU59vXe3UEl3l5+NmwHbQht
waf9F7XLmoLK/WoVJA6tICRpCl1oQrpziqN+gjdMpz9i8I1sMFE7+Y7xf+7S2u7
c1MRPUWqgdS9yViQVh3vZi25m5CyKRvN0B0hpNuZ7nrJymtADYSWt9wV2W1fX+MX
UUoYfxyQQvWryHhGdedU7GGAnoEdblUcDkBuAaFmsm1P8K4HQZLWP4v6pYlW2JLa
Zoaerb3BAoGBANCRev10CLB0HBU7sCs0eN9fTkIEsh30VIxPSBqDnKsynJrIWovK
cs37Vb6phzdQ03ADoFJvR9ck8+v6Cv0KR8IOF19wfC4ZoxkKBBEq94ZLN+YhE2PW
KiRFybqcgCtzxKS3MyWgpIcT9xFtHVjlorZ8Jk51fgLZbGzamtLhderVAoGBAM00
mIiiv4l2vXzu4tFfkpu/G0x/D9/vAic3X9F0ky09BNCyuMXMQgI8e3wWsGEZghls
Vg9KDV5EPxAmpumcdPFK2IMACaH41ac7vys3ZD8kMK0INQkuDAcG4YsxMaTwEPo0
p1i3zwwEWwknw1yJkOyozz0EcIzS9NrZZEjnBHEjAoGAQ81XdeqzvHEyg/CQd6sq
NCtubGXMZYyi1C4d2Yi5kKn2YRcK4HDi23V+TwodK+0oNWToZIQKjbVUmn0Bv3rt
EvezbDlMFUx+SfCIng0VRJIFTQmpnQYNUxdg2gpwXC/ZWfa6CNxtQABMjFy1cqXM
PJild1IYseJurgBu3mkvBTUCgYBqA/T1X2woLUis2wPIBAv5juXDh3lkB6eU8uxX
CEe2I+3t2EM781B2wajrKadWkmjluMhN9AGV5UZ8S1P0DStUYwUywdx1/8RnmZIP
qSwHAGXV9jI0zNr7G4Em0/leriWkRM26w6fHjLx8EyxDfsohSbkqBrOptcWqoEUx
MOQ5HQKBgAS4sbddOas2MapuhKU2surEb3Kz3RCIpta4bXgTQMt9wawcZSSpvnfT
zs5sehYvBFszL3MV98Uc50HXMf7gykRCmPRmB9S+f+kiVRvQDHfc9nRNng2XgcotU
KAE16PQM8GihQ0C+EcXHouyud5CRJGfyurokR1H/jY3BiRAG5c+6
-----END RSA PRIVATE KEY-----
```

El ambiente de integración, cuenta con un emisor de pruebas, al que serán redirigidos para realizar la autenticación.

Para autenticar, el usuario (RUT) a ingresar es 11.111.111-1 y la clave es 123.

Una vez autenticados, dentro del emisor podrán aceptar ó rechazar la transacción.

Para pruebas de **Crédito** a continuación se listan las tarjetas de crédito a utilizar:

MARCA	VISA	MASTERCARD
Nº de Tarjeta	4051885600446623	5186059559590568
Año Expiración	Cualquiera	Cualquiera
Mes Expiración	Cualquiera	Cualquiera
CVV	123	123
Resultado	APROBADO	RECHAZADO

Para pruebas de **Redcompra** utilizar la siguiente información:

Nº de Tarjeta	4051885600446623	5186059559590568
Resultado	APROBADO	RECHAZADO

**Nota Importante:** Transbank no se hace responsable por el uso de otras tarjetas de crédito no indicadas en el manual. Si utiliza sus tarjetas propias, las transacciones podrían ser aprobadas, sin embargo recuerde que el ambiente considerado en este manual es un ambiente de integración y no productivo, lo que implica que no se verá reflejado los cargos en la cartola del tarjeta habiente ni en los abonos de las liquidaciones del comercio.

Al momento de la redacción de este manual, se mantiene operativo el ambiente de Integración de Webpay sólo crédito, por lo que se deberá cambiar la url de los parámetros URLCGITRA y URLCGIMEDTRA. En ambiente de integración estos valores deben ser /filtroUnificado/bp\_revision.cgi y /filtroUnificado/bp\_validacion.cgi respectivamente.

En ambiente de producción estos valores deberán ser /cgi-bin/bp\_revision.cgi y /cgi-bin/bp\_validacion.cgi respectivamente.

A modo de ejemplo, se adjunta la configuración para integración, de un comercio ficticio transacción normal pesos, ip 123.123.123.123.

```
IDCOMERCIO    = 597020000501
MEDCOM        = 1
TBK_KEY_ID    = 101
PARAMVERIFCOM = 1
URLCGICOM     = http://www.comercioFicticio.cl/cgi-bin/tbk_bp_resultado.cgi
SERVERCOM     = 123.123.123.123
PORTCOM       = 80
WHITELISTCOM  = ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 0123456789./:=&?_
HOST          = 123.123.123.123
WPORT         = 80
URLCGITRA     = /filtroUnificado/bp_revision.cgi
URLCGIMEDTRA  = /filtroUnificado/bp_validacion.cgi
SERVERTRA     = https://webpay3gint.transbank.cl
PORTTRA       = 443
PREFIJO_CONF_TR = HTML_
HTML_TR_NORMAL = http://www.comercioFicticio.cl/webpay/compra.php
```

## ANEXO B

### Recomendaciones a la seguridad del KCC.

1. Dejar habilitado un puerto exclusivo para la comunicación con Transbank S.A. en el firewall del comercio si es que lo tuviese.
2. Manejar en forma segura el par de llaves generadas, en especial privada.pem, para evitar algún posible plagio de la tienda. La llave, junto con el código de comercio, constituyen la identidad de su tienda de comercio electrónico.
3. Validar que no se pueda tener acceso al contenido de CGI-BIN, en especial a llave privada del comercio, correspondiente al archivo privada.pem ubicada en cgi-bin/maestros/. La validación se debe realizar a través del browser invocando las siguientes URLs:

```
SERVERCOM:PORTCOM/.../cgi-bin/  
SERVERCOM: PORTCOM/.../cgi-bin/datos  
SERVERCOM:PORTCOM/.../cgi-bin/maestros  
SERVERCOM: PORTCOM/.../cgi-bin/log
```

Resultado esperado para todos los casos: No permite listar y acceder a contenidos.

4. Las páginas de cierre de transacciones deben ubicarse en un directorio virtual distinto del resto de las páginas de la tienda virtual. Este directorio debe protegerse de tal forma que permita la ejecución de dichas páginas sólo desde el servidor donde se encuentra tbk\_bp\_resultado. Habitualmente el mismo directorio.



## ANEXO C

### Alcance de la validación de integración entre el comercio y Transbank.

Los productos y/o servicios ofertados en el sitio web deben corresponder al rubro indicado en el contrato suscrito con Transbank.

Página de cierre debe incorporar validación de:

- ✓ MAC - El comercio debe validar y controlar que el certificado / MAC sea de Transbank en cada transacción.
- ✓ Orden de compra (OC) - La OC debe ser única- El comercio debe validar que la OC, enviada por el comercio, sea igual a la OC recibida desde Transbank.
- ✓ Monto - El comercio debe validar y controlar que el monto enviado por el comercio, sea igual al monto recibido desde Transbank.

Para dar por aceptado este ítem, se requiere evidencias que muestren el cumplimiento de los puntos anteriores. (Revisar Anexo Formato de evidencias).

- Información a desplegar en página de Éxito o Resultado. El Comercio Internet está obligado a entregar al tarjetahabiente, una vez concluida la venta, un recibo de la compra que debe contener, como mínimo, la siguiente información:
  - ✓ Número de Orden de Compra.
  - ✓ Monto y moneda de la transacción.
  - ✓ Código de Autorización de la transacción.
  - ✓ Fecha de la transacción.
  - ✓ Hora de la transacción.
  - ✓ Número de Cuotas: Se debe indicar el número de cuotas informado por Transbank (para las transacciones sin cuotas se debe informar 00).
  - ✓ Tipo de Cuotas: se debe señalar lo siguiente dependiendo de la respuesta informada por Transbank:

Respuesta informada por Transbank	VD	VN	VC	SI	S2	NC
Tipo de Pago	Débito	Crédito	Crédito	Crédito	Crédito	Crédito
Tipo Cuotas	Venta débito	Sin cuotas	Cuotas normales	Sin interés	Sin interés	Sin interés
Número de cuotas	0	0	4-48	3	2	2-10

- Información a desplegar en página de Fracaso, debe contener como mínimo la siguiente información:
  - ✓ Transacción Rechazada
  - ✓ Número de Orden de Compra.

El siguiente texto es opcional:

Las posibles causas de este rechazo son:

- Error en el ingreso de los datos de su tarjeta (fecha y/o código de seguridad).
- Su tarjeta no cuenta con saldo suficiente.
- Tarjeta aun no habilita en el sistema financiero.

Información de Etapas de implementación:

**Integración:** Los plazos son de responsabilidad del comercio en la integración del kit, en caso de requerir apoyo, favor dirigir sus consultas a [sosporte@transbank.cl](mailto:sosporte@transbank.cl)

**Validación:** Proceso que consta de la revisión de evidencias enviadas por el comercio. Este es un requisito necesario para el paso de producción, en caso de requerir apoyo favor dirigir sus consultas a [sosporte@transbank.cl](mailto:sosporte@transbank.cl) .  
**Transbank se reserva el derecho a efectuar pruebas de validación complementarias que consisten en pruebas en el sitio del comercio.**

**Puesta en producción:** Proceso en el cual se realiza la carga de llaves dejando operativo al comercio. Requisito en esta etapa es realizar una transacción en producción. Esta última acción es realizada por el comercio.

## ANEXO D

### Tipificación de errores KCC

- 1 ERR\_LECTURA\_INPUT
- 2 ERR\_LECTURA\_PARAMETROS
- 3 ERR\_PARAM\_CODIGO\_COMERCIO
- 4 ERR\_PARAM\_URL\_CGI\_COMERCIO
- 5 ERR\_PARAM\_SERVIDOR\_COMERCIO
- 6 ERR\_PARAM\_PUERTO\_COMERCIO
- 7 ERR\_PARAM\_URL\_CGI\_TRANSBANK
- 8 ERR\_PARAM\_SERVIDOR\_TRANSBANK
- 9 ERR\_PARAM\_PUERTO\_TRANSBANK
- 10 ERR\_MEM\_MENSAJE
- 11 ERR\_MAC
- 12 ERR\_VERSION\_KCC
- 13 ERR\_MSG\_OLDKCC
- 14 ERR\_CODIGO\_COMERCIO
- 15 ERR\_DATOS\_COMERCIO
- 16 ERR\_TIPO\_TRANSACCION
- 17 ERR\_URL\_CGI\_COMERCIO
- 18 ERR\_SERVIDOR\_COMERCIO
- 19 ERR\_PUERTO\_COMERCIO
- 20 ERR\_CVV
- 21 ERR\_DUPLICADO
- 22 ERR\_ESTADO\_COMERCIO
- 23 ERR\_TEMPLATE\_COMPRA
- 24 ERR\_TEMPLATE\_COMPRA\_DET
- 25 ERR\_FECHA\_EXPIRACION
- 26 ERR\_MONEDA\_INVALIDA
- 27 ERR\_TEMPLATE\_REINTENTO
- 28 ERR\_EDITA\_MONTO
- 29 ERR\_ORDEN\_COMPRA
- 30 ERR\_ID\_TRANSACCION
- 31 ERR\_URL\_EXITO
- 32 ERR\_URL\_FRACASO
- 33 ERR\_MONTO
- 34 ERR\_REFERER
- 35 ERR\_NUM\_TRX
- 36 ERR\_MAX\_TIENDA
- 37 ERR\_CODIGO\_TIENDA

38 ERR\_TIENDA\_NO\_SOCIADA  
39 ERR\_MONTO\_TIENDA  
40 ERR\_CONF\_TIENDA  
41 ERR\_URL\_RESULTADO  
42 ERR\_CONECTA\_SERVIDOR  
43 ERR\_SOCKET\_LECTURA  
44 ERR\_SOCKET\_ESCRITURA  
45 ERR\_TIMEOUT\_ACK  
46 ERR\_OBTENER\_ACK  
47 ERR\_ORDEN\_TIENDA  
48 ERR\_NUMERO\_TARJETA  
49 ERR\_NUMERO\_CUOTAS  
50 ERR\_MES\_VENCIMIENTO  
51 ERR\_AÑO\_VENCIMIENTO  
52 ERR\_TARJETA  
53 ERR\_TIPO\_PAGO  
54 ERR\_RESPUESTA\_BASE24  
55 ERR\_ACK  
57 ERR\_TIENDA\_EN\_NULO  
58 ERR\_PARAM\_PREF\_CONF\_TR  
59 ERR\_PARAM\_URL\_CONF\_TR  
60 ERR\_PARAM\_PUERTO\_CONF\_TR  
61 ERR\_PARAM\_SERVIDOR\_CONF\_TR  
62 ERR\_FECHA\_TRANSACCION  
63 ERR\_ABRIR\_BITACORA  
64 ERR\_FALLO\_CUOTA\_NORMAL  
65 ERR\_FALLO\_CUOTA\_C3C  
66 ERR\_MONEDA\_DESCONOCIDA  
67 ERR\_TIENDA\_NOENCONTRADA  
68 ERR\_TIENDA\_NOASOCIADA  
69 ERR\_TIENDA\_DIF\_MONEDA  
70 ERR\_SINMEMORIA  
71 ERR\_MONTO\_DESCUADRADO  
72 ERR\_TRX\_DESCUADRADAS  
73 ERR\_TEMPLATE\_ONECLICK  
74 ERR\_TIENDA\_TIPO\_PAGO  
75 ERR\_ID\_USUARIO  
76 ERR\_USUARIO\_YA\_REGISTRADO  
77 ERR\_INSERTAR\_ONECLICK  
78 ERR\_USUARIO\_NO\_REGISTRADO  
79 ERR\_LEER\_ONECLICK  
80 ERR\_TEMPLATE\_TRANSICION

81 ERR\_LOGO\_WEBPAY  
82 ERR\_TEMPLATE\_LOGO\_WEBPAY  
83 ERR\_ACTUALIZAR\_ONECLICK  
84 ERR\_ELIMINAR\_ONECLICK  
85 ERR\_TEMPLATE\_MENSAJE  
86 ERR\_NUMERO\_TARJETA\_ANTIGUA  
87 ERR\_ONECLICK\_DESHABILITADO  
88 ERR\_TR\_COMPLETA\_DESHABILITADA  
89 ERR\_CAMBIO\_MULTICODE  
90 ERR\_RESPUESTA  
91 ERR\_LARGO\_TIENDA  
92 ERR\_CORRESPONDENCIA\_CUOTAS  
93 ERR\_TIPO\_PAGO\_DESHABILITADO  
94 ERR\_ENVIO\_REG\_ONECLICK  
95 ERR\_ID\_SESION  
96 ERR\_TIPO\_TBK\_TIPO\_TRANSACCION  
97 ERR\_TIPO\_TBK\_CODIGO\_COMERCIO  
98 ERR\_TIPO\_TBK\_ID\_TRANSACCION  
99 ERR\_TIPO\_TBK\_URL\_CGI\_COMERCIO  
100 ERR\_TIPO\_TBK\_SERVIDOR\_COMERCIO  
101 ERR\_TIPO\_TBK\_PUERTO\_COMERCIO  
102 ERR\_TIPO\_TBK\_VERSION\_KCC  
103 ERR\_TIPO\_TBK\_MAC  
104 ERR\_TIPO\_TBK\_MONTO  
105 ERR\_TIPO\_TBK\_ORDEN\_COMPRA  
106 ERR\_TIPO\_TBK\_ID\_SESION  
107 ERR\_TIPO\_TBK\_URL\_EXITO  
108 ERR\_TIPO\_TBK\_URL\_FRACASO  
109 ERR\_TIPO\_TBK\_TARJETA  
110 ERR\_TIPO\_TBK\_TIPO\_PAGO  
111 ERR\_TIPO\_TBK\_NUMERO\_CUOTAS  
112 ERR\_TIPO\_TBK\_NUMERO\_TARJETA  
113 ERR\_TIPO\_TBK\_MES\_VENCIMIENTO  
114 ERR\_TIPO\_TBK\_AÑO\_VENCIMIENTO  
115 ERR\_TIPO\_TBK\_CVV  
116 ERR\_LEN\_TBK\_TIPO\_TRANSACCION  
117 ERR\_LEN\_TBK\_CODIGO\_COMERCIO  
118 ERR\_LEN\_TBK\_ID\_TRANSACCION  
119 ERR\_LEN\_TBK\_URL\_CGI\_COMERCIO  
120 ERR\_LEN\_TBK\_SERVIDOR\_COMERCIO  
121 ERR\_LARGO\_TIPO\_TBK\_PUERTO\_COMERCIO  
122 ERR\_LEN\_TBK\_VERSION\_KCC

123 ERR\_LEN\_TBK\_MAC  
124 ERR\_LEN\_TBK\_MONTO  
125 ERR\_LEN\_TBK\_ORDEN\_COMPRA  
126 ERR\_LEN\_TBK\_ID\_SESION  
127 ERR\_LEN\_TBK\_URL\_EXITO  
128 ERR\_LEN\_TBK\_URL\_FRACASO  
129 ERR\_LEN\_TBK\_TARJETA  
130 ERR\_LEN\_TBK\_TIPO\_PAGO  
131 ERR\_LEN\_TBK\_NUMERO\_CUOTAS  
132 ERR\_LEN\_TBK\_NUMERO\_TARJETA  
133 ERR\_LEN\_TBK\_MES\_VENCIMIENTO  
134 ERR\_LEN\_TBK\_ANO\_VENCIMIENTO  
135 ERR\_LEN\_TBK\_CVV  
136 ERR\_TIPO\_TBK\_URL\_RESULTADO  
137 ERR\_TIPO\_TBK\_NUM\_TRX  
138 ERR\_TIPO\_TBK\_CODIGO\_TIENDA  
139 ERR\_TIPO\_TBK\_ORDEN\_TIENDA  
140 ERR\_TIPO\_TBK\_MONTO\_TIENDA  
141 ERR\_TIPO\_TBK\_ID\_USUARIO  
142 ERR\_LEN\_TBK\_URL\_RESULTADO  
143 ERR\_LARGO\_TIPO\_TBK\_NUM\_TRX  
144 ERR\_LARGO\_TIPO\_TBK\_CODIGO\_TIENDA  
145 ERR\_LARGO\_TIPO\_TBK\_ORDEN\_TIENDA  
146 ERR\_LARGO\_TIPO\_TBK\_MONTO\_TIENDA  
147 ERR\_LARGO\_TIPO\_TBK\_ID\_USUARIO  
148 ERR\_LEN\_PARAM\_TR\_NORMAL  
149 ERR\_LEN\_PARAM\_TR\_MALL  
150 ERR\_LEN\_PARAM\_TR\_COMPLETA  
151 ERR\_LEN\_PARAM\_TR\_ONECLICK  
152 ERR\_LEN\_PARAM\_TR\_INGRESO\_ONECLICK  
153 ERR\_LEN\_PARAM\_TR\_ELIMINACION\_ONECLICK  
154 ERR\_LEN\_PARAM\_TR\_MODIFICACION\_ONECLICK  
155 ERR\_LEN\_PARAM\_TR\_MALL\_COMPLETA  
156 ERR\_LEN\_PARAM\_TR\_MALL\_ONECLICK  
157 ERR\_LEN\_PARAM\_TR\_LOGO\_WEBPAY  
158 ERR\_CANT\_PARAM\_TR\_NORMAL  
159 ERR\_CANT\_PARAM\_TR\_MALL  
160 ERR\_CANT\_PARAM\_TR\_COMPLETA  
161 ERR\_CANT\_PARAM\_TR\_ONECLICK  
162 ERR\_CANT\_PARAM\_TR\_INGRESO\_ONECLICK  
163 ERR\_CANT\_PARAM\_TR\_ELIMINACION\_ONECLICK  
164 ERR\_CANT\_PARAM\_TR\_MODIFICACION\_ONECLICK

165 ERR\_CANT\_PARAM\_TR\_MALL\_COMPLETA  
166 ERR\_CANT\_PARAM\_TR\_MALL\_ONECLICK  
167 ERR\_CANT\_PARAM\_TR\_LOGO\_WEBPAY  
170 ERR\_TIPO\_TBK\_FECHA\_EXPIRACION  
171 ERR\_LEN\_TBK\_FECHA\_EXPIRACION  
172 ERR\_TIPO\_TBK\_URL\_COMERCIO  
173 ERR\_LEN\_TIPO\_TBK\_URL\_COMERCIO  
174 ERR\_TIPO\_TBK\_MONTO\_CUOTA  
175 ERR\_LEN\_TBK\_MONTO\_CUOTA  
176 ERR\_MONTO\_CUOTA  
177 ERR\_TR\_TASA\_INTERES\_DESHABILITADA  
178 ERR\_LEN\_PARAM\_TR\_TASA\_INTERES\_MAX  
179 ERR\_CANT\_PARAM\_TR\_TASA\_INTERES\_MAX  
180 ERR\_CONSISTENCIA\_CIC  
181 ERR\_FALLO\_CUOTA\_CIC  
241 ERR\_TBK\_TOKEN\_NO\_ENCONTRADO  
242 ERR\_RESPUESTA\_AUTH  
243 ERR\_EMITOR\_NO\_ENCONTRADO  
244 ERR\_ARCHIVO\_EMITOR  
245 ERR\_COMERCIO\_NO\_ENCONTRADO  
246 ERR\_ARCHIVO\_COMERCIO  
247 ERR\_BIN\_NO\_ENCONTRADO  
248 ERR\_ARCHIVO\_BINES  
249 ERR\_EMITOR\_NO\_PARTICIPA  
250 ERR\_COMERCIO\_NO\_PARTICIPA  
251 ERR\_TRANSACCION\_NO\_PARTICIPA  
252 ERR\_NO\_SE\_PUEDE\_GENERAR\_TOKEN  
253 ERR\_BIN\_NO\_PARTICIPA  
254 ERR\_VVR  
255 ERR\_TOKEN\_STATUS  
256 ERR\_GEN\_TOKENCOM  
257 ERR\_SIN\_VALIDACION  
258 ERR\_TBK\_PARAM  
259 ERR\_SSL\_CONEXION  
260 ERR\_SSL\_ESCRITURA  
261 ERR\_SSL\_REINTENTAR  
262 ERR\_SSL\_LECTURA  
263 ERR\_PUB\_KEY  
264 ERR\_TOKEN\_INVFOR  
265 ERR\_BUFF\_OVERFLOW  
266 ERR\_TRANSACCION\_DEBITO\_NO\_PERMITIDA  
267 ERR\_TRANSACCION\_NECESITA\_AUTENTICAR

268 ERR\_TRANSACCION\_DEBITO\_FALTA\_PARAMETRO  
269 ERR\_CORRESPONDENCIA\_TRANSACCION\_DEBITO  
270 ERR\_CANT\_PARAM\_COMUNES  
271 ERR\_FALTA\_PARAM  
272 ERR\_TIMEOUT  
273 ERR\_MONTO\_CERO  
274 ERR\_TRANSACCION\_NO\_ENCONTRADA  
275 ERR\_INSTRUMENTO\_DE\_PAGO  
276 ERR\_IR\_PAGINA\_FALLO  
277 ERR\_TIPO\_CONEXION\_COMERCIO  
278 ERR\_MALL\_NO\_IGUAL\_TIENDA  
279 ERR\_MALL\_SIN\_TIENDA  
280 ERR\_HTTP  
281 ERR\_FILE\_POPULATOR  
282 ERR\_FORMAT\_PARAM  
283 ERR\_ENCRPTION  
284 ERR\_EXPIRED\_TIME  
285 ERR\_RUT  
286 ERR\_SET\_STATUS  
287 ERR\_PARAM\_LEN  
288 ERR\_DOUBLE\_SUBMIT  
289 ERR\_INCONSISTENT\_BIN\_INFO  
290 ERR\_VCI\_DECISION\_TABLE  
291 ERR\_NOT\_FOUND\_PARAM  
292 ERR\_INVALID\_STATUS  
293 ERR\_INVOCATION\_METHOD  
294 ERR\_ANOTHER\_TRANSACCION  
295 ERR\_SEND\_MAIL  
296 ERR\_UNKNOWN  
300 ERR\_PARAM\_TBK\_TIPO\_TRANSACCION  
301 ERR\_LEN\_PARAM\_TBK\_TIPO\_TRANSACCION  
302 ERR\_FORMATO\_PARAM\_TBK\_TIPO\_TRANSACCION  
303 ERR\_PARAM\_TBK\_MONTO  
304 ERR\_LEN\_PARAM\_TBK\_MONTO  
305 ERR\_FORMATO\_PARAM\_TBK\_MONTO  
306 ERR\_PARAM\_TBK\_ORDEN\_COMPRA  
307 ERR\_LEN\_PARAM\_TBK\_ORDEN\_COMPRA  
308 ERR\_FORMATO\_PARAM\_TBK\_ORDEN\_COMPRA  
309 ERR\_PARAM\_TBK\_ID\_SESION  
310 ERR\_LEN\_PARAM\_TBK\_ID\_SESION  
311 ERR\_FORMATO\_PARAM\_TBK\_ID\_SESION  
312 ERR\_PARAM\_TBK\_URL\_EXITO



313 ERR\_LEN\_PARAM\_TBK\_URL\_EXITO  
314 ERR\_FORMATO\_PARAM\_TBK\_URL\_EXITO  
315 ERR\_PARAM\_TBK\_URL\_FRACASO  
316 ERR\_LEN\_PARAM\_TBK\_URL\_FRACASO  
317 ERR\_FORMATO\_PARAM\_TBK\_URL\_FRACASO  
318 ERR\_PARAM\_TBK\_ID\_USUARIO  
319 ERR\_LEN\_PARAM\_TBK\_ID\_USUARIO  
320 ERR\_FORMATO\_PARAM\_TBK\_ID\_USUARIO  
321 ERR\_PARAM\_TBK\_TIPO\_PAGO  
322 ERR\_LEN\_PARAM\_TBK\_TIPO\_PAGO  
323 ERR\_FORMATO\_PARAM\_TBK\_TIPO\_PAGO  
324 ERR\_PARAM\_TBK\_NUMERO\_CUOTAS  
325 ERR\_LEN\_PARAM\_TBK\_NUMERO\_CUOTAS  
326 ERR\_FORMATO\_PARAM\_TBK\_NUMERO\_CUOTAS  
327 ERR\_PARAM\_TBK\_MONTO\_CUOTA  
328 ERR\_LEN\_PARAM\_TBK\_MONTO\_CUOTA  
329 ERR\_FORMATO\_PARAM\_TBK\_MONTO\_CUOTA  
330 ERR\_PARAM\_TBK\_NUM\_TRX  
331 ERR\_LEN\_PARAM\_TBK\_NUM\_TRX  
332 ERR\_FORMATO\_PARAM\_TBK\_NUM\_TRX  
333 ERR\_PARAM\_TBK\_NUMERO\_TARJETA  
334 ERR\_LEN\_PARAM\_TBK\_NUMERO\_TARJETA  
335 ERR\_FORMATO\_PARAM\_TBK\_NUMERO\_TARJETA  
336 ERR\_PARAM\_TBK\_FECHA\_EXPIRACION  
337 ERR\_LEN\_PARAM\_TBK\_FECHA\_EXPIRACION  
338 ERR\_FORMATO\_PARAM\_TBK\_FECHA\_EXPIRACION  
339 ERR\_PARAM\_TBK\_CVV  
340 ERR\_LEN\_PARAM\_TBK\_CVV  
341 ERR\_FORMATO\_PARAM\_TBK\_CVV  
342 ERR\_PARAM\_TBK\_URL\_RESULTADO  
343 ERR\_LEN\_PARAM\_TBK\_URL\_RESULTADO  
344 ERR\_FORMATO\_PARAM\_TBK\_URL\_RESULTADO  
345 ERR\_PARAM\_TBK\_RESPUESTA  
346 ERR\_LEN\_PARAM\_TBK\_RESPUESTA  
347 ERR\_FORMATO\_PARAM\_TBK\_RESPUESTA  
348 ERR\_PARAM\_TBK\_FINAL\_NUMERO\_TARJETA  
349 ERR\_LEN\_PARAM\_TBK\_FINAL\_NUMERO\_TARJETA  
350 ERR\_FORMATO\_PARAM\_TBK\_FINAL\_NUMERO\_TARJETA  
351 ERR\_PARAM\_TBK\_FECHA\_CONTABLE  
352 ERR\_LEN\_PARAM\_TBK\_FECHA\_CONTABLE  
353 ERR\_FORMATO\_PARAM\_TBK\_FECHA\_CONTABLE  
354 ERR\_PARAM\_TBK\_FECHA\_TRANSACCION

355 ERR\_LEN\_PARAM\_TBK\_FECHA\_TRANSACCION  
356 ERR\_FORMATO\_PARAM\_TBK\_FECHA\_TRANSACCION  
357 ERR\_PARAM\_TBK\_HORA\_TRANSACCION  
358 ERR\_LEN\_PARAM\_TBK\_HORA\_TRANSACCION  
359 ERR\_FORMATO\_PARAM\_TBK\_HORA\_TRANSACCION  
360 ERR\_PARAM\_TBK\_ID\_TRANSACCION  
361 ERR\_LEN\_PARAM\_TBK\_ID\_TRANSACCION  
362 ERR\_FORMATO\_PARAM\_TBK\_ID\_TRANSACCION  
363 ERR\_PARAM\_TBK\_CODIGO\_AUTORIZACION  
364 ERR\_LEN\_PARAM\_TBK\_CODIGO\_AUTORIZACION  
365 ERR\_FORMATO\_PARAM\_TBK\_CODIGO\_AUTORIZACION  
366 ERR\_PARAM\_TBK\_MAC  
367 ERR\_LEN\_PARAM\_TBK\_MAC  
368 ERR\_FORMATO\_PARAM\_TBK\_MAC  
369 ERR\_PARAM\_TBK\_CODIGO\_TIENDA\_MXXX  
370 ERR\_LEN\_PARAM\_TBK\_CODIGO\_TIENDA\_MXXX  
371 ERR\_FORMATO\_PARAM\_TBK\_CODIGO\_TIENDA\_MXXX  
372 ERR\_PARAM\_TBK\_ORDEN\_TIENDA\_MXXX  
373 ERR\_LEN\_PARAM\_TBK\_ORDEN\_TIENDA\_MXXX  
374 ERR\_FORMATO\_PARAM\_TBK\_ORDEN\_TIENDA\_MXXX  
375 ERR\_PARAM\_TBK\_MONTO\_TIENDA\_MXXX  
376 ERR\_LEN\_PARAM\_TBK\_MONTO\_TIENDA\_MXXX  
377 ERR\_FORMATO\_PARAM\_TBK\_MONTO\_TIENDA\_MXXX  
378 ERR\_PARAM\_TBK\_MONTO\_CUOTA\_MXXX  
379 ERR\_LEN\_PARAM\_TBK\_MONTO\_CUOTA\_MXXX  
380 ERR\_FORMATO\_PARAM\_TBK\_MONTO\_CUOTA\_MXXX  
381 ERR\_PARAM\_TBK\_NUMERO\_CUOTAS\_MXXX  
382 ERR\_LEN\_PARAM\_TBK\_NUMERO\_CUOTAS\_MXXX  
383 ERR\_FORMATO\_PARAM\_TBK\_NUMERO\_CUOTAS\_MXXX  
384 ERR\_PARAM\_TBK\_TIPO\_PAGO\_MXXX  
385 ERR\_LEN\_PARAM\_TBK\_TIPO\_PAGO\_MXXX  
386 ERR\_FORMATO\_PARAM\_TBK\_TIPO\_PAGO\_MXXX  
387 ERR\_PARAM\_TBK\_COD\_AUT\_MXXX  
388 ERR\_LEN\_PARAM\_TBK\_COD\_AUT\_MXXX  
389 ERR\_FORMATO\_PARAM\_TBK\_COD\_AUT\_MXXX  
390 ERR\_PARAM\_TBK\_COD\_RESP\_MXXX  
391 ERR\_LEN\_PARAM\_TBK\_COD\_RESP\_MXXX  
392 ERR\_FORMATO\_PARAM\_TBK\_COD\_RESP\_MXXX  
393 ERR\_CORRESPONDENCIA\_TIPO\_PAGO\_NUMERO\_CUOTAS  
394 ERR\_NRO\_TIENDAS  
396 ERR\_CONFIGDAT\_WHITELISTCOM  
397 ERR\_STRUCT\_ARCH\_PARAMCOM

398 ERR\_ARCH\_PARAMCOM  
399 ERR\_ARCH\_PARAMCOM\_VARIABLE\_NO\_EXISTE  
400 ERR\_ARCH\_PARAMCOM\_CGI\_NO\_ENCONTRADA  
402 ERR\_PARAM\_MEDIO\_COMERCIO  
403 ERR\_PARAM\_URL\_CGI\_TBK\_VALIDACION  
404 ERR\_ARCH\_TBK\_PARAM  
405 ERR\_FCOMUNES\_NO\_HAY\_ESPACIO\_SUFICIENTE\_UNO  
406 ERR\_FCOMUNES\_NO\_HAY\_ESPACIO\_SUFICIENTE\_DOS  
425 ERR\_ENCRYPTANDO\_CODIGO\_COMERCIO  
426 ERR\_LARGO\_ENC\_CODIGO\_COMERCIO  
427 ERR\_ENCRYPTANDO\_TBK\_PARAM  
428 ERR\_LARGO\_ENC\_TBK\_PARAM  
429 ERR\_SSL\_CTX  
430 ERR\_SSL\_CONEXION  
431 ERR\_SSL\_ESCRITURA  
432 ERR\_SSL\_ESCRITURA\_CONEXION  
433 ERR\_SSL\_REINTENTAR  
434 ERR\_SSL\_DATOS\_NO\_VALIDOS  
435 ERR\_LARGO\_PARAMETRO  
436 ERR\_LARGO\_TBK\_PARAM\_INSUFICIENTE  
440 ERR\_POST\_CODIGO\_COMERCIO  
441 ERR\_POST\_CODIGO\_COMERCIO\_ENC  
442 ERR\_POST\_TBK\_PARAM  
443 ERR\_LARGO\_UB64\_CODIGO\_COMERCIO  
444 ERR\_DESENCRIPTANDO\_CODIGO\_COMERCIO  
445 ERR\_LARGO\_A\_DESENCRIPTAR\_CODIGO\_COMERCIO  
446 ERR\_LARGO\_UB64\_TBK\_PARAM  
447 ERR\_DESENCRIPTANDO\_TBK\_PARAM  
448 ERR\_LARGO\_A\_DESENCRIPTAR\_TBK\_PARAM  
449 ERR\_P\_CC\_CLARO\_DISTINTO\_P\_DESENCRIPTADO  
450 ERR\_P\_CC\_CLARO\_DISTINTO\_PARAM\_CC  
451 ERR\_P\_CC\_DESENCRIPTADO\_DISTINTO\_PARAM\_CC  
452 ERR\_LARGO\_MY\_PARSE\_CGI  
460 ERR\_BUFFER\_OVERFLOW\_CERO\_PAGO  
461 ERR\_BUFFER\_OVERFLOW\_DOS\_PAGO  
462 ERR\_BUFFER\_OVERFLOW\_TRES\_PAGO  
463 ERR\_BUFFER\_OVERFLOW\_CUATRO\_PAGO  
464 ERR\_BUFFER\_OVERFLOW\_CINCO\_PAGO  
465 ERR\_BUFFER\_OVERFLOW\_SEIS\_PAGO  
466 ERR\_BUFFER\_OVERFLOW\_SIETE\_PAGO  
467 ERR\_BUFFER\_OVERFLOW\_OCHO\_RESULTADO  
468 ERR\_BUFFER\_OVERFLOW\_NUEVE\_RESULTADO

469 ERR\_BUFFER\_OVERFLOW\_DIEZ\_RESULTADO  
470 ERR\_BUFFER\_OVERFLOW\_ONCE\_FCOMUNES  
471 ERR\_BUFFER\_OVERFLOW\_DOCE\_FCOMUNES  
472 ERR\_BUFFER\_OVERFLOW\_TRECE\_FCOMUNES  
473 ERR\_BUFFER\_OVERFLOW\_CATORCE\_FCOMUNES  
474 ERR\_BUFFER\_OVERFLOW\_QUINCE\_FCOMUNES  
475 ERR\_BUFFER\_OVERFLOW\_DIECISEIS\_FCOMUNES  
476 ERR\_BUFFER\_OVERFLOW\_DIECISIETE\_FCOMUNES  
477 ERR\_BUFFER\_OVERFLOW\_DIECIOCHO\_FCOMUNES  
478 ERR\_BUFFER\_OVERFLOW\_DIECINUEVE\_FCOMUNES

## ANEXO E

### Cálculo del Valor Cuota

Las siguientes son las fórmulas utilizadas para el cálculo del Valor Presente (precio contado), y del Valor Cuota, referidas a la tasa máxima convencional.

El VALOR PRESENTE a la máxima se obtiene aplicando la siguiente fórmula:

$$\text{Valor Presente a Máxima} = \frac{V_C * [1 - (1 + T_m)^{-n}]}{T_m}$$

Donde:

Vc: es el monto de la cuota (informado en la Tx).

Tm: es la tasa Máxima Definida (almacenada en tabla).

n: es el número de cuotas (informado en Tx).

EL VALOR CUOTA se obtiene despejando de la fórmula anterior:

$$V_c = \frac{VP * T_m * (1 + T_m)^n}{(1 + T_m)^n - 1}$$

El extracto  $\frac{T_m * (1 + T_m)^n}{(1 + T_m)^n - 1}$  corresponde al factor F.

Luego:

$V_c = \text{Precio Contado} * F$

Donde:

F: es función del n° cuotas y de la Tm.

N° cuotas: 6 a 36

Tm: Se modifica mensualmente y se publica en el diario oficial entre el día 8 y 10 de cada mes.

## ANEXO F

### Ejemplo integración de tienda Transacción Normal ASP

A continuación se describe un ejemplo de integración de tienda con tipo de transacción Normal, sobre el ambiente de integración de Webpay. Los pasos a seguir son los siguientes:

- Paso 1: Descomprimiendo KCC 6.0.2
- Paso 2: Configurando archivo de parámetros tbk\_config.dat
- Paso 3: Configurando parámetros de transacción tbk\_param.txt
- Paso 4: Instalando la llave privada del comercio de prueba privada.pem y publica de transbank
- Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat
- Paso 6: Páginas de pago, cierre, éxito y fracaso

#### Requerimientos:

- ✓ Contar con un servidor web para ejecución de páginas .asp, ejemplo IIS
- ✓ Editor de texto
- ✓ Acceso a Internet con tráfico hacia <https://webpay3gint.transbank.cl> puerto 443

## **Paso 1: Descomprimiendo KCC 6.0.2**

---

El KCC versión 6.0.2 proporcionado por Transbank, es específico para cada de plataforma de sistema operativo (Windows, Linux ó Solaris), es un archivo comprimido con formato rar.

Se debe descomprimir y copiar el contenido de la carpeta CGI-BIN en la ubicación física del disco, en donde residen los CGIs en el servidor Web.

## **Paso 2: Configurando archivo de parámetros tbk\_config.dat**

---

Supuestos del ejemplo:

Código de Comercio	59702000541
Tipo De Transacción	Normal
Dirección Física	SUPERCOMERCE
IP Servidor de Comercio	198.22.250.29
Puerto de acceso al Servidor de Comercio	80
URL Servidor de Pago	https://webpay3gint.transbank.cl
Puerto de acceso al Servidor de Pago	443
Plataformas	Windows NT
Servidor Web	IIS
Carpeta de archivos de tienda virtual y CGI-BIN	D:\tienda

Archivo tbk\_config.dat:

```
IDCOMERCIO      = 597026007976
MEDCOM          = 1
TBK_KEY_ID      = 101
PARAMVERIFCOM   = 1
URLCGICOM       = http://www.comercioFicticio.cl/cgi-
bin/tbk_bp_resultado.cgi
SERVERCOM       = 123.123.123.123
PORTCOM         = 80
WHITELISTCOM     =
ABCDEFGHIJKLMNOSTUVWXYZabcdefghijklmnopqrstuvwxyz 0123456789./:=&?_
HOST            = 123.123.123.123
WPORT           = 80
URLCGITRA       = /filtroUnificado/bp_revision.cgi
URLCGIMEDTRA    = /filtroUnificado/bp_validacion.cgi
SERVERTRA       = https://webpay3gint.transbank.cl
PORTTRA         = 443
PREFIJO_CONF_TR = HTML_
HTML_TR_NORMAL  = http://www.comercioFicticio.cl/xt_compra.asp
```

### Paso 3: Configurando parámetros de transacción tbk\_param.txt

---

Verificar que los parámetros para la transacción normal (TRX\_NORMAL) se encuentren bien configurados

```
BP_PAGO>
  <TR_NORMAL>
    TBK_TIPO_TRANSACCION#A#50#1
    TBK_MONTO#M#10#1
    TBK_ORDEN_COMPRA#A#26#1
    TBK_ID_SESION#A#61#0
    TBK_URL_FRACASO#A#256#1
    TBK_URL_EXITO#A#256#1
    TBK_MONTO_CUOTA#M#9#0
    TBK_NUMERO_CUOTAS#N#2#0
  </TR_NORMAL>
  ...
```





#### **Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat**

---

Validar la siguiente línea al archivo tbk\_trace.dat:

```
GLEVEL = 7
```

#### **Paso 6: Páginas de pago, cierre, éxito y fracaso**

---

##### **Página de pago**

La llamada al KCC se hace a través del cgi tbk\_bp\_pago el que a su vez debe ser llamado desde la página de pago del comercio. Este archivo tiene como objetivo recoger la información de la compra que se pretende realizar. Así de acuerdo a los parámetros de entrada de una transacción normal, este formulario de pago tiene la siguiente forma:

```

<%
' GENERA ORDEN DE COMPRA A PARTIR DE FECHA

FECHAACTUAL = NOW()

ANO = YEAR(FECHAACTUAL)
MES = MONTH(FECHAACTUAL)
DIA = DAY(FECHAACTUAL)
MINUTO = MINUTE(FECHAACTUAL)
SEGUNDO = SECOND(FECHAACTUAL)

OC = "OC_"&ANO&MES&DIA&MINUTO&SEGUNDO
%>

<HTML> <HEAD> <TITLE>TIENDA ASP KCC 6.0.2 - TRANSACCION NORMAL</TITLE> </HEAD>

<BODY BGCOLOR="#3069C6" TOPMARGIN="10" LEFTMARGIN="0" MARGINWIDTH="0" MARGINHEIGHT="0">
<BR> <P ALIGN="CENTER">
  <FONT FACE=ARIAL SIZE="5" COLOR="WHITE"> TRANSACCION NORMAL - WINDOWS ASP – KCC6.0.2</FONT>
</P> <BR>

<FORM METHOD="POST" ACTION="/CGI-BIN/TBK_BP_PAGO.CGI">
<TABLE BORDER="0" ALIGN="CENTER">
<TR>
<TD ALIGN="CENTER">
  <FONT FACE=ARIAL SIZE="3" COLOR="WHITE">MONTO TRANSACCIÓN</FONT> <BR>
  <INPUT TYPE="TEXT" NAME="TBK_MONTO" VALUE="10000"> <BR>
</TD>
<TD ALIGN="CENTER"> <BR>
  <INPUT TYPE="HIDDEN" NAME="TBK_TIPO_TRANSACCION" VALUE="TR_NORMAL"> <BR>
</TD>
</TR>

<TR>
<TD ALIGN="CENTER">
  <FONT FACE="ARIAL" SIZE="3" COLOR="WHITE">Nº DE ORDEN</FONT> <BR>
  <INPUT TYPE="TEXT" NAME="TBK_ORDEN_COMPRA" VALUE="<%=OC%>"> <BR>
</TD>
<TD ALIGN="CENTER"> <BR>
  <INPUT TYPE="HIDDEN" NAME="TBK_ID_SESION" VALUE="<%=OC%>"> <BR>
</TD>
</TR>
</TABLE>

<TABLE BORDER=0 ALIGN="CENTER">
<TR>
<TD ALIGN="CENTER"> <BR>
  <INPUT TYPE="HIDDEN" NAME="TBK_URL_EXITO" SIZE="40"
  VALUE="HTTP://EXITO.ASP" SIZE="50"> <BR>
</TD>
<TD ALIGN="CENTER"> <BR>
  <INPUT TYPE="HIDDEN" NAME="TBK_URL_FRACASO" SIZE=40
  VALUE="HTTP://FRACASO.ASP" SIZE="50"> <BR>
</TD>
</TR>
</TABLE>

<TABLE BORDER="0" ALIGN="CENTER">
<TR> <TD ALIGN="CENTER"> <BR>
  <INPUT TYPE="SUBMIT" VALUE="PAGAR CON TARJETA DE CRÉDITO" SIZE="20"> </BR>
</TD>
</TR>
</TABLE>
</FORM>
</BODY>
</HTML>

```

URL ubicación TBK\_BP\_PAGO.CGI

URL de página de éxito

URL de página de fracaso

## Página De Cierre

Esta página extrae los parámetros de respuesta que entrega Transbank, realiza validación de MAC y si corresponde confirma la transacción entregando como respuesta "ACEPTADO". Para simplificar el ejemplo, se ha omitido las validaciones de monto y orden de compra que dependen de cada implementación del comercio.

xt\_compra.asp

```
<HTML>

<%

RESPUESTA = Request.Form("TBK_RESPUESTA")

if (RESPUESTA="0") then
    if (check_mac="CORRECTO") then
        ' Sólo si MAC ok, oc ok, monto ok se debe responder "ACEPTADO"
        ' Aqui debe validar estado de OC
        ' Aqui debe validar Monto
        response.write("ACEPTADO")
    end if
else
    ' Acepta no autorización de la transacción
    response.write("ACEPTADO")
end if

function check_mac ()

    OC = Request.Form("TBK_ORDEN_COMPRA")

    carpetaLogs = "d:\TIENDA\cgi-bin\log\"
    archivoBat = "d:\TIENDA\cgi-bin\tbk_check_mac.bat"
    ejecutable_CheckMac = "d:\TIENDA\cgi-bin\tbk_check_mac.exe"
    archivo_temporal = carpetaLogs & "DatosParaCheckMac_" & OC & ".txt"
    archivo_resultado = carpetaLogs & "ResultadoCheckMac_" & OC & ".txt"

    set filesys = CreateObject("Scripting.FileSystemObject")
    set file = filesys.CreateTextFile(archivo_temporal)

    ' Recupera parametros y guarda en archivo
    file.write(Request.Form())
    file.Close

    cmd = archivoBat & " " & ejecutable_CheckMac & " " & archivo_temporal & " " & archivo_resultado

    Set WshShell = server.CreateObject ("WScript.Shell")
    iReturn = WshShell.Run(cmd,1,true)
    set WshShell = nothing

    ' Lee resultado de validación de MAC
    set ArchResultado = CreateObject("Scripting.FileSystemObject")
    set tf = ArchResultado.OpenTextFile(archivo_resultado)

    check_mac = tf.readLine

end function

%>
</HTML>
```

### Nota:

En el ejemplo se hace uso del archivo tbk\_check\_mac.bat, el cual se entrega con el KCC 6.0.2 y permite centralizar la llamada al tbk\_check\_mac.exe y facilitar la generación del archivo de resultado, el cual almacenará el resultado de la validación del MAC (CORRECTO - INVALIDO).

## Página de éxito

El parámetro TBK\_URL\_EXITO corresponde a la URL que contiene a la página que se llamará en caso que la transacción se haya ejecutado correctamente. Esta página debiera desplegar información sobre el producto o servicio pagado, condiciones de entrega, etc. información que puede ser extraída desde la base de datos del comercio. La información puede ser desplegada recuperando el valor del parámetro TBK\_ORDEN\_DE\_COMPRA.

### Ejemplo página éxito:

exito.asp

```
<html>
<head>
<title>Ejemplo pagina de éxito</title>
</head>

<body bgcolor="#3069C6" topmargin="10" leftmargin="0"
marginwidth="0" marginheight="0">
<%
' Recupera número de Orden de Compra
OC = Request.Form("TBK_ORDEN_COMPRA")
carpeta = "d:\tienda\cgi-bin\log\"

archivo = carpeta & "DatosParaCheckMac_" & OC & ".txt"
set fs = CreateObject("Scripting.FileSystemObject")
set a = fs.OpenTextFile(archivo)
id_tbk = a.ReadAll
id_tbk = replace(id_tbk,"&","<br>")
a.Close
%>

<center>

<br><br><br><br>

<b>
<a>
<font face="arial" size="5" color="white">Transacción Exitosa !!!</font>
</a>
</b>

<br><br>

<font face="arial" size="3" color="white"> Orden de Compra :<b>
<% =request.form("TBK_ORDEN_COMPRA") %> </b> procesada</font>
<br></br>
<font face="arial" size="2" color="AAAAAA"><b><% =id_tbk %> </b> </font>
<br></br>
</center>
</body>
</html>
```

## Página de fracaso

El parámetro TBK\_URL\_FRACASO corresponde a la URL que contiene a la página que se llamará en el caso que la transacción fracase. No hay ningún requerimiento sobre esta página, basta con que contenga un mensaje apropiado que informe al usuario que la transacción no se realizó.

Ejemplo página de fracaso:

fracaso.asp

```
<HTML>  
<BODY> LA TRANSACCION NO SE PUDO REALIZAR </BODY >  
</HTML>
```

## ANEXO G

### Ejemplo integración de tienda Transacción Normal PHP

A continuación se describe un ejemplo de integración de tienda con tipo de transacción Normal, sobre el ambiente de integración de Webpay. Los pasos a seguir son los siguientes:

- Paso 1: Descomprimiendo KCC 6.0.2
- Paso 2: Configurando archivo de parámetros tbk\_config.dat
- Paso 3: Configurando parámetros de transacción tbk\_param.txt
- Paso 4: Instalando la llave privada del comercio de prueba privada.pem y la pública de Transbank tbk\_public\_key.pem
- Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat
- Paso 6: Páginas de pago, cierre, éxito y fracaso

#### Requerimientos:

- ✓ Contar con un servidor web para ejecución de páginas .PHP, ejemplo APACHE
- ✓ Editor de texto
- ✓ Acceso a Internet con tráfico hacia <https://webpay3gint.transbank.cl> puerto 443

## **Paso 1: Descomprimiendo KCC 6.0.2**

---

El KCC versión 6.0.2 proporcionado por Transbank, es específico para cada de plataforma de sistema operativo (Windows, Linux ó Solaris), es un archivo comprimido con formato rar.

Se debe descomprimir y copiar el contenido de la carpeta CGI-BIN en la ubicación física del disco, en donde residen los CGIs en el servidor Web.

## **Paso 2: Configurando archivo de parámetros tbk\_config.dat**

---

Supuestos del ejemplo:

<b>Código de Comercio</b>	597020000541
<b>Tipo De Transacción</b>	Normal
<b>Dirección Física</b>	/PATH_WWW/TIENDA_TEST
<b>IP Servidor de Comercio</b>	IP_SERVIDOR
<b>Puerto de acceso al Servidor de Comercio</b>	PUERTO_SERVIDOR
<b>URL Servidor de Pago</b>	https://webpay3gint.transbank.cl
<b>Puerto de acceso al Servidor de Pago</b>	443
<b>Plataformas</b>	LINUX
<b>Servidor Web</b>	APACHE
<b>Carpeta de archivos de tienda virtual y CGI-BIN</b>	~/PATH_WWW/TIENDA_TEST



Archivo tbk\_config.dat:

```
IDCOMERCIO      = 597026000541
MEDCOM          = 1
TBK_KEY_ID      = 101
PARAMVERIFCOM   = 1
URLCGICOM       = http://www.comercioFicticio.cl/cgi-
bin/tbk_bp_resultado.cgi
SERVERCOM       = 123.123.123.123
PORTCOM        = 80
WHITELISTCOM    =
ABCDEFGHIJKLMNOSTUVWXYZabcdefghijklmnopqrstuvwxyz 0123456789./:=&?_
HOST           = 123.123.123.123
WPORT          = 80
URLCGITRA       = /filtroUnificado/bp_revision.cgi
URLCGIMEDTRA    = /filtroUnificado/bp_validacion.cgi
SERVERTRA       = https://webpay3gint.transbank.cl
PORTTRA        = 443
PREFIJO_CONF_TR = HTML_
HTML_TR_NORMAL  = http://www.comercioFicticio.cl/webpay/compra.php
```

### Paso 3: Configurando parámetros de transacción tbk\_param.txt

---

Verificar que los parámetros para la transacción normal (TRX\_NORMAL) se encuentren bien configurados

```
BP_PAGO>
  <TR_NORMAL>
    TBK_TIPO_TRANSACCION#A#50#1
    TBK_MONTO#M#10#1
    TBK_ORDEN_COMPRA#A#26#1
    TBK_ID_SESION#A#61#0
    TBK_URL_FRACASO#A#256#1
    TBK_URL_EXITO#A#256#1
    TBK_MONTO_CUOTA#M#9#0
    TBK_NUMERO_CUOTAS#N#2#0
  </TR_NORMAL>
  ...
```

## Paso 4: Instalando la llave privada del comercio de prueba privada.pem y publica de transbank

La llave privada es un archivo de texto que debe ser copiado en la carpeta cgi-bin/maestros/

El contenido del archivo es el siguiente:

```
-----BEGIN RSA PRIVATE KEY-----
MIIIEowIBAQAQEAn3HzPC1ZBzC03edUCf/XJiWj3bzJpjjTi/zB090+DDzZCaMp
14aspXQryvJHv8644E19Q+NHfxtz1cxd2wnSYKvay1gJx30Z1T0AkzUj4QMimR16
vomL1Q3T2MAz1znt/PVPVU7T/JOG9R+EbiHNVKa/hUjWJEFVXLQNME97nHoLjb3v
V5yV2aVhmox7b54n6F3UVPVHCsHKbJpXpE+vnLpVmdETbNpFVrDyGxyG+mnEvyi0
BLIwEY3XTMrGxVs069groLi5Gg8C5LdaYOWjE9084T4fiWGrHhn2781R1rykunTu
77wiWPUQHMS0+YC7mhnSk8Z/iLD+aWz/vhsgHwIDAQABAoIBAQCm+Nrt4cpNKQmn
+Ne8348CGRS9ACXp6WRg6OCXO4zM71RZAminVgZgSQXE6aJR+T9rIWMeG7GwydX
aJGzEEQJZ0jV0MkUr+7mk9qiTOGkGHmGlyHnRQU8jDU59vXe3UEl315+NmwHbQht
waf9F7XLmoLK/WoVJA6tICRpC11oQrpziqN+gjdMpz9i8I1sMFE7+Y7xf+7S2u7
c1MRPUWqgdS9yViQVh3vZi25m5CyKRVn0B0hpNuZ7nrJymtADYSwt9wV2W1fX+MX
UUoYfxyQVwryHhGdedU7GGAAnoEdblUcDkBuAaFmsm1P8K4HQZLWP4v6pYlW2JLa
Zoaerb3BAoGBANCRev10CLB0HBU7sCs0eN9fTkIEsh30VixPSBqDnKsynJrIWovK
cs37Vb6phzdQ03AdoFJvR9ck8+v6Cv0KR8IOF19wFC4ZoxkKBBEq94ZLN+YhE2PW
KiRfYbqCtZxKS3MyWgpIcT9xFTHVjlorZ8Jk51fgLZbGzamtLhderVAoGBAM00
mIiiV412vXzu4tFfkpu/G0x/D9/vAic3X9F0ky09BNCyuMXMQgI8e3wWsGEZgh1s
Vg9KDV5EPxAmumcdPFK2IMACaH41ac7vys3ZD8kMK0INQkuDacG4YsxMaTwEPo0
p1i3zwwEwknw1yJkOyozz0EcIzS9NrZZEjnBHEjAoGAQ81XdeqzvHEyg/CQd6sq
NctubGXMYZYi1C4d2Yi5kKn2YRcK4HDi23V+TWodK+0oNwToZIQKjBvUmn0Bv3rt
EvezbdlMFux+SfCIng0VRJIFTQmpnQYNUxdg2gpwXC/ZWfa6CNxtQABMjFy1cqXM
PJild1IYseJurgBu3mkvBTUCgYBqA/T1X2woLuis2wPIBAV5juXDh3lkB6eU8uxX
CEe2I+3t2EM781B2wajrKadWkmjluMhN9AGV5UZ8S1P0DSUyUywdx1/8RnmZIP
qSWHAGXV9jI0zNr7G4Em0/1eriWkRM26w6fHjLx8EyxDFsohSbkqBrOptCWqoEUx
MOQ5HQKBGAs4sbdd0as2MapuhKU2surEb3Kz3RCIpta4bXgTQmt9wawcZSSpvnfT
zs5sehYvBFszL3MV98Uc50HXMf7gykRCmPRmB9S+f+kiVRvQDHFc9nRNg2XgcotU
KAE16PQM8GihQ0C+EcXHouyud5CRJGfyurokR1H/jY3BiRAG5c+6
-----END RSA PRIVATE KEY-----
```

Al igual que la llave del comercio, la pública de Transbank debe ir en el mismo directorio. El contenido de este archivo es:

```
-----BEGIN CERTIFICATE-----
MIIDKTCcAhECBZ17uIwDQYJKoZIhvcNAQEFBQAwTElMAkGA1UEBhMCQ0wxZDZCaMp
aWx1MREwDwYDVQQHDAhTYW50aWFnbnzEMMAoGA1UECgwDa2R1MQwwCgYDVQQLDANr
BAMMAjEwMB4XDTE1MTIwNzIwNDEwN1oXDTE4MDkzMjIwNDEwN1owTElMAkGA1UEBhMCQ0wxZDZCaMp
BgNVBAGMBUNoawx1MREwDwYDVQQHDAhTYW50aWFnbnzEMMAoGA1UECgwDa2R1MQwwCgYDVQQLDANr
ZHUxCzAJBgNVBAMTAjEwMB4XDTE1MTIwNzIwNDEwN1oXDTE4MDkzMjIwNDEwN1owTElMAkGA1UEBhMCQ0wxZDZCaMp
3jmZpWxfDG9oKyBrU0Bdl6fKiF9a1GrwevThsU5Dq3wiRfYvomStNjFDYFX0s9pRIxqX2AWdybJA
X/+bDdTVbM+xx11A9stJY8s7hxAvvw07IEu0mYDpmlKP7J+4KkNH7yxsKZyLL9trG3isJv6Y6S05
EEHusdxoJFAow/h7qizJW0k0aWRc1jf7kpqJAL3AadIuqV+h1f+Ts/64aMsfSJJ6xDbdp9ddgVF
oqU11M8vpm4g1x1SrYmEkbYwdI9uF2d6bAeaneBPJFZr6KQqlbbrVyeJZqmM1EPy0qPco1TIxrd
EHlXgIFJLyyMRAyJx9i4170xjwIDAQABMA0GCSqGSIb3DQEBBQUAA4IBAQBn3tUPS6e2USgMrPKp
sxU40TfW64+mFD6QrVeB0h81f6aGHa67sMJn8FE/cG6jrUmX/FP1/Cpbpvkm5U11FKpgaFfHv+Kg
CpEvgcRiv/OeIi6Jbuu3NrpDGPwZyKz10Qnmgio5RgB6GSS+OQ0mUwZ9J1+YtdZc+XTga0x7nsCT
5xNcUXsZKhjyokHxtxJm3eyB3ysLNUyL/RHy/EyNEWiUhtv1SIEpNw+Y4/cjQWYwNqSqMzTSW9TP
2QR2bX/W2H6ktRcLsgBK9mq71E36p3q6c9DtZJE+xfA4NGCYWM9hd8pbusnoNO7AFxJZOuuVZ17
JvD7YlhpCYKry7N6x31
-----END CERTIFICATE-----
```

## Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat

Validar la siguiente línea al archivo tbk\_trace.dat:

```
GLEVEL = 7
```

### Página de pago

La llamada al KCC se hace a través del cgi tbk\_bp\_pago el que a su vez debe ser llamado desde la página de pago del comercio. Este archivo tiene como objetivo recoger la información de la compra que se pretende realizar. Así de acuerdo a los parámetros de entrada de una transacción normal, este formulario de pago tiene la siguiente forma:

```
<?php
$TBK_MONTO="10000";
$TBK_ORDEN_COMPRA = date("Ymdhis");
$TBK_ID_SESION = date("Ymdhis")
/***** CONFIGURACION *****/
$TBK_TIPO_TRANSACCION = "TR_NORMAL";
$TBK_URL_EXITO          = "http://IP_SERVIDOR/TIENDA_TEST/html/exito.php";
$TBK_URL_FRACASO        = "http://IP_SERVIDOR/TIENDA_TEST/html/fracaso.php";
$url_cgi                = "http://IP_SERVIDOR/TIENDA_TEST/cgi-bin/tbk_bp_pago.cgi";
//Archivos de datos para uso de pagina de cierre
$myPath
"/PATH_WWW/TIENDA_TEST/comun/dato$TBK_ID_SESION.log";
/***** FIN CONFIGURACION *****/
//formato Moneda
$partesMonto=split("",$TBK_MONTO);
$TBK_MONTO=$partesMonto[0]."00";
//Grabado de datos en archivo de transaccion
$fic = fopen($myPath, "w+");
$linea="$TBK_MONTO;$TBK_ORDEN_COMPRA";
fwrite ($fic,$linea);
fclose($fic);
?>
<html>
<body >
  <form action="<?php echo $url_cgi;?>" name="frm" method="post">
    <table>
      <tr>
        <th align="right">Tipo Transaccion:</th>
        <td><input type="text" name="TBK_TIPO_TRANSACCION" value="<?php echo $TBK_TIPO_TRANSACCION;?>" /></td>
      </tr>
      <tr>
        <th align="right">Monto:</th>
        <td><input type="text" name="TBK_MONTO" value="<?php echo $TBK_MONTO;?>" /></td>
      </tr>
      <tr>
        <th align="right">Orden de Compra:</th>
        <td><input type="text" name="TBK_ORDEN_COMPRA" value="<?php echo $TBK_ORDEN_COMPRA;?>" /></td>
      </tr>
      <tr>
        <th align="right">TBK_ID_SESSION:</th>
        <td><input type="text" name="TBK_ID_SESION" value="<?php echo $TBK_ID_SESION;?>" /></td>
      </tr>
      <tr>
        <th align="right">Pagina de EXITO:</th>
        <td><input type="text" name="TBK_URL_EXITO" value="<?php echo $TBK_URL_EXITO;?>" /></td>
      </tr>
      <tr>
        <th align="right">Pagina de Fracaso:</th>
        <td><input type="text" name="TBK_URL_FRACASO" value="<?php echo $TBK_URL_FRACASO;?>" /></td>
      </tr>
      <tr>
        <td align="right" colspan="2"><input type="submit"></td>
      </tr>
    </table>
  </form>
</body>
```

## Página De Cierre

Esta página extrae los parámetros de respuesta que entrega Transbank, realiza validación de MAC y si corresponde confirma la transacción entregando como respuesta "ACEPTADO". Para simplificar el ejemplo, se ha omitido las validaciones de monto y orden de compra que dependen de cada implementación del comercio.

xt\_compra.php

```
<?php
//rescate de datos de POST.
$TBK_RESPUESTA=$ _POST["TBK_RESPUESTA"];
$TBK_ORDEN_COMPRA=$ _POST["TBK_ORDEN_COMPRA"];
$TBK_MONTO=$ _POST["TBK_MONTO"];
$TBK_ID_SESION=$ _POST["TBK_ID_SESION"];
/***** CONFIGURAR AQUI *****/
$myPath = "/PATH_WWW/TIENDA_TEST/comun/dato$TBK_ID_SESION.log";
//GENERA ARCHIVO PARA MAC
$filename_txt = "/PATH_WWW/TIENDA_TEST/comun/MAC01Normal$TBK_ID_SESION.txt";
// Ruta Checkmac
$cmdline = "/PATH_WWW/TIENDA_TEST/cgi-bin/tbk_check_mac.cgi $filename_txt";
/***** FIN CONFIGURACION *****/
$accepta=false;
//lectura archivo que guardo pago.php
if ($fic = fopen($myPath, "r")){
    $linea=fgets($fic);
    fclose($fic);
}
$detalle=split(";", $linea);
if (count($detalle)>=1){
    $monto=$detalle[0];
    $ordenCompra=$detalle[1];
}
//guarda los datos del post uno a uno en archivo para la ejecución del MAC
$f=fopen($filename_txt,"wt");
while(list($key, $val)=each($_POST)){
    fwrite($f, "$key=$val&");
}
fclose($f);
//Validación de respuesta de Transbank, solo si es 0 continua con la pagina de cierre
if($TBK_RESPUESTA=="0"){ $accepta=true; } else { $accepta=false; }
//validación de monto y Orden de compra
if ($TBK_MONTO==$monto && $TBK_ORDEN_COMPRA==$ordenCompra && $accepta==true){ $accepta=true;}
else{ $accepta=false;}
//Validación MAC
if ($accepta==true){
    exec ($cmdline, $result, $retint);
    if ($result [0] == "CORRECTO") $accepta=true; else $accepta=false;
}
?>
<html>
    <?php if ($accepta==true){?>
        ACEPTADO
    <?php } else {?>
        RECHAZADO
    <?php }?>
</html>
```

### Nota:

En el ejemplo se hace uso del archivo `tbk_check_mac.cgi`, el cual se entrega con el KCC 6.0.2 y permite la validación del MAC (CORRECTO - INVALIDO).

## Página de éxito

El parámetro TBK\_URL\_EXITO corresponde a la URL que contiene a la página que se llamará en caso que la transacción se haya ejecutado correctamente. La información puede ser desplegada recuperando el valor del parámetro TBK\_ORDEN\_DE\_COMPRA.

Ejemplo página éxito:

exitos.php

```
<?php
    $TBK_ID_SESION          = $_POST["TBK_ID_SESION"];
    $TBK_ORDEN_COMPRA       = $_POST["TBK_ORDEN_COMPRA"];
    /***** CONFIGURAR AQUI *****/
    $myPath                  =
"/PATH_WWW/TIENDA_TEST/comun/MAC01Normal$TBK_ID_SESION.txt";
    $pathSubmit              = "http://IP_SERVIDOR/TIENDA_TEST/html/index.php";
    /***** FIN CONFIGURACION *****/

    //Rescate de los valores informados por transbank
    $fic = fopen($myPath, "r");
    $linea=fgets($fic);
    fclose($fic);
    $detalle=explode("&", $linea);
    $TBK_ORDEN_COMPRA=explode("=", $detalle[0]);
    $TBK_TIPO_TRANSACCION=explode("=", $detalle[1]);
    $TBK_RESPUESTA=explode("=", $detalle[2]);
    $TBK_MONTO=explode("=", $detalle[3]);
    $TBK_CODIGO_AUTORIZACION=explode("=", $detalle[4]);
    $TBK_FINAL_NUMERO_TARJETA=explode("=", $detalle[5]);
    $TBK_FECHA_CONTABLE=explode("=", $detalle[6]);
    $TBK_FECHA_TRANSACCION=explode("=", $detalle[7]);
    $TBK_HORA_TRANSACCION=explode("=", $detalle[8]);
    $TBK_ID_TRANSACCION=explode("=", $detalle[10]);
    $TBK_TIPO_PAGO=explode("=", $detalle[11]);
    $TBK_NUMERO_CUOTAS=explode("=", $detalle[12]);
    $TBK_MAC=explode("=", $detalle[13]);
    $TBK_FECHA_CONTABLE[1]=substr($TBK_FECHA_CONTABLE[1],2,2)."-
".substr($TBK_FECHA_CONTABLE[1],0,2);
    $TBK_FECHA_TRANSACCION[1]=substr($TBK_FECHA_TRANSACCION[1],2,2)."-
".substr($TBK_FECHA_TRANSACCION[1],0,2);
    $TBK_HORA_TRANSACCION[1]=substr($TBK_HORA_TRANSACCION[1],0,2).":".substr($TBK_HORA_TRANSACCION[1],2,2).":".substr($TBK_HORA_TRANSACCION[1],4,2);
    ?>
<html>
    <head><title>TIENDA PHP KCC6.0.2</title></head>
    <body>
        <P ALIGN="CENTER">Transaccion Normal - EXITO</P><br>
        <table align="center">
            <tr><td align=left width=100% >El pago se ha realizado con &acute;xito.</td></tr>
            <tr><td align=left width=100% >Datos de la Compra:</td>
            </tr>
            <tr>
                <td align=left width=100% >
                    <table>
                        <tr valign=top>
                            <th align=left>Nro Orden:</th>
                            <td align=left><?php echo
$TBK_ORDEN_COMPRA[1];?></td>
                        <tr>
                            <th align=left>Monto (pesos chilenos):</th>
                            <td align=left><?php echo
($TBK_MONTO[1]/100); ?></td>
                        </tr>
                    </table>
                </td>
            </tr>
        </table>
    </body>
</html>
```

```

<tr><td align=left width=100%><h3>Datos de la Transacci&oacute;n:</h3></td>
</tr>
<tr>
<td align=left width=100%>
<table width=90% border=0 align=center>
<tr valign=top >
<th align=left>Respuesta:
Transacci&oacute;n:</th>
<td align=left><?php echo
$TBK_RESPUESTA[1];?></td>
<th align=left>Codigo:
Autorizaci&oacute;n:</th>
<td align=left><?php echo
$TBK_CODIGO_AUTORIZACION[1];?></td>
<th align=left>Fecha Contable:</th>
<td align=left><?php echo
$TBK_FECHA_CONTABLE[1];?></td>
</tr>
<tr valign=top >
<th align=left>Fecha:
Transacci&oacute;n:</th>
<td align=left><?php echo
$TBK_FECHA_TRANSACCION[1];?></td>
<th align=left>Hora:
Transacci&oacute;n:</th>
<td align=left><?php echo
$TBK_HORA_TRANSACCION[1];?></td>
<th align=left>tarjeta de
cr&eacute;dito:</th>
<td align=left>XXXXXXXXXXXX<?php echo
$TBK_FINAL_NUMERO_TARJETA[1];?></td>
</tr>
<tr valign=top>
<th align=left>Tipo:
Transacci&oacute;n:</th>
<td align=left><?php echo
$TBK_TIPO_TRANSACCION[1];?></td>
<th align=left>Tipo Pago:</th>
<td align=left><?php echo
$TBK_TIPO_PAGO[1];?></td>
<th align=left>Numero cuotas:</th>
<td align=left><?php echo
$TBK_NUMERO_CUOTAS[1];?></td>
</tr>
</table>
</td>
</tr>
</table>
<FORM ACTION="<?php echo $pathSubmit;?>" METHOD=POST>
<center><INPUT TYPE="SUBMIT" VALUE="VOLVER A INICIO"></center>
</form>
</body>
</html>

```

## Página de fracaso

El parámetro TBK\_URL\_FRACASO corresponde a la URL que contiene a la página que se llamará en el caso que la transacción fracase. No hay ningún requerimiento sobre esta página, basta con que contenga un mensaje apropiado que informe al usuario que la transacción no se realizó.

Ejemplo página de fracaso:

fracaso.php

```
<?PHP
/***** CONFIGURAR AQUI *****/
$PATHSUBMIT = "HTTP://IP_SERVIDOR/TIENDA_TEST/HTML/INDEX.PHP";
/***** FIN CONFIGURACION *****/
$TBK_ID_SESION = $_POST["TBK_ID_SESION"];
$TBK_ORDEN_COMPRA = $_POST["TBK_ORDEN_COMPRA"];
?>
<HTML>
<HEAD><TITLE>TIENDA PHP KCC6.0.2</TITLE></HEAD>
<BODY>
<CENTER>
<B>TRANSACCION FRACASADA !!!</B>
<TABLE>
<TR><TH>FRACASO</TH></TR>
<TR><TD>
TBK_ID_SESION=<?PHP ECHO $TBK_ID_SESION; ?><BR>
TBK_ORDEN_COMPRA=<?PHP ECHO $TBK_ORDEN_COMPRA; ?><BR>
</TD></TR>
</TABLE>
</CENTER>
<FORM ACTION="<?PHP ECHO $PATHSUBMIT; ?>" METHOD=POST>
<CENTER>
<INPUT TYPE="SUBMIT" VALUE="VOLVER A INICIO">
</CENTER>
</FORM>
</BODY>
</HTML>
```

## ANEXO H

### Ejemplo integración de tienda Transacción Normal Completa PHP

A continuación se describe un ejemplo de integración de tienda PHP con tipo de transacción Normal Completa, sobre el ambiente de integración de Webpay. Los pasos a seguir son los siguientes:

Paso 1: Descomprimir KCC 6.0.2

Paso 2: Configurar archivo de parámetros tbk\_config.dat

Paso 3: Configurar parámetros de transacción tbk\_param.txt

Paso 4: Instalar llave privada del comercio de pruebas privada.pem y la llave pública de Transbank tbk\_public\_key.pem

Paso 5: Configurar el nivel de log en archivo tbk\_trace.dat

Paso 6: Crear páginas de pago, cierre, éxito y fracaso

#### Requerimientos:

- ✓ Contar con un servidor web para ejecución de páginas PHP con soporte CGI. Ejemplo Apache 2 y PHP5.
- ✓ Editor de texto.
- ✓ Servidor web con acceso a Internet y tráfico específicamente hacia la url <https://webpay3gint.transbank.cl>, puerto 443.
- ✓ Acceso desde Transbank a la cgi tbk\_bp\_resultado.cgi por puerto 80.



## **Paso 1: Descomprimiendo KCC 6.0.2**

---

El KCC versión 6.0.2 proporcionado por Transbank, es específico para cada de plataforma de sistema operativo (Windows, Linux o Solaris), es un archivo comprimido con formato rar.

Se debe descomprimir y copiar todo el contenido de la carpeta cgi-bin en la ubicación para ejecución de CGIs que esté configurada en el servidor Web.

## **Paso 2: Configurando archivo de parámetros tbk\_config.dat**

---

Supuestos del ejemplo:

Código de Comercio	597026000507
Tipo De Transacción	Normal Completa
Dirección Física	/PATH_WWW/TIENDA_TEST
IP Servidor de Comercio	IP_SERVIDOR
Puerto de acceso al Servidor de Comercio	PUERTO_SERVIDOR
URL Servidor de Pago	https://webpay3gint.transbank.cl
Puerto de acceso al Servidor de Pago	443
Plataformas	LINUX
Servidor Web	APACHE
Carpeta de archivos de tienda virtual y CGI-BIN	/PATH_WWW/TIENDA_TEST/CGI-BIN

Archivo tbk\_config.dat:

```
IDCOMERCIO = 597026000507
MEDCOM = 1
TBK_KEY_ID = 101
PARAMVERIFCOM = 1
URLCGICOM = http://www.comercioFicticio.cl/cgi-bin/tbk_bp_resultado.cgi
SERVERCOM = 123.123.123.123
PORTCOM = 80
WHITELISTCOM = ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz0123456789./:=&?_
HOST = 123.123.123.123
WPORT = 80
URLCGITRA = /filtroUnificado/bp_revision.cgi
URLCGIMEDTRA = /filtroUnificado/bp_validacion.cgi
SERVERTRA = https://webpay3gint.transbank.cl
PORTTRA = 443
PREFIJO_CONF_TR = HTML_
HTML_TR_COMPLETA = http://www.comercioFicticio.cl/webpay/xt_compra.php
```

Tener las siguientes precauciones en la creación de este archivo:

- No debe tener espacios al final de cada parámetro.
- No debe tener tabuladores.

### Paso 3: Configurando parámetros de transacción tbk\_param.txt

---

Verificar que los parámetros para la transacción normal completa (TRX\_COMPLETA) se encuentren bien configurados

```
<BP_PAGO>
...
  <TR_COMPLETA>
    TBK_TIPO_TRANSACCION#A#50#1
    TBK_MONTO#M#10#1
    TBK_ORDEN_COMPRA#A#26#1
    TBK_ID_SESION#A#61#0
    TBK_URL_EXITO#A#256#1
    TBK_URL_FRACASO#A#256#1
    TBK_NUMERO_TARJETA#N#19#1
    TBK_FECHA_EXPIRACION#N#4#1
    TBK_CVV#N#4#1
    TBK_TIPO_PAGO#A#2#1
    TBK_NUMERO_CUOTAS#N#2#0
    TBK_MONTO_CUOTA#M#9#0
  </TR_COMPLETA>
...
</BP_PAGO>
```

#### **Paso 4: Instalando la llave privada del comercio y la llave pública de Transbank**

---

Las llaves pública (tbk\_public\_key.pem o tbk\_public\_key101prod.pem) y privada (privada.pem) son archivos de texto que vienen dentro del kit en la carpeta /cgi-bin/maestros/ y que por lo tanto ya deberían estar copiadas junto con todo el contenido de cgi-bin en la carpeta destinada por el servidor web para ejecución de CGI.

De las dos llaves públicas que vienen en la carpeta maestros se utiliza una sola. La elección dependerá del valor del parámetro TBK\_KEY\_ID configurado en el archivo tbk\_config.dat:

Si TBK\_KEY\_ID = 100, se debe utilizar la llave pública tbk\_public\_key.pem, es decir, no hacer ningún cambio respecto de lo que venía dentro del kit.

Si TBK\_KEY\_ID = 101 (como es el caso de este ejemplo), se debe utilizar la llave pública tbk\_public\_key101prod.pem pero con el nombre tbk\_public\_key.pem. Esto es, eliminar tbk\_public\_key.pem y renombrar tbk\_public\_key101prod.pem a tbk\_public\_key.pem.

#### **Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat**

---

Validar la siguiente línea del archivo cgi-bin/datos/tbk\_trace.dat:

```
GLEVEL = 7
```

## Paso 6: Páginas de pago, cierre, éxito y fracaso

### Página de pago

La conexión con el KCC se hace a través del archivo `cgi-bin/tbk_bp_pago.cgi`, que debe ser llamado desde la página de pago del comercio. Este archivo tiene como objetivo recoger la información de la compra que se pretende realizar.

De acuerdo a los parámetros de entrada de una transacción **normal completa**, este formulario de pago tiene la siguiente forma:

```
<?php
/*
 * Valores de Ejemplo.
 * Estos valores, deben obtenerse del flujo de Transacción.
 */

// Transacción normal Completa
$TBK_TIPO_TRANSACCION = "TR_COMPLETA";

// Monto de la transacción.
$TBK_MONTO = "10000";

// Orden de compra. Este valor debe ser único.
$TBK_ORDEN_COMPRA = uniqid("oc");

// Identificador de sesión
$TBK_ID_SESION = date("Ymdhis");

// Página a la cuál se redireccionará al tarjeta habiente, en caso de éxito.
$TBK_URL_EXITO = "http://www.comercioFicticio.cl/webpay/exito.php";

// Página a la cuál se redireccionará al tarjeta habiente, en caso de fracaso.
$TBK_URL_FRACASO = "http://www.comercioFicticio.cl/webpay/fracaso.php";

// Número de Tarjeta.
$TBK_NUMERO_TARJETA = 4051885600446623;

// Fecha de expiración de la Tarjeta formato mmaa, ejemplo: 0120
$TBK_FECHA_EXPIRACION = date("my");

// Código de verificación del tarjeta.
$TBK_CVV = 123;

// Tipo de pago: VN (Venta Normal), VC (Venta Cuotas), SI (Cuotas Sin Interés), CI (Cuotas Comercio)
$TBK_TIPO_PAGO = "VN";

// Número de cuotas, obligatorio en caso de venta en cuotas.
$TBK_NUMERO_CUOTAS = "";

// Monto de cada cuota, obligatorio en caso de venta en cuotas.
$TBK_MONTO_CUOTA = "";

// Url de la cgi de pago
$url_cgi = "http://www.comercioFicticio.cl/cgi-bin/tbk_bp_pago.cgi";

// Formateo de Moneda
if (!empty($TBK_MONTO)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO .= "00";
}

if (!empty($TBK_MONTO_CUOTA)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO_CUOTA .= "00";
}

/**
 * En necesario, almacenar los datos de transacción, en algún mecanismo persistente,
 * para su consulta posterior. Se asume la existencia de una función almacenar_transaccion
 * creada para dicho propósito, que retorna true en caso de éxito y false en cualquier otro caso.
 * Cada integración, debe crear un método de persistencia para la transacción.
 * Se deben almacenar al menos, la orden de compra y el monto.
 */
$ok = almacenar_transaccion($TBK_ORDEN_COMPRA, $TBK_MONTO);
```

```

if ($ok) {
?>
<html>
<body>
<form name="frmPago" method="POST" action="<?php echo $url_cgi; ?>">
<table border=0 align="center">
<tr>
<td><input type="text" name="TBK_TIPO_TRANSACCION" value="<?php echo $TBK_TIPO_TRANSACCION; ?>"
/></td>
<td><input type="text" name="TBK_MONTO" value="<?php echo $TBK_MONTO; ?>" /></td>
</tr>
<tr>
<td><input type="text" name="TBK_ORDEN_COMPRA" value="<?php echo $TBK_ORDEN_COMPRA; ?>" /></td>
<td><input type="text" name="TBK_ID_SESION" value="<?php echo $TBK_ID_SESION; ?>" /></td>
</tr>
<tr>
<td><input type="text" name="TBK_URL_EXITO" SIZE="30" value="<?php echo $TBK_URL_EXITO; ?>" /></td>
<td><input type="text" name="TBK_URL_FRACASO" SIZE="30" value="<?php echo $TBK_URL_FRACASO; ?>" /></td>
</tr>
<tr>
<td><input type="text" name="TBK_NUMERO_TARJETA" value="<?php echo $TBK_NUMERO_TARJETA; ?>" /></td>
<td><input type="text" name="TBK_FECHA_EXPIRACION" value="<?php echo $TBK_FECHA_EXPIRACION; ?>"
/></td>
</tr>
<tr>
<td><input type="text" name="TBK_CVV" value="<?php echo $TBK_CVV; ?>" /></td>
<td><input type="text" name="TBK_TIPO_PAGO" value="<?php echo $TBK_TIPO_PAGO; ?>" /></td>
</tr>
<tr>
<td><input type="text" name="TBK_NUMERO_CUOTAS" value="<?php echo $TBK_NUMERO_CUOTAS; ?>" /></td>
<td><input type="text" name="TBK_MONTO_CUOTA" value="<?php echo $TBK_MONTO_CUOTA; ?>" /></td>
</tr>
<tr>
<td align="center" colspan="2">
<input type="submit" />
</td>
</tr>
</table>
</form>
</body>
</html>
<?php
} else {
?>
<html>
<body>
<script type="text/javascript">
window.location = <?php echo $TBK_URL_FRACASO; ?>;
</script>
</body>
</html>
<?php
}
?>

```

## Página De Cierre

Esta página extrae los parámetros de respuesta que entrega Transbank, realiza validación de seguridad (MAC) y si corresponde, confirma la transacción entregando como respuesta "ACEPTADO". Para simplificar el ejemplo, se han realizado supuestos para las validaciones de monto y orden de compra. Es importante notar que estas validaciones dependen de la lógica de negocios de cada comercio.

### xt\_compra.php

```
<?php

// Iniciamos la variable, asumiendo 'RECHAZADO'.
$ack = "RECHAZADO";

try {
    /*
     * Recate de datos por POST, enviados desde TBK
     * Si al recibir TBK_RESPUESTA la variable es nula la pongo en 0. con esto se caerá por validación de check_mac
     */
    $TBK_RESPUESTA = (int) trim(strip_tags($_POST["TBK_RESPUESTA"]));
    $TBK_ORDEN_COMPRA = trim(strip_tags($_POST["TBK_ORDEN_COMPRA"]));
    $TBK_MONTO = (int) trim(strip_tags($_POST["TBK_MONTO"]));
    $TBK_ID_SESION = trim(strip_tags($_POST["TBK_ID_SESION"]));
    $TBK_TIPO_TRANSACCION = trim(strip_tags($_POST["TBK_TIPO_TRANSACCION"]));
    $TBK_FINAL_NUMERO_TARJETA = trim(strip_tags($_POST["TBK_FINAL_NUMERO_TARJETA"]));
    $TBK_FECHA_CONTABLE = trim(strip_tags($_POST["TBK_FECHA_CONTABLE"]));
    $TBK_FECHA_TRANSACCION = trim(strip_tags($_POST["TBK_FECHA_TRANSACCION"]));
    $TBK_HORA_TRANSACCION = trim(strip_tags($_POST["TBK_HORA_TRANSACCION"]));
    $TBK_ID_TRANSACCION = trim(strip_tags($_POST["TBK_ID_TRANSACCION"]));
    $TBK_CODIGO_AUTORIZACION = trim(strip_tags($_POST["TBK_CODIGO_AUTORIZACION"]));
    $TBK_TIPO_PAGO = trim(strip_tags($_POST["TBK_TIPO_PAGO"]));
    $TBK_NUMERO_CUOTAS = trim(strip_tags($_POST["TBK_NUMERO_CUOTAS"]));
    $TBK_VCI = trim(strip_tags($_POST["TBK_VCI"]));
    $TBK_TASA_INTERES_MAX = trim(strip_tags($_POST["TBK_TASA_INTERES_MAX"]));
    $TBK_MAC = trim(strip_tags($_POST["TBK_MAC"]));

    /*
     * Se asume la existencia de una función que permita actualizar los datos de la transacción
     * en un mecanismos de persistencia (En nuestro ejemplo la llamaremos actualizar_transaccion)
     * Esta función debería retornar true en caso de éxito y false en cualquier otro caso.
     */
    actualizar_transaccion($TBK_ORDEN_COMPRA, $TBK_TIPO_TRANSACCION, $TBK_RESPUESTA, $TBK_MONTO,
        $TBK_CODIGO_AUTORIZACION, $TBK_FINAL_NUMERO_TARJETA, $TBK_FECHA_CONTABLE,
        $TBK_FECHA_TRANSACCION, $TBK_HORA_TRANSACCION, $TBK_ID_SESION, $TBK_ID_TRANSACCION,
        $TBK_TIPO_PAGO, $TBK_NUMERO_CUOTAS, $TBK_TASA_INTERES_MAX, $TBK_VCI, $TBK_MAC);

    /*
     * se asume que existe una función (en nuestro ejemplo: obtener_transaccion)
     * que permite obtener los datos de la transacción, desde el almacenamiento
     * persistente usando la orden de compra, en este ejemplo la función retorna
     * un array asociativo con los datos de la transacción o false en cualquier
     * otro caso
     */
    $transaccion = obtener_transaccion($TBK_ORDEN_COMPRA);

    // Verifico que la orden de compra exista en el almacenamiento persistente
    if ($transaccion !== false) {
        $monto = $transaccion["TBK_MONTO"];
        $acepta = false;

        /*
         * Continuo la Validación, si el monto obtenido desde el almacenamiento
         * persistente, es igual al monto notificado por Webpay.
         */
        if ($monto == $TBK_MONTO) {
            /*
             * Si el TBK_RESPUESTA es igual a 0, implica que webpay aprueba la
             * transacción, un valor distinto implica rechazo de la transacción en
             * webpay.
             */
            if ($TBK_RESPUESTA == 0) {
                $data = $_POST;
                /*
                 * Genero el archivo de validación checkmac
                 * Es importante que la carpeta tenga permisos de escritura.
                 */
                $archivo_checkmac = "/tmp/mac_tr_completa_{$TBK_ORDEN_COMPRA}.txt";
                $fp = fopen($archivo_checkmac, "w+");
            }
        }
    }
}
```

```

        foreach ($data as $key => $val) {
            fwrite($fp, "$key=$val&");
        }
        fclose($fp);
        //ejecución del check_mac
        $cmd = "/PATH_WWW/TIENDA_TEST/CGI-BIN/tbk_check_mac.cgi $archivo_checkmac";
        exec($cmd, $resultado, $retint);
        // Después de realizar la comprobación, elimino el archivo de checkmac
        unlink($archivo_checkmac);

        /*
        * Si el resultado obtenido de la ejecución de la CheckMac es 'CORRECTO'
        * Aceptamos la Transacción de WebPay
        */
        if (trim($resultado[0]) == "CORRECTO") {
            $accepta = true;
        }
    } else {
        /*
        * Si TBK_RESPUESTA es distinto de 0, la transacción es rechazada por Webpay
        * Pero es aceptada por la tienda.
        */
        $accepta = true;
    }
}

$ack = $accepta ? "ACEPTADO" : "RECHAZADO";
}
} catch (Exception $e) {
    error_log($e->getMessage());
    $ack = "RECHAZADO";
}
}

/*
* Si la tienda acepta la transacción, respondemos 'ACEPTADO', si no, responderemos
* 'RECHAZADO' en cualquier otro caso.
* Es importante recalcar que la respuesta no puede tener ningún otro carácter fuera de la palabra ACEPTADO o RECHAZADO
* Ni siquiera debe incluir código HTML.
*/

echo ($ack);
N?>

```

En el ejemplo se hace uso del archivo `cgi-bin/tbk_check_mac.cgi`, incluido dentro del kit. Esta CGI realiza la validación de MAC (CORRECTO - INVALIDO).

## Página de éxito

El parámetro TBK\_URL\_EXITO corresponde a la URL que se llamará en caso que la transacción se haya llevado a cabo correctamente. La información de la transacción puede ser desplegada recuperando el valor del parámetro TBK\_ORDEN\_DE\_COMPRA.

Ejemplo página éxito:

exitos.php

```
<?php

/*
 * Obtención de datos por POST, enviados desde TBK
 */
$TBK_ORDEN_COMPRA = trim(strip_tags($_POST["TBK_ORDEN_COMPRA"]));

/*
 * se asume que existe una función (en nuestro ejemplo: obtener_transacción)
 * que permite obtener los datos de la transacción, desde el almacenamiento
 * persistente usando la orden de compra, en este ejemplo la función retorna
 * un array asociativo con los datos de la transacción o false en cualquier
 * otro caso
 */
$transaccion = obtener_transaccion($TBK_ORDEN_COMPRA);

if ($transaccion !== false) {
    $TBK_TIPO_TRANSACCION = $transaccion['TBK_TIPO_TRANSACCION'];

    $TBK_RESPUESTA = $transaccion['TBK_RESPUESTA'];
    $TBK_MONTO = $transaccion['TBK_MONTO'];
    $TBK_CODIGO_AUTORIZACION = $transaccion['TBK_CODIGO_AUTORIZACION'];
    $TBK_FINAL_NUMERO_TARJETA = $transaccion['TBK_FINAL_NUMERO_TARJETA'];
    $TBK_FECHA_CONTABLE = $transaccion['TBK_FECHA_CONTABLE'];
    $TBK_FECHA_TRANSACCION = $transaccion['TBK_FECHA_TRANSACCION'];
    $TBK_HORA_TRANSACCION = $transaccion['TBK_HORA_TRANSACCION'];
    $TBK_ID_TRANSACCION = $transaccion['TBK_ID_TRANSACCION'];
    $TBK_TIPO_PAGO = $transaccion['TBK_TIPO_PAGO'];
    $TBK_NUMERO_CUOTAS = $transaccion['TBK_NUMERO_CUOTAS'];
    $TBK_MONTO_CUOTA = $transaccion['TBK_MONTO_CUOTA'];
    $TBK_TASA_INTERES_MAX = $transaccion['TBK_TASA_INTERES_MAX'];
    $TBK_VCI = $transaccion['TBK_VCI'];
    $TBK_MAC = $transaccion['TBK_MAC'];

    $TBK_FECHA_TRANSACCION = substr($TBK_FECHA_TRANSACCION, 2, 2) . "/" .
    substr($TBK_FECHA_TRANSACCION, 0, 2) . "/" . date("Y") . " " . substr($TBK_HORA_TRANSACCION, 0, 2) . ":" .
    substr($TBK_HORA_TRANSACCION, 2, 2) . ":" . substr($TBK_HORA_TRANSACCION, 4, 2);

    ?>

<html>
<head>
<title>TRANSACCION EXITOSA</title>
<style type="text/css">
    body {
        color: white;
    }
</style>
</head>
```



```

<body bgcolor="#3069C6">
  <table width='750' border='1' align="center">
    <tr>
      <td align='left' width='100%'>
        El pago se ha realizado con éxito.
      </td>
    </tr>
    <tr>
      <td align='left' width='100%'>
        <h3>Datos de la Compra:</h3>
      </td>
    </tr>
    <tr>
      <td align='left' width='100%'>
        <table width='90%' align='center' border='0'>
          <tr valign='top'>
            <th align='left'>Nro Orden:</th>
            <td align='left'><?php echo $TBK_ORDEN_COMPRA; ?></td>
            <td width='30%'>&nbsp;</td>
            <th align='left'>Monto:</th>
            <td align='left'><?php echo ($TBK_MONTO / 100); ?></td>
          </tr>
        </table>
      </td>
    </tr>
    <tr>
      <td align='left' width='100%'>
        <h3>Datos de la Transacción:</h3>
      </td>
    </tr>
    <tr>
      <td align='left' width='100%'>
        <table width='90%' border='0' align='center'>
          <tr valign='top'>
            <th align='left'>Fecha Transacción:</th>
            <td align='left'><?php echo $TBK_FECHA_TRANSACCION; ?></td>
            <td width='30%'>&nbsp;</td>
            <th align='left'>tarjeta de crédito:</th>
            <td align='left'>XXXXXXXXXXXX<?php echo $TBK_FINAL_NUMERO_TARJETA; ?></td>
          </tr>
          <tr valign='top'>
            <th align='left'>Tipo Transacción:</th>
            <td align='left'><?php echo $TBK_TIPO_TRANSACCION; ?></td>
            <td width='30%'>&nbsp;</td>
          </tr>
        </table>
      </td>
    </tr>
  </table>
  <br />
  <p><a href="index.php">Volver a la tienda</a></p>
</body>
</html>

```

## Página de fracaso

El parámetro TBK\_URL\_FRACASO corresponde a la URL que se llamará en el caso que la transacción no se pueda llevar a cabo. No hay ningún requerimiento sobre esta página, basta con que contenga un mensaje apropiado que informe al usuario que la transacción no se realizó.

### Ejemplo página de fracaso:

fracaso.php

```
<?php
$order = trim(strip_tags($_REQUEST['TBK_ORDEN_COMPRA']));
if (empty($order)) {
    $order = trim(strip_tags($_REQUEST['TBK_ID_SESION']));
}
?>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf8" />
  </head>
  <body>
    <h1>Error</h1>
    <p>
      La transacción <?php echo $order; ?> fue rechazada.
    </p>
    <div class="codigo">
      <ul>
        <li>Error en el ingreso de los datos de su tarjeta de crédito (fecha y/o código de seguridad).</li>
        <li>Su tarjeta de crédito no cuenta con el cupo necesario para cancelar la compra.</li>
        <li>Tarjeta aún no habilitada en el sistema financiero.</li>
      </ul>
    </div>
    <br />
    <p><a href="index.php">Volver a la tienda</a></p>
  </body>
</html>
```

## ANEXO I

### Ejemplo integración de tienda Transacción Mall Completa PHP

A continuación se describe un ejemplo de integración de tienda PHP con tipo de transacción Mall Completa, sobre el ambiente de integración de Webpay. Los pasos a seguir son los siguientes:

- Paso 1: Descomprimir KCC 6.0.2
- Paso 2: Configurar archivo de parámetros tbk\_config.dat
- Paso 3: Configurar parámetros de transacción tbk\_param.txt
- Paso 4: Instalar llave privada del comercio de pruebas privada.pem y la llave pública de Transbank tbk\_public\_key.pem
- Paso 5: Configurar el nivel de log en archivo tbk\_trace.dat
- Paso 6: Crear páginas de pago, cierre, éxito y fracaso

#### Requerimientos:

- ✓ Contar con un servidor web para ejecución de páginas PHP con soporte CGI. Ejemplo Apache 2 y PHP5.
- ✓ Editor de texto.
- ✓ Servidor web con acceso a Internet y tráfico específicamente hacia la url <https://webpay3g.transbank.cl>, puerto 443.
- ✓ Acceso desde Transbank a la cgi tbk\_bp\_resultado.cgi por puerto 80.

## **Paso 1: Descomprimiendo KCC 6.0.2**

---

El KCC versión 6.0.2 proporcionado por Transbank, es específico para cada de plataforma de sistema operativo (Windows, Linux o Solaris), es un archivo comprimido con formato rar.

Se debe descomprimir y copiar todo el contenido de la carpeta cgi-bin en la ubicación para ejecución de CGIs que esté configurada en el servidor Web.

## **Paso 2: Configurando archivo de parámetros tbk\_config.dat**

---

Supuestos del ejemplo:

Código de Comercio del Mall	597020000504
Código de Comercio Tienda 01:	597020000505
Código de Comercio Tienda 02:	597020000506
Tipo De Transacción	Mall Completa
Dirección Física	/PATH_WWW/TIENDA_TEST
IP Servidor de Comercio	IP_SERVIDOR
Puerto de acceso al Servidor de Comercio	PUERTO_SERVIDOR
URL Servidor de Pago	https://webpay3gint.transbank.cl
Puerto de acceso al Servidor de Pago	443
Plataformas	LINUX
Servidor Web	APACHE
Carpeta de archivos de tienda virtual y CGI-BIN	/PATH_WWW/TIENDA_TEST/CGI-BIN

Archivo tbk\_config.dat:

```
IDCOMERCIO = 597020000504
MEDCOM = 1
TBK_KEY_ID = 101
PARAMVERIFCOM = 1
URLCGICOM = http://www.comercioFicticio.cl/cgi-bin/tbk_bp_resultado.cgi
SERVERCOM = 123.123.123.123
PORTCOM = 80
WHITELISTCOM = ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz0123456789./:=&?_
HOST = 123.123.123.123
WPORT = 80
URLCGITRA = /filtroUnificado/bp_revision.cgi
URLCGIMEDTRA = /filtroUnificado/bp_validacion.cgi
SERVERTRA = https://webpay3gint.transbank.cl
PORTTRA = 443
PREFIJO_CONF_TR = HTML_
HTML_TR_MALL_COMPLETA = http://www.comercioFicticio.cl/webpay/xt_compra.php
```

Tener las siguientes precauciones en la creación de este archivo:

- No debe tener espacios al final de cada parámetro.
- No debe tener tabuladores.

### **Paso 3: Configurando parámetros de transacción tbk\_param.txt**

---

Verificar que los parámetros para la transacción **mall completa** (TRX\_MALL\_COMPLETA) se encuentren bien configurados

```
<BP_PAGO>
...
<TR_MALL_COMPLETA>
  TBK_TIPO_TRANSACCION#A#50#1
  TBK_MONTO#M#10#1
  TBK_ORDEN_COMPRA#A#26#1
  TBK_ID_SESION#A#61#1
  TBK_URL_RESULTADO#A#256#1
  TBK_URL_FRACASO#A#256#1
  TBK_NUM_TRX#N#4#1
  TBK_NUMERO_TARJETA#N#19#1
  TBK_FECHA_EXPIRACION#N#4#1
  TBK_CVV#N#4#1
</TR_MALL_COMPLETA>
...
</BP_PAGO>
<BP_PAGO_MALL>
...
<TR_MALL_COMPLETA>
  TBK_CODIGO_TIENDA_M#N#12#1
  TBK_ORDEN_TIENDA_M#A#26#1
  TBK_MONTO_TIENDA_M#M#10#1
  TBK_TIPO_PAGO_M#A#2#1
  TBK_NUMERO_CUOTAS_M#N#2#0
  TBK_MONTO_CUOTA_M#M#9#0
</TR_MALL_COMPLETA>
</BP_PAGO_MALL>
```

#### **Paso 4: Instalando la llave privada del comercio y la llave pública de Transbank**

---

Las llaves pública (tbk\_public\_key.pem o tbk\_public\_key101prod.pem) y privada (privada.pem) son archivos de texto que vienen dentro del kit en la carpeta /cgi-bin/maestros/ y que por lo tanto ya deberían estar copiadas junto con todo el contenido de cgi-bin en la carpeta destinada por el servidor web para ejecución de CGI.

De las dos llaves públicas que vienen en la carpeta maestros se utiliza una sola. La elección dependerá del valor del parámetro TBK\_KEY\_ID configurado en el archivo tbk\_config.dat:

Si TBK\_KEY\_ID = 100, se debe utilizar la llave pública tbk\_public\_key.pem, es decir, no hacer ningún cambio respecto de lo que venía dentro del kit.

Si TBK\_KEY\_ID = 101 (como es el caso de este ejemplo), se debe utilizar la llave pública tbk\_public\_key101prod.pem pero con el nombre tbk\_public\_key.pem. Esto es, eliminar tbk\_public\_key.pem y renombrar tbk\_public\_key101prod.pem a tbk\_public\_key.pem.

#### **Paso 5: Configurando el nivel de log en archivo tbk\_trace.dat**

---

Validar la siguiente línea del archivo cgi-bin/datos/tbk\_trace.dat:

```
GLEVEL = 7
```

## Paso 6: Páginas de pago, cierre, éxito y fracaso

### Página de pago

La conexión con el KCC se hace a través del archivo `cgi-bin/tbk_bp_pago.cgi`, que debe ser llamado desde la página de pago del comercio. Este archivo tiene como objetivo recoger la información de la compra que se pretende realizar.

De acuerdo a los parámetros de entrada de una transacción **mall completa**, este formulario de pago tiene la siguiente forma:

```
<?php
/*
 * Valores de Ejemplo.
 * Estos valores, deben obtenerse del flujo de Transacción.
 */
// Transacción normal Completa
$TBK_TIPO_TRANSACCION = "TR_MALL_COMPLETA";

// Monto total de las transacciones.
$TBK_MONTO = 1000;

// Orden de compra. Este valor debe ser único.
$TBK_ORDEN_COMPRA = uniqid("oc");

// Identificador de sesión
$TBK_ID_SESION = date("Ymdhis");

// Número de transacciones
$TBK_NUM_TRX = 2;

// Página a la cuál se redireccionará al tarjeta habiente, en caso de éxito.
$TBK_URL_RESULTADO = "http://www.comercioFicticio.cl/webpay/resultado.php";

// Página a la cuál se redireccionará al tarjeta habiente, en caso de fracaso.
$TBK_URL_FRACASO = "http://www.comercioFicticio.cl/webpay/fracaso.php";

// Número de Tarjeta.
$TBK_NUMERO_TARJETA = 4051885600446623;

// Fecha de expiración de la Tarjeta formato mmaa, ejemplo: 0120
$TBK_FECHA_EXPIRACION = date("my");

// Código de verificación del tarjeta.
$TBK_CVV = 123;

// Tipo de pago: VN (Venta Normal), VC (Venta Cuotas), SI (Cuotas Sin Interes), CI (Cuotas Comercio)
$TBK_TIPO_PAGO = "VN";

// Código de tienda mall 01
$TBK_CODIGO_TIENDA_M001 = 597020000504;

// Orden de compra tienda mall 01
$TBK_ORDEN_TIENDA_M001 = uniqid("oc_tienda01_");

// Monto Transacción tienda mall 01
$TBK_MONTO_TIENDA_M001 = (int) (0.6 * $TBK_MONTO);

// Número de cuotas tienda mall 01, obligatorio en caso de venta en cuotas.
$TBK_NUMERO_CUOTAS_M001 = "";

// Monto de cada cuota tienda mall 01, obligatorio en caso de venta en cuotas.
$TBK_MONTO_CUOTA_M001 = "";

// Código de tienda mall 02
$TBK_CODIGO_TIENDA_M002 = 597020000505;

// Orden de compra tienda mall 02
$TBK_ORDEN_TIENDA_M002 = uniqid("oc_tienda02_");

// Monto Transacción tienda mall 02
$TBK_MONTO_TIENDA_M002 = (int) (0.4 * $TBK_MONTO);

// Número de cuotas tienda mall 02, obligatorio en caso de venta en cuotas.
$TBK_NUMERO_CUOTAS_M002 = "";

// Monto de cada cuota tienda mall 02, obligatorio en caso de venta en cuotas.
$TBK_MONTO_CUOTA_M002 = "";

// Url de la cgi de pago
$url_cgi = "http://www.comercioFicticio.cl/cgi-bin/tbk_bp_pago.cgi";
```



```

// Formateo de Moneda
if (!empty($TBK_MONTO)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO .= "00";
}

if (!empty($TBK_MONTO_TIENDA_M001)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO_TIENDA_M001 .= "00";
}

if (!empty($TBK_MONTO_TIENDA_M002)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO_TIENDA_M002 .= "00";
}

if (!empty($TBK_MONTO_CUOTA_M001)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO_CUOTA_M001 .= "00";
}

if (!empty($TBK_MONTO_CUOTA_M002)) {
    // Para la venta en pesos chilenos (CLP) se debe agregar al final del monto 00
    $TBK_MONTO_CUOTA_M002 .= "00";
}

/**
 * Es necesario almacenar los datos de transacción en algún mecanismo persistente,
 * para su consulta posterior. Se asume la existencia de una función almacenar_transaccion_mall
 * creada para dicho propósito, que retorna true en caso de éxito y false en cualquier otro caso.
 * Cada integración, debe crear un método de persistencia para la transacción.
 * Se deben almacenar al menos, la orden de compra y el monto.
 */
$tiendas = array();
$tiendas[] = array('TBK_CODIGO_COMERCIO' => $TBK_CODIGO_TIENDA_M001, 'TBK_ORDEN_COMPRA' =>
$TBK_ORDEN_TIENDA_M001, 'TBK_MONTO' => $TBK_MONTO_TIENDA_M001);
$tiendas[] = array('TBK_CODIGO_COMERCIO' => $TBK_CODIGO_TIENDA_M002, 'TBK_ORDEN_COMPRA' =>
$TBK_ORDEN_TIENDA_M002, 'TBK_MONTO' => $TBK_MONTO_TIENDA_M002);
$ok = almacenar_transaccion_mall($TBK_ORDEN_COMPRA, $TBK_MONTO, $tiendas);

if ($ok) {
    ?>
    <html>
        <head>
            <title>Pago</title>
            <meta charset="utf8" />
        </head>
        <body>
            <form name="frmPago" method="POST" action="<?php echo $url_cgi; ?>">
                <table border=0 align="center">
                    <tr>
                        <td><input type="text" name="TBK_TIPO_TRANSACCION" value="<?php echo $TBK_TIPO_TRANSACCION; ?>"
/></td>
                        <td><input type="text" name="TBK_MONTO" value="<?php echo $TBK_MONTO; ?>" /></td>
                    </tr>
                    <tr>
                        <td><input type="text" name="TBK_ORDEN_COMPRA" value="<?php echo $TBK_ORDEN_COMPRA; ?>" /></td>
                        <td><input type="text" name="TBK_ID_SESION" value="<?php echo $TBK_ID_SESION; ?>" /></td>
                    </tr>
                    <tr>
                        <td><input type="text" name="TBK_URL_RESULTADO" SIZE="30" value="<?php echo $TBK_URL_RESULTADO;
?>" /></td>
                        <td><input type="text" name="TBK_URL_FRACASO" SIZE="30" value="<?php echo $TBK_URL_FRACASO; ?>"
/></td>
                    </tr>
                    <tr>
                        <td>Número de Transacciones</td>
                        <td><input type="text" name="TBK_NUM_TRX" SIZE="30" value="<?php echo $TBK_NUM_TRX; ?>" /></td>
                    </tr>
                    <tr>
                        <td><input type="text" name="TBK_NUMERO_TARJETA" value="<?php echo $TBK_NUMERO_TARJETA; ?>"
/></td>
                        <td><input type="text" name="TBK_FECHA_EXPIRACION" value="<?php echo $TBK_FECHA_EXPIRACION; ?>"
/></td>
                    </tr>
                    <tr>
                        <td><input type="text" name="TBK_CVV" value="<?php echo $TBK_CVV; ?>" /></td>
                        <td></td>
                    </tr>
                </table>
            </form>
        </body>
    </html>
}

```

```

        <tr>
            <td>Tienda</td>
            <td>01</td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_CODIGO_TIENDA_M001" value="<?php echo $TBK_CODIGO_TIENDA_M001;
?>" /></td>
            <td><input type="text" name="TBK_ORDEN_TIENDA_M001" value="<?php echo $TBK_ORDEN_TIENDA_M001;
?>" /></td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_NUMERO_CUOTAS_M001" value="<?php echo
$TBK_NUMERO_CUOTAS_M001; ?>" /></td>
            <td><input type="text" name="TBK_MONTO_CUOTA_M001" value="<?php echo $TBK_MONTO_CUOTA_M001; ?>"
/></td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_MONTO_TIENDA_M001" value="<?php echo $TBK_MONTO_TIENDA_M001;
?>" /></td>
            <td><input type="text" name="TBK_TIPO_PAGO_M001" value="<?php echo $TBK_TIPO_PAGO; ?>" /></td>
        </tr>

        <tr>
            <td>Tienda</td>
            <td>02</td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_CODIGO_TIENDA_M002" value="<?php echo $TBK_CODIGO_TIENDA_M002;
?>" /></td>
            <td><input type="text" name="TBK_ORDEN_TIENDA_M002" value="<?php echo $TBK_ORDEN_TIENDA_M002;
?>" /></td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_NUMERO_CUOTAS_M002" value="<?php echo
$TBK_NUMERO_CUOTAS_M002; ?>" /></td>
            <td><input type="text" name="TBK_MONTO_CUOTA_M002" value="<?php echo $TBK_MONTO_CUOTA_M002; ?>"
/></td>
        </tr>
        <tr>
            <td><input type="text" name="TBK_MONTO_TIENDA_M002" value="<?php echo $TBK_MONTO_TIENDA_M002;
?>" /></td>
            <td><input type="text" name="TBK_TIPO_PAGO_M002" value="<?php echo $TBK_TIPO_PAGO; ?>" /></td>
        </tr>

        <tr>
            <td align="center" colspan="2">
                <input type="submit" />
            </td>
        </tr>
    </table>
</form>
</body>
</html>

```

## Página De Cierre

Esta página extrae los parámetros de respuesta que entrega Transbank, realiza validación de seguridad (MAC) y si corresponde, confirma la transacción entregando como respuesta "ACEPTADO". Para simplificar el ejemplo, se han realizado supuestos para las validaciones de monto y orden de compra. Es importante notar que estas validaciones dependen de la lógica de negocios de cada comercio.

```
<?php
// Iniciamos la variable, asumiendo 'RECHAZADO'.
$ack = "RECHAZADO";

try {
    error_log(print_r($_POST, true));

    /*
     * Recate de datos por POST, enviados desde TBK
     * Si al recibir TBK_RESPUESTA la variable es nula la pongo en 0. con esto se caerá por validación de check_mac
     */

    $TBK_RESPUESTA = (int) trim(strip_tags($_POST["TBK_RESPUESTA"]));
    $TBK_ORDEN_COMPRA = trim(strip_tags($_POST["TBK_ORDEN_COMPRA"]));
    $TBK_MONTO = (int) trim(strip_tags($_POST["TBK_MONTO"]));
    $TBK_ID_SESION = trim(strip_tags($_POST["TBK_ID_SESION"]));
    $TBK_TIPO_TRANSACCION = trim(strip_tags($_POST["TBK_TIPO_TRANSACCION"]));
    $TBK_FINAL_NUMERO_TARJETA = trim(strip_tags($_POST["TBK_FINAL_NUMERO_TARJETA"]));
    $TBK_FECHA_CONTABLE = trim(strip_tags($_POST["TBK_FECHA_CONTABLE"]));
    $TBK_FECHA_TRANSACCION = trim(strip_tags($_POST["TBK_FECHA_TRANSACCION"]));
    $TBK_HORA_TRANSACCION = trim(strip_tags($_POST["TBK_HORA_TRANSACCION"]));
    $TBK_ID_TRANSACCION = trim(strip_tags($_POST["TBK_ID_TRANSACCION"]));
    $TBK_VCI = trim(strip_tags($_POST["TBK_VCI"]));
    $TBK_TASA_INTERES_MAX = trim(strip_tags($_POST["TBK_TASA_INTERES_MAX"]));
    $TBK_MAC = trim(strip_tags($_POST["TBK_MAC"]));

    $TBK_COD_AUT_M001 = trim(strip_tags($_POST["TBK_COD_AUT_M001"]));
    $TBK_TIPO_PAGO_M001 = trim(strip_tags($_POST["TBK_TIPO_PAGO_M001"]));
    $TBK_NUMERO_CUOTAS_M001 = trim(strip_tags($_POST["TBK_NUMERO_CUOTAS_M001"]));
    $TBK_COD_RESP_M001 = (int) trim(strip_tags($_POST["TBK_COD_RESP_M001"]));
    $TBK_MONTO_TIENDA_M001 = (int) trim(strip_tags($_POST["TBK_MONTO_TIENDA_M001"]));
    $TBK_ORDEN_TIENDA_M001 = trim(strip_tags($_POST["TBK_ORDEN_TIENDA_M001"]));
    $TBK_VCI_M001 = trim(strip_tags($_POST["TBK_VCI_M001"]));

    $TBK_COD_AUT_M002 = trim(strip_tags($_POST["TBK_COD_AUT_M002"]));
    $TBK_TIPO_PAGO_M002 = trim(strip_tags($_POST["TBK_TIPO_PAGO_M002"]));
    $TBK_NUMERO_CUOTAS_M002 = trim(strip_tags($_POST["TBK_NUMERO_CUOTAS_M002"]));
    $TBK_COD_RESP_M002 = (int) trim(strip_tags($_POST["TBK_COD_RESP_M002"]));
    $TBK_MONTO_TIENDA_M002 = (int) trim(strip_tags($_POST["TBK_MONTO_TIENDA_M002"]));
    $TBK_ORDEN_TIENDA_M002 = trim(strip_tags($_POST["TBK_ORDEN_TIENDA_M002"]));
    $TBK_VCI_M002 = trim(strip_tags($_POST["TBK_VCI_M002"]));

    /*
     * Se asume la existencia de una función que permita actualizar los datos de la transacción
     * en un mecanimos de persistencia (En nuestro ejemplo la llamaremos actualizar_transaccion)
     * Esta función debería retornar true en caso de éxito y false en cualquier otro caso.
     */

    $tiendas = array();
    $tiendas[] = array('TBK_COD_AUTH' => $TBK_COD_AUT_M001, 'TBK_TIPO_PAGO' => $TBK_TIPO_PAGO_M001,
    'TBK_NUMERO_CUOTAS' => $TBK_NUMERO_CUOTAS_M001, 'TBK_COD_RESP' => $TBK_COD_RESP_M001,
    'TBK_MONTO_TIENDA' => $TBK_MONTO_TIENDA_M001, 'TBK_ORDEN_TIENDA' => $TBK_ORDEN_TIENDA_M001, 'TBK_VCI'
    => $TBK_VCI_M001);
    $tiendas[] = array('TBK_COD_AUTH' => $TBK_COD_AUT_M002, 'TBK_TIPO_PAGO' => $TBK_TIPO_PAGO_M002,
    'TBK_NUMERO_CUOTAS' => $TBK_NUMERO_CUOTAS_M002, 'TBK_COD_RESP' => $TBK_COD_RESP_M002,
    'TBK_MONTO_TIENDA' => $TBK_MONTO_TIENDA_M002, 'TBK_ORDEN_TIENDA' => $TBK_ORDEN_TIENDA_M002, 'TBK_VCI'
    => $TBK_VCI_M002);
    actualizar_transaccion_mall($TBK_ORDEN_COMPRA, $TBK_TIPO_TRANSACCION, $TBK_RESPUESTA, $TBK_MONTO,
    $TBK_FINAL_NUMERO_TARJETA, $TBK_FECHA_CONTABLE, $TBK_FECHA_TRANSACCION, $TBK_HORA_TRANSACCION,
    $TBK_ID_SESION, $TBK_ID_TRANSACCION, $TBK_TASA_INTERES_MAX, $TBK_VCI, $TBK_MAC, $tiendas);

    /*
     * se asume que existe una función (en nuestro ejemplo: obtener_transacción)
     * que permite obtener los datos de la transacción, desde el almacenamiento
     * persistente usando la orden de compra, en este ejemplo la función retorna
     * un array asociativo con los datos de la transacción o false en cualquier
     * otro caso
     */
    $transaccion = obtener_transaccion_mall($TBK_ORDEN_COMPRA);
```

xt\_compra.php

```

// Verifico que la orden de compra exista en el almacenamiento persistente
if ($transaccion !== false) {
    $monto = $transaccion['TBK_MONTO'];
    $acepta = false;

    /*
     * Continuo la Validación, si el monto obtenido desde el almacenamiento
     * persistente, es igual al monto notificado por Webpay.
     */
    if ($monto == $TBK_MONTO) {
        /*
         * Si el TBK_RESPUESTA es igual a 0, implica que webpay aprueba la
         * transacción, un valor distinto implica rechazo de la transacción en
         * webpay.
         */
        if ($TBK_RESPUESTA == 0) {
            $data = $_POST;
            /*
             * Genero el archivo de validación checkmac
             * Es importante que la carpeta tenga permisos de escritura.
             */
            $archivo_checkmac = "/tmp/mac_tr_mall_completa_$TBK_ORDEN_COMPRA.txt";
            $fp = fopen($archivo_checkmac, "w+");
            foreach ($data as $key => $val) {
                fwrite($fp, "$key=$val&");
            }
            fclose($fp);
            //ejecución del check_mac
            $cmd = "/srv/httpd/cgi-bin/kcc_mall/tbk_check_mac.cgi $archivo_checkmac";
            exec($cmd, $resultado, $retint);
            // Después de realizar la comprobación, elimino el archivo de checkmac
            unlink($archivo_checkmac);

            /*
             * Si el resultado obtenido de la ejecución de la CheckMac es 'CORRECTO'
             * Aceptamos la Transacción de WebPay
             */
            if (trim($resultado[0]) == "CORRECTO") {
                $acepta = true;
            }
        } else {
            /*
             * Si TBK_RESPUESTA es distinto de 0, la transacción es rechazada por Webpay
             * Pero es aceptada por la tienda.
             */
            $acepta = true;
        }
    }
}

$ack = $acepta ? "ACEPTADO" : "RECHAZADO";
}
} catch (Exception $e) {
    error_log($e->getMessage());
    $ack = "RECHAZADO";
}
}

/*
 * Si la tienda acepta la transacción, respondemos 'ACEPTADO', si no, responderemos
 * 'RECHAZADO' en cualquier otro caso.
 * Es importante recalcar que la respuesta no puede tener ningún otro carácter fuera de la palabra ACEPTADO o RECHAZADO
 * Ni siquiera debe incluir código HTML.
 */

echo ($ack);
?>

```

del kit. Esta CGI realiza la validación de MAC (CORRECTO - INVALIDO).

## Página de resultado

El parámetro TBK\_URL\_RESULTADO corresponde a la URL que se llamará en caso que la transacción se haya llevado a cabo correctamente. La información de la transacción puede ser desplegada recuperando el valor del parámetro TBK\_ORDEN\_DE\_COMPRA.

Ejemplo página resultado:

resultado.php

```
<?php

// Obtención de datos por POST, enviados desde TBK
$TBK_ORDEN_COMPRA = trim(strip_tags($_POST["TBK_ORDEN_COMPRA"]));

/*
 * Se asume que existe una función (en nuestro ejemplo: obtener_transacción) que permite obtener los datos
 * de la transacción desde el almacenamiento persistente, usando la orden de compra. En este ejemplo la
 * función retorna un array asociativo con los datos de la transacción o false en cualquier otro caso
 */
$transaccion = obtener_transaccion_mall($TBK_ORDEN_COMPRA);
if ($transaccion !== false) {
    $TBK_TIPO_TRANSACCION = $transaccion['TBK_TIPO_TRANSACCION'];
    $TBK_RESPUESTA = $transaccion['TBK_RESPUESTA'];
    $TBK_MONTO = $transaccion['TBK_MONTO'];
    $TBK_FINAL_NUMERO_TARJETA = $transaccion['TBK_FINAL_NUMERO_TARJETA'];
    $TBK_FECHA_CONTABLE = $transaccion['TBK_FECHA_CONTABLE'];
    $TBK_FECHA_TRANSACCION = $transaccion['TBK_FECHA_TRANSACCION'];
    $TBK_HORA_TRANSACCION = $transaccion['TBK_HORA_TRANSACCION'];
    $TBK_ID_TRANSACCION = $transaccion['TBK_ID_TRANSACCION'];
    $TBK_TASA_INTERES_MAX = $transaccion['TBK_TASA_INTERES_MAX'];
    $TBK_VCI = $transaccion['TBK_VCI'];
    $TBK_MAC = $transaccion['TBK_MAC'];
    $tiendas = $transaccion['tiendas'];

    $TBK_FECHA_TRANSACCION = substr($TBK_FECHA_TRANSACCION, 2, 2) . "/" .
    substr($TBK_FECHA_TRANSACCION, 0, 2) . "/" . date("Y") . " " . substr($TBK_HORA_TRANSACCION, 0, 2) . ":" .
    substr($TBK_HORA_TRANSACCION, 2, 2) . ":" . substr($TBK_HORA_TRANSACCION, 4, 2);
    $monto = (int) ($TBK_MONTO / 100); ?>

<html>
<head>
<title>TRANSACCION EXITOSA</title>
<meta charset="utf8" />
</head>
<body>
<h1>Datos de la Transacción</h1>
<p>
    El pago se ha realizado con éxito. Imprima este comprobante de pago.
</p>
<div class="codigo">
<table>
<tr>
<td><b>Orden de Compra</b></td>
<td><?php echo $TBK_ORDEN_COMPRA; ?></td>
</tr>
<tr>
<td><b>Monto</b></td>
<td><?php echo $monto; ?></td>
</tr>
<tr>
<td><b>Fecha Transacción</b></td>
<td><?php echo $TBK_FECHA_TRANSACCION; ?></td>
</tr>
<tr>
<td><b>Tipo Transacción</b></td>
<td><?php echo $TBK_TIPO_TRANSACCION; ?></td>
</tr>
<tr>
<td><b>Monto Total</b></td>
<td><?php echo $TBK_MONTO; ?></td>
</tr>
</table>
</div>
</body>
</html>
```

```

<?php
$num_tiendas = (int) count($tiendas);
if ($num_tiendas > 0) {
    ?>
    <table>
        <thead>
            <tr>
                <th>Código Comercio</th>
                <th>Orden Compra</th>
                <th>Monto</th>
                <th>Código de Autorización</th>
                <th>Tipo Pago</th>
                <th>Número de Cuotas</th>
                <th>Monto de Cuota</th>
                <th>Código Respuesta</th>
            </tr>
        </thead>
        <tbody>
            <?php
            for ($i = 0; $i < $num_tiendas; $i++) {
                $codigo_comercio = $tiendas[$i]['TBK_CODIGO_COMERCIO'];
                $soc_tienda = $tiendas[$i]['TBK_ORDEN_TIENDA'];
                $monto_tienda = $tiendas[$i]['TBK_MONTO_TIENDA'];
                $codigo_auth = $tiendas[$i]['TBK_COD_AUTH'];
                $tipo_pago = $tiendas[$i]['TBK_TIPO_PAGO'];
                $num_cuotas = (int) $tiendas[$i]['TBK_NUMERO_CUOTAS'];
                if ($num_cuotas == 0) {
                    $num_cuotas = 1;
                }
                $monto_cuota = (int) $tiendas[$i]['TBK_MONTO_TIENDA'];
                $tienda_respuesta = $tiendas[$i]['TBK_COD_RESP'];

                echo "<tr>
                    <td>$codigo_comercio</td>
                    <td>$soc_tienda</td>
                    <td>$monto_tienda</td>
                    <td>$codigo_auth</td>
                    <td>$tipo_pago</td>
                    <td>$num_cuotas</td>
                    <td>$monto_tienda</td>
                    <td>$tienda_respuesta</td>
                </tr>";
            }
            ?>
        </tbody>
    </table>
    <?php
    }
    ?>

</div>
<br />
<p><a href="index.php">Volver a la tienda</a></p>
</body>
</html>

```

## Página de fracaso

El parámetro TBK\_URL\_FRACASO corresponde a la URL que se llamará en el caso que la transacción no se pueda llevar a cabo. No hay ningún requerimiento sobre esta página, basta con que contenga un mensaje apropiado que informe al usuario que la transacción no se realizó.

### Ejemplo página de fracaso:

fracaso.php

```
<?php
$orden = trim(strip_tags($_REQUEST['TBK_ORDEN_COMPRA']));
if (empty($orden)) {
    $orden = trim(strip_tags($_REQUEST['TBK_ID_SESION']));
}
?>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf8" />
  </head>
  <body>
    <h1>Error</h1>
    <p>
      La transacción <?php echo $orden; ?> fue rechazada.
    </p>
    <div class="codigo">
      <ul>
        <li>Error en el ingreso de los datos de su tarjeta de crédito (fecha y/o código de seguridad).</li>
        <li>Su tarjeta de crédito no cuenta con el cupo necesario para cancelar la compra.</li>
        <li>Tarjeta aún no habilitada en el sistema financiero.</li>
      </ul>
      Si el problema persiste favor comunicarse con su Banco emisor.
    <br />

    </div>
    <br />
    <p><a href="index.php">Volver a la tienda</a></p>
  </body>
</html>
```

**Transbank S.A.**  
**COPYRIGHT©2001**

(Última Actualización Junio de 2016)