# Manual de integración con el TPV Virtual para comercios con conexión por Web Service

Versión: 2.4

31/05/2018

**RS.TE.CEL.MAN.0004**

Redsys · C/ Francisco Sancha, 12 · 28034 · Madrid · ESPAÑA

# Autorizaciones y control de versión

| Versión | Fecha | Afecta | Breve descripción del cambio |
|---|---|---|---|
| 1.0 | 16/10/2015 | | Versión inicial del documento |
| 1.1 | 29/10/2015 | | Se añade el detalle sobre la decodificación de la clave del comercio, previo al cálculo de la clave específica de la operación |
| 1.2 | 04/11/2015 | | Se añade el código de anulación autorizada en la tabla de valores del Ds_Response |
| 1.3 | 10/11/2015 | | Modificaciones del API Java |
| 1.4 | 13/11/2015 | | Se añade todo lo relacionado con el API .NET |
| 1.5 | 01/12/2015 | | Se añade un apartado con información para realización de pruebas en entorno de test. |
| 1.6 | 11/12/2015 | | Se añade información sobre el Pago por Referencia (Pago 1-Clic).<br>Además se modifica el apartado del cálculo de la firma para los comercios configurados con envío de tarjeta en la respuesta. |
| 1.7 | 09/02/2016 | | Añadida operativa DCC con ejemplos |
| 1.8 | 23/02/2016 | | Incorporación del error SIS0444 |
| 1.9 | 23/05/2016 | | Incorporación de nuevos códigos de error en el "Glosario de errores" |
| 2.0 | 30/05/2016 | | Modificación códigos de error |
| 2.1 | 20/09/2016 | | Incorporación de nuevos códigos de error en el "Glosario de errores" |
| 2.2 | 12/05/2017 | | Se añade el parámetro de respuesta Ds_Card_Brand |
| 2.3 | 26/05/2017 | | Se añade anexo con operativa ApplePay y AndroidPay |
| 2.4 | 31/05/2018 | | Se elimina pago tradicional A y recurrente |

# ÍNDICE DE CONTENIDO

# 1. Introducción

## 1.1 Objetivo

Este documento recoge los aspectos técnicos necesarios para que un comercio realice la integración con el TPV Virtual mediante conexión Web Service.

Esta forma de conexión permite a los comercios tener integrado el TPV Virtual dentro de su propia aplicación Web. No están permitidos aquellos comercios que posean métodos de pago seguros, que solicitan autenticación del titular por parte de la entidad emisora de la tarjeta, ya que solo permite realizar pagos tradicionales.

**NOTA: la conexión requiere del uso de un sistema de firma basado en HMAC SHA-256, que autentica entre sí al servidor del comercio y al TPV Virtual. Para desarrollar el cálculo de este tipo de firma, el comercio puede realizar el desarrollo por sí mismo utilizando las funciones estándar de los diferentes entornos de desarrollo, si bien para facilitar los desarrollos ponemos a su disposición librerías (PHP, JAVA y .NET) cuya utilización se presenta en detalle en esta guía y que están a su disposición en la siguiente dirección:**

http://www.redsys.es/wps/portal/redsys/publica/areadeserviciosweb/descargaDeDocumentacionYEjecutables/

## 1.2 Definiciones, siglas y abreviaturas

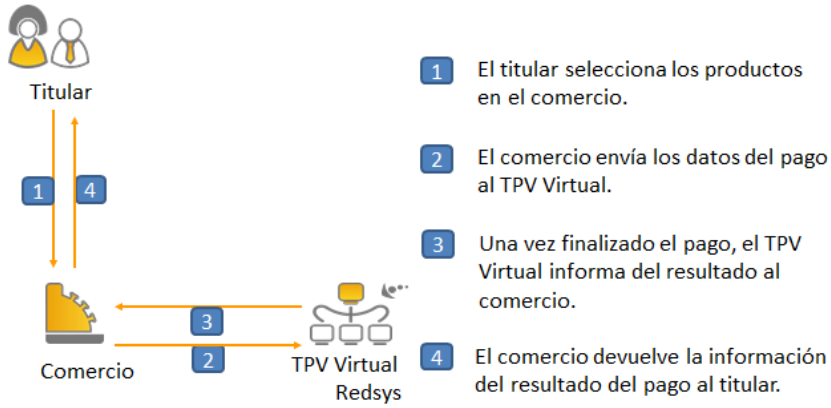
SIS. Servidor Integrado de Redsys (Servidor del TPV Virtual).

## 1.3 Referencias

- Documentación de Integración con el SIS
- Guía de comercios del SIS.

# 2. Descripción general del flujo

El siguiente esquema presenta el flujo general de una operación realizada con el Web Service de Redsys.




## 2.1 Envío de petición al TPV Virtual

Como se muestra en el paso 2 del esquema anterior, el comercio debe enviar al TPV Virtual los datos de la petición de pago vía Web Service con codificación UTF-8. Para ello el Web Service tiene publicados varios métodos sobre los cuales operan los TPV Virtuales. El método "**trataPeticion"**, permite la realización de operaciones a través del Web Service, para lo cual se debe construir un XML que incluye los datos de la petición de pago. La descripción exacta de esta petición XML se presenta mediante el fichero WSDL en el Anexo 6 (Web Service de petición de pago - WSDL) del apartado Anexos del presente documento.

Esta petición de pago debe enviarse a las siguientes URLs dependiendo de si se quiere realizar una petición de pruebas u operaciones reales:

| URL Conexión | Entorno |
|---|---|
| **https://sis-t.redsys.es:25443/sis/services/SerClsWSEntrada** | Pruebas |
| **https://sis.redsys.es/sis/services/SerClsWSEntrada** | Real |

Una vez enviada la petición el TPV Virtual la interpretará y realizará las validaciones necesarias para, a continuación, procesar la operación, tal y como se muestra en el paso 3 del esquema anterior. Dependiendo del resultado de la operación, se construye un documento XML de respuesta con el resultado de la misma con codificación UTF-8.

**NOTA: Todo lo relacionado con la respuesta del Web Service se expone en el apartado 4.**

# 3. Mensaje de petición de pago Web Service

Para que el comercio pueda realizar la petición a través del Web Service de Redsys, es necesario intercambiar una serie de datos, tanto en los mensajes de petición como en los mensajes de respuesta.

La estructura del mensaje siempre será la misma, estableciendo como raíz del mismo el elemento **<REQUEST>**. En su interior siempre deben encontrarse tres elementos que hacen referencia a:

- Datos de la petición de pago. Elemento identificado por la etiqueta **<DATOSENTRADA>**.
- Versión del algoritmo de firma. Elemento identificado por la etiqueta **<DS_SIGNATUREVERSION>**.
- Firma de los datos de la petición de pago. Elemento identificado por la etiqueta **<DS_SIGNATURE>**.

A continuación se muestra un ejemplo de un mensaje de petición de pago:

```
<REQUEST>
        <DATOSENTRADA>
                <DS_MERCHANT_AMOUNT>145</DS_MERCHANT_AMOUNT>
                <DS_MERCHANT_ORDER>1444904795</DS_MERCHANT_ORDER>
                <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
                <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
                <DS_MERCHANT_PAN>XXXXXXXXXXXXXXXX</DS_MERCHANT_PAN>
                <DS_MERCHANT_CVV2>XXX</DS_MERCHANT_CVV2>
                <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
                <DS_MERCHANT_TERMINAL>871</DS_MERCHANT_TERMINAL>
                <DS_MERCHANT_EXPIRYDATE>XXXX</DS_MERCHANT_EXPIRYDATE>
        </DATOSENTRADA>
        <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>
        <DS_SIGNATURE>
                VV3acxBgABrS5VYcLyJD1KqIsa2pPdvajPBG510lFfg=
        </DS_SIGNATURE>
</REQUEST>
```

Para facilitar la integración del comercio, a continuación se explica de forma detallada los pasos a seguir para montar el mensaje de petición de pago.

## 3.1 Montar la cadena de datos de la petición

Se debe montar una cadena con todos los datos de la petición en formato XML dando como resultado el elemento **<DATOSENTRADA>**.

Se debe tener en cuenta que existen varios tipos de peticiones y según el tipo varía la estructura del mensaje y los parámetros que se envían y reciben.

Podemos diferenciar tres tipos de peticiones:

- Peticiones de pago (con envío de datos de tarjeta). En el Anexo 1 (Peticiones de pago) del apartado Anexos del presente documento, se presentan los parámetros necesarios para este tipo de petición incluyendo un ejemplo.
- Peticiones de pagos recurrentes (con envío de datos de tarjeta). En el Anexo 2 (Peticiones de pago recurrentes) del apartado Anexos del presente documento, se presentan los parámetros necesarios para este tipo de petición incluyendo un ejemplo.
- Peticiones de Confirmación/Devolución. En el Anexo 3 (Peticiones de Confirmación/Devolución) del apartado Anexos del presente documento, se presentan los parámetros necesarios para este tipo de petición incluyendo un ejemplo.

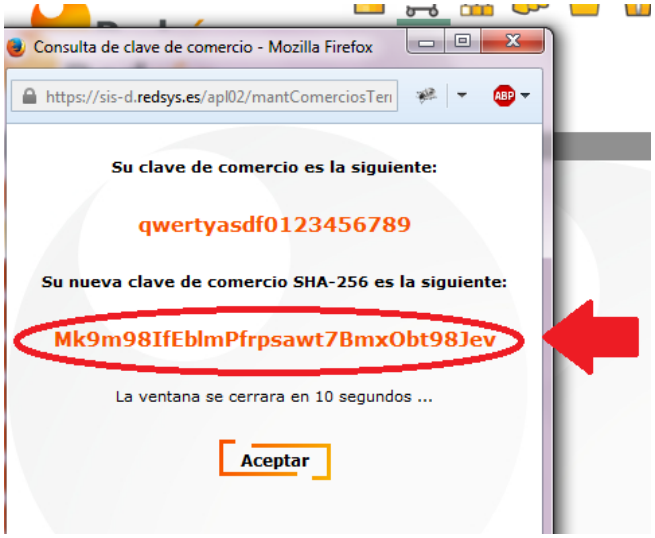
Para comercios que utilicen operativas especiales como el "Pago por referencia" (Pago 1-Clic), deberán incluir los campos específicos de este tipo de operativa en el elemento <DATOSENTRADA>. En el Anexo 4 (Peticiones de pago por Referencia) del apartado Anexos del presente documento, se presentan los parámetros necesarios para este tipo de petición incluyendo un ejemplo.

## 3.2 Identificar la versión de algoritmo de firma a utilizar

En la petición se debe identificar la versión concreta de algoritmo que se está utilizando para la firma. Actualmente se utiliza el valor **HMAC_SHA256_V1** para identificar la versión de todas las peticiones, por lo que este será el valor del elemento **<DS_SIGNATUREVERSION>**, tal y como se puede observar en el ejemplo de mensaje mostrado al inicio del apartado 3.

## 3.3 Identificar la clave a utilizar para la firma

Para calcular la firma es necesario utilizar una clave específica para cada terminal. Se puede obtener la clave accediendo al Módulo de Administración, opción Consulta datos de Comercio, en el apartado "Ver clave", tal y como se muestra en la siguiente imagen:




**NOTA IMPORTANTE: Esta clave debe ser almacenada en el servidor del comercio de la forma más segura posible para evitar un uso fraudulento de la misma. El comercio es responsable de la adecuada custodia y mantenimiento en secreto de dicha clave.**

## 3.4 Firmar los datos de la petición

Una vez se tiene montada el elemento con los datos de la petición de pago (<DATOSENTRADA>) y la clave específica del terminal se debe calcular la firma siguiendo los siguientes pasos:

1. Se genera una clave específica por operación. Para obtener la clave derivada a utilizar en una operación se debe realizar un cifrado 3DES entre la clave del comercio, la cual debe ser previamente decodificada en BASE 64, y el valor del número de pedido de la operación (DS_MERCHANT_ORDER).

2. Se calcula el HMAC SHA256 del elemento <DATOSENTRADA>.

3. El resultado obtenido se codifica en BASE 64, y el resultado de la codificación será el valor del elemento **<DS_SIGNATURE>**, tal y como se puede observar en el ejemplo de formulario mostrado al inicio del apartado 3.

**NOTA: La utilización de las librerías de ayuda proporcionadas por Redsys para la generación de este campo, se expone en el apartado 3.5.**

## 3.5 Utilización de librerías de ayuda

En los apartados anteriores se ha descrito la forma de acceso al SIS utilizando conexión por Web Service y el sistema de firma basado en HMAC SHA256. En este apartado se explica cómo se utilizan las librerías disponibles en PHP, JAVA y .NET para facilitar los desarrollos y la generación de la firma. El uso de las librerías suministradas por Redsys es opcional, si bien simplifican los desarrollos a realizar por el comercio.

### 3.5.1 Librería PHP

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería PHP proporcionada por Redsys:

1. Importar el fichero principal de la librería, tal y como se muestra a continuación:

```
include_once 'redsysHMAC256_API_WS_PHP_4.0.2/apiRedsysWs.php';
```

   El comercio debe decidir si la importación desea hacerla con la función "include" o "required", según los desarrollos realizados.

2. Definir un objeto de la clase principal de la librería, tal y como se muestra a continuación:

```
$miObj = new RedsysAPIWs;
```

3. Calcular el elemento **<DS_SIGNATURE>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createMerchantSignatureHostToHost()" con la clave obtenida del módulo de administración y el elemento con los datos de la petición de pago (**<DATOSENTRADA>**), tal y como se muestra a continuación:

```
$datosEntrada="<DATOSENTRADA><DS_MERCHANT_AMOUNT>200</DS_MERCHANT_AMOUNT><DS_MERCHANT_CURRENCY>978
$claveModuloAdmin = 'Mk9m98IfEblmPfrpsawt7BmxObt98Jev';
$signature = $miObj->createMerchantSignatureHostToHost($claveModuloAdmin, $datosEntrada);
```

   Una vez obtenido el valor del elemento **<DS_SIGNATURE>**, ya se puedo completar el mensaje de petición de pago y realizar la llamada Web Service.

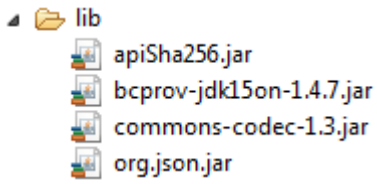
### 3.5.2 Librería JAVA

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería JAVA proporcionada por Redsys:

1. Importar la librería, tal y como se muestra a continuación:

```
<%@page import="sis.redsys.api.ApiWsMacSha256"%>
```

El comercio debe incluir en la vía de construcción del proyecto todas las librerías(JARs) que se proporcionan:




2. Definir un objeto de la clase principal de la librería, tal y como se muestra a continuación:

```
ApiWsMacSha256 apiWsMacSha256 = new ApiWsMacSha256();
```

3. Calcular el elemento **<DS_SIGNATURE>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createMerchantSignatureHostToHost()" con la clave obtenida del módulo de administración y el elemento con los datos de la petición de pago (**<DATOSENTRADA>**), tal y como se muestra a continuación:

```
String datosEntrada = "<DATOSENTRADA><DS_MERCHANT_AMOUNT>200</DS_MERCHANT_AMOUNT><DS_MERCHANT_CURRENCY>978
String claveModuloAdmin = "Mk9m98IfEblmPfrpsawt7BmxObt98Jev";
String signature = apiWsMacSha256.createMerchantSignatureHostToHost(claveModuloAdmin, datosEntrada);
```

Una vez obtenido el valor del elemento **<DS_SIGNATURE>**, ya se puedo completar el mensaje de petición de pago y realizar la llamada Web Service.

### 3.5.3 Librería .NET

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería .NET proporcionada por Redsys:

1. Importar la librería, tal y como se muestra a continuación:

```
using RedsysAPIPrj;
```

2. Crear un objeto de la clase del Web Service de Redsys. Para poder realizar esto es necesario añadir una nueva referencia web con el fichero SerClsWSEntrada.wsdl.

```
WebRedsysWs.SerClsWSEntradaService s = new WebRedsysWs.SerClsWSEntradaService();
```

Nota: En el atributo *location* de la etiqueta <wsdlsoap:address> Del fichero SerClsWSEntrada.wsdl, indicar si se trata del entorno real o pruebas:

*https://sis-t.redsys.es:25443/sis/services/SerClsWSEntrada (Pruebas)*
*https://sis.redsys.es/sis/services/SerClsWSEntrada (Real)*

3. Definir un objeto de la clase principal de la librería, tal y como se muestra a continuación:

```
RedsysAPIWs r = new RedsysAPIWs();
```

Al realizar este paso se inicializan los atributos diccionario clave/valor m_keyvalues y cryp de la clase Cryptogra (Clase auxiliar para realizar las operaciones criptográficas necesarias)

4. Generar parametros de DATOSENTRADA (Modalidad Petición de Pago con envío de datos de tarjeta) mediante la función:

```
string dataEntrada = r.GenerateDatoEntradaXML(amount, fuc, currency, pan, cvv2, trans, terminal, expire);
```

5. Calcular el elemento **<DS_SIGNATURE>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createMerchantSignatureHostToHost()" con la clave obtenida del módulo de administración y el elemento con los datos de la petición de pago (**<DATOSENTRADA>**), tal y como se muestra a continuación:

```
string signature = r.createMerchantSignatureHostToHost(kc, dataEntrada);
```

Una vez obtenido el valor del elemento **<DS_SIGNATURE>**, ya se puedo completar el mensaje de petición de pago y realizar la llamada Host to Host.
Se genera el string XML final de petición de pago con DATOSENTRADA, DS_SIGNATUREVERSION y DS_SIGNATURE calculado en punto 5.

```
string requestXML = r.GenerateRequestXML(dataEntrada, signature);
```

Después se llama al método trataPeticion del Web service de Redsys pasándole como parámetro el string XML final calculado con el método GenerateRequestXML.

```
string result = s.trataPeticion(requestXML);
```

# 4. Respuesta de petición Web Service

En el presente apartado se describen los datos que forman parte del mensaje de respuesta de una petición al TPV Virtual WebService. Este mensaje se genera en formato XML y a continuación se muestran ejemplos:

Ejemplo de respuesta de pago **(comercio configurado sin envío de datos de tarjeta):**

```
<RETORNOXML>
	<CODIGO>0</CODIGO>
	<OPERACION>
		<Ds_Amount>145</Ds_Amount>
		<Ds_Currency>978</Ds_Currency>
		<Ds_Order>1444912789</Ds_Order>
		<Ds_Signature>
			bAuiQOymGvYzqHi7dEeuWrRYFeUjtFH6NyOoWSl0vHU=
		</Ds_Signature>
		<Ds_MerchantCode>999008881</Ds_MerchantCode>
		<Ds_Terminal>871</Ds_Terminal>
		<Ds_Response>0000</Ds_Response>
		<Ds_AuthorisationCode>050372</Ds_AuthorisationCode>
		<Ds_TransactionType>0</Ds_TransactionType>
		<Ds_SecurePayment>0</Ds_SecurePayment>
		<Ds_Language>1</Ds_Language>
		<Ds_Card_Type>D</Ds_Card_Type>
		<Ds_MerchantData></Ds_MerchantData>
		<Ds_Card_Country>724</Ds_Card_Country>
		<Ds_Card_Brand>1</Ds_Card_Brand>
	</OPERACION>
</RETORNOXML>
```

Ejemplo de respuesta de pago **(comercio configurado con envío de datos de tarjeta):**

```
<RETORNOXML>
	<CODIGO>0</CODIGO>
	<OPERACION>
		<Ds_Amount>145</Ds_Amount>
		<Ds_Currency>978</Ds_Currency>
		<Ds_Order>1449821545</Ds_Order>
		<Ds_Signature>
			6quLImPCOSTFpwhC7+ai1L+SPdKbcGx2sgC2A/1hwQo=
		</Ds_Signature>
		<Ds_MerchantCode>999008881</Ds_MerchantCode>
		<Ds_Terminal>871</Ds_Terminal>
		<Ds_Response>0000</Ds_Response>
		<Ds_AuthorisationCode>109761</Ds_AuthorisationCode>
		<Ds_TransactionType>0</Ds_TransactionType>
		<Ds_SecurePayment>0</Ds_SecurePayment>
		<Ds_Language>1</Ds_Language>
		<Ds_CardNumber>4548812049400004</Ds_CardNumber>
		<Ds_MerchantData></Ds_MerchantData>
		<Ds_Card_Country>724</Ds_Card_Country>
		<Ds_Card_Brand>1</Ds_Card_Brand>
	</OPERACION>
</RETORNOXML>
```

Como se puede observar en el ejemplo anterior, la respuesta está formada por dos elementos principales:

- Código (**<CODIGO>**): Indica si la operación ha sido correcta o no, (no indica si ha sido autorizada, solo si se ha procesado). Un 0 indica que la operación ha sido correcta. En el caso de que sea distinto de 0, tendrá un código. (Ver códigos de error en apartado 5 de esta Guía)

- Datos de la operación (**<OPERACION>**): Recoge toda la información necesaria sobre la operación que se ha realizado. Mediante este elemento se determina si la operación ha sido autorizada o no.

**NOTA: La relación de parámetros que forman parte de la respuesta se describe en el Anexo 5 (Respuesta Host to Host) del apartado Anexos del presente documento.**

## 4.1 Firma del mensaje de respuesta

Una vez se ha obtenido el mensaje de respuesta y la clave específica del terminal, siempre y cuando la operación se autorice, se debe comprobar la firma de la respuesta siguiendo los siguientes pasos:

1. Se genera una clave específica por operación. Para obtener la clave derivada a utilizar en una operación se debe realizar un cifrado 3DES entre la clave del comercio, la cual debe ser previamente decodificada en BASE 64, y el valor del número de pedido de la operación (DS_ORDER).

2. Se calcula el HMAC SHA256 de la cadena formada por la concatenación del valor de los siguientes campos:

   **Cadena = Ds_Amount + Ds_Order + Ds_MerchantCode + Ds_Currency + Ds_Response + Ds_TransactionType + Ds_SecurePayment**

   Si tomamos como ejemplo la respuesta que se presenta al inicio de este apartado la cadena resultante sería:

   **Cadena = 1451444912789999008881978000000**

   Si el comercio tiene configurado envío de tarjeta en la respuesta, se debe calcular el HMAC SHA256 de la cadena formada por la concatenación del valor de los siguientes campos:

   **Cadena = Ds_Amount + Ds_Order + Ds_MerchantCode + Ds_Currency + Ds_Response + Ds_CardNumber + Ds_TransactionType + Ds_SecurePayment**

Si tomamos como ejemplo la respuesta que se presenta al inicio de este apartado la cadena resultante sería:

**Cadena = 145144982154599900888197800004548812049400000400**

3. El resultado obtenido se codifica en BASE 64, y el resultado de la codificación debe ser el mismo que el valor del parámetro **<Ds_Signature>** obtenido en la respuesta.

**NOTA: La utilización de las librerías de ayuda proporcionadas por Redsys para la generación de este parámetro, se expone en el apartado 4.2.**

## 4.2 Utilización de librerías de ayuda

En este apartado se explica cómo se utilizan las librerías disponibles en PHP, JAVA y .NET para facilitar los desarrollos y la generación de la firma de respuesta. El uso de las librerías suministradas por Redsys es opcional, si bien simplifican los desarrollos a realizar por el comercio.

### 4.2.1 Librería PHP

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería PHP proporcionada por Redsys:

1. Importar el fichero principal de la librería, tal y como se muestra a continuación:

```
include_once 'redsysHMAC256_API_WS_PHP_4.0.2/apiRedsysWs.php';
```

El comercio debe decidir si la importación desea hacerla con la función "include" o "required", según los desarrollos realizados.

2. Definir un objeto de la clase principal de la librería, tal y como se muestra a continuación:

```
$miObj = new RedsysAPIWs;
```

3. Calcular el parámetro **<Ds_Signature>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createSignatureResponseHostToHost()" con la clave obtenida del módulo de administración, la cadena que se desea firmar(concatenación de campos descrita en el punto 2 del apartado 4.1 del presente documento) y el número de pedido.

```
$cadenaConcatenada="14514449127899990088819780000000";
$numPedido="1444912789";
$claveModuloAdmin = 'Mk9m98IfEblmPfrpsawt7BmxObt98Jev';
$signature = $miObj->createSignatureResponseHostToHost($claveModuloAdmin,
                                                       $cadenaConcatenada,
                                                       $numPedido);
```

El resultado obtenido debe ser el mismo que el valor del parámetro **<Ds_Signature>** obtenido en la respuesta.

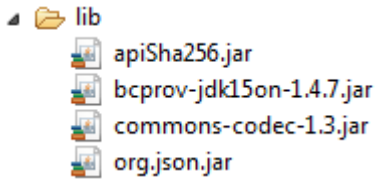
### 4.2.2 Librería JAVA

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería JAVA proporcionada por Redsys:

1. Importar la librería, tal y como se muestra a continuación:

```
<%@page import="sis.redsys.api.ApiWsMacSha256"%>
```

El comercio debe incluir en la vía de construcción del proyecto todas las librerías(JARs) que se proporcionan:




2. Definir un objeto de la clase principal de la librería, tal y como se muestra a continuación:

```
ApiWsMacSha256 apiWsMacSha256 = new ApiWsMacSha256();
```

3. Calcular el parámetro **<Ds_Signature>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createSignatureResponseHostToHost()" con la clave obtenida del módulo de administración, la cadena que se desea firmar(concatenación de campos descrita en el punto 2 del apartado 4.1 del presente documento) y el número de pedido.

```
String cadenaConcatenada="145144491278999900888197800000 0";
String numPedido="1444912789";
String claveModuloAdmin = "Mk9m98IfEblmPfrpsawt7BmxObt98Jev";
String signature = apiWsMacSha256.createSignatureResponseHostToHost(claveModuloAdmin,
                                                                    cadenaConcatenada,
                                                                    numPedido);
```

El resultado obtenido debe ser el mismo que el valor del parámetro **<Ds_Signature>** obtenido en la respuesta.

### 4.2.3 Librería .NET

A continuación se presentan los pasos que debe seguir un comercio para la utilización de la librería .NET proporcionada por Redsys:

1. Convertir la cadena respuesta XML al atributo diccionario m_keyvalues de la clave RedsysAPIWs:

```
r.XMLToDiccionary(result);
```

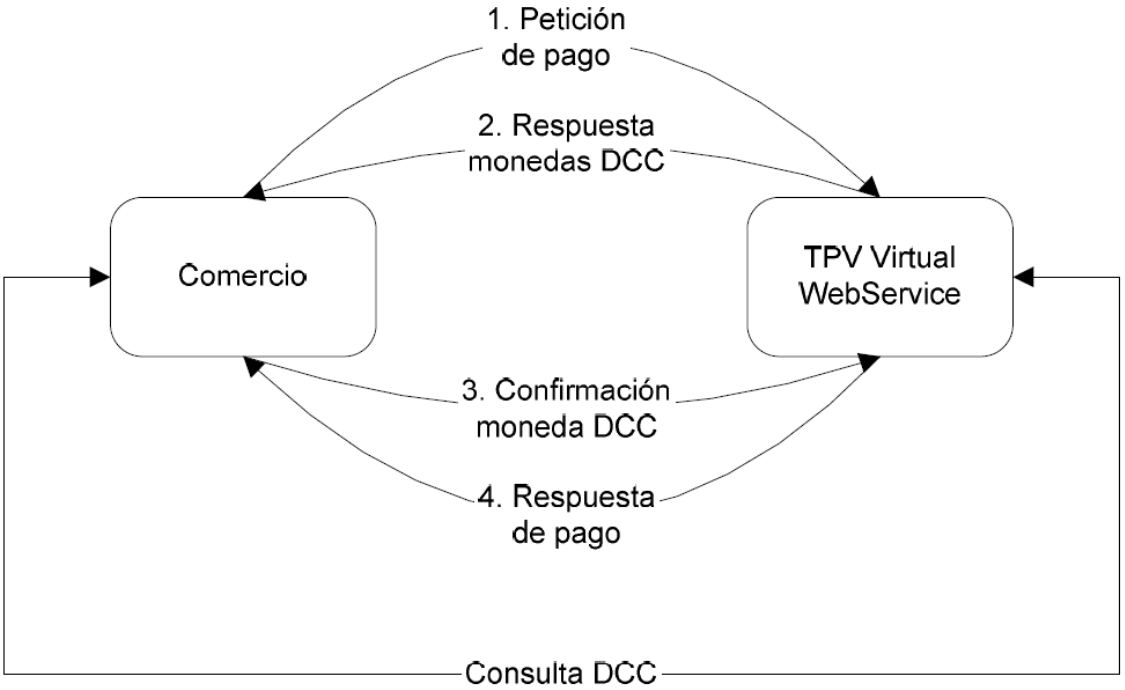
2. Calcular el parámetro **<Ds_Signature>**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createSignatureResponseHostToHost()" con la clave obtenida del módulo de administración, la cadena que se desea firmar(concatenación de campos descrita en el punto 2 del apartado 5.1 del presente documento) y el número de pedido.

```
string cadena = r.GenerateCadena(result);
string numOrder = r.GetDictionary("Ds_Order");
string signatureCalculate = r.createSignatureResponseHostToHost(kc, cadena, numOrder);
```

El resultado obtenido debe ser el mismo que el valor del parámetro **<Ds_Signature>** obtenido en la respuesta.

# 5. Operativa DCC

A continuación se detallarán todas aquellas características adicionales de la operativa DCC que tengan los comercios que hayan contratado este servicio.




**NOTA:** Como se muestra en el gráfico la operativa DCC se basa en el envío de dos peticiones al WebService del TPV Virtual. Para garantizar el correcto funcionamiento del sistema, es necesario que el comercio mantenga la sesión entre la primera y la segunda llamada al WebService. El mantenimiento de la sesión dependerá del software utilizado para realizar la llamada al WebService. Por ejemplo si se utiliza el API de Axis, será suficiente con utilizar el mismo "Stub" para las dos peticiones y fijar la propiedad setMaintainSession(true) antes de realizar la primera llamada.

## 5.1 Métodos de acceso

El método de acceso "trataPeticion": permite la realización de operaciones a través del TPV Virtual WebService. Se usa el mismo método tanto para realizar los pagos tradicionales como para la operativa DCC y, en función de los campos que se remitan en el XML de petición, se realizará una u otra opción.

El método de acceso "consultaDCC": permite hacer consultas del DCC asociado a un importe y una moneda con anterioridad a ejecutar la transacción. Es meramente informativo.

## 5.2 Mensaje de petición inicial

El mensaje de petición inicial (1. Petición de pago) posee las mismas características que lo descrito anteriormente en el apartado 3.1 de la guía del TPV Virtual WebService.

### 5.2.1 Ejemplo de petición inicial

```
<REQUEST>
    <DATOSENTRADA>
        <DS_MERCHANT_AMOUNT>145</DS_MERCHANT_AMOUNT>
        <DS_MERCHANT_ORDER>1444904795</DS_MERCHANT_ORDER>

<DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
        <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
        <DS_MERCHANT_PAN>XXXXXXXXXXXXXXXX</DS_MERCHANT_PAN>
        <DS_MERCHANT_CVV2>XXX</DS_MERCHANT_CVV2>
        <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
        <DS_MERCHANT_TERMINAL>6</DS_MERCHANT_TERMINAL>
        <DS_MERCHANT_EXPIRYDATE>XXXX</DS_MERCHANT_EXPIRYDATE>
    </DATOSENTRADA>
    <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>

<DS_SIGNATURE>7ZEurW0QJVbRcCHf23kM3vNh50dDQAvu8HLmiJ/eDQA=</DS_SIGNA
TURE>
</REQUEST>
```

## 5.3 Mensaje de respuesta DCC

A continuación se describen los datos necesarios y sus características, que se recibirán en los mensajes de respuesta DCC (2. Respuesta monedas DCC) del TPV Virtual en el formato XML descrito anteriormente para la operativa DCC y que sirven como ejemplo para la posterior confirmación DCC.

| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| **CAMPOS ESPECÍFICOS DE LA OPERTIVA DCC** | | |
| *moneda* | *3 / N* | Obligatorio. Valor del identificador de la moneda (ISO-4217). |
| *litMoneda* | *- / A* | Obligatorio. Literal asociado a la moneda. |
| *litMonedaR* | *3 / R* | Obligatorio. Literal reducida asociado a la moneda. |
| *cambio* | *- / N* | Obligatorio. Valor del cambio de la moneda. |
| *importe* | *- / N* | Obligatorio. Importe en la moneda. |
| *checked* | *true/false* | Obligatorio. Indica divisa comprobada. |
| *margenDCC* | *- / N* | Obligatorio. Margen DCC aplicado por la entidad al importe. |
| *nombreEntidad* | *- / A* | Obligatorio. Nombre de la entidad bancaria que aplica el DCC. |
| *DS_MERCHANT_SESION* | *- / AN* | Obligatorio. Identificador de la sesión para continuar la operación en operativas DCC. |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

### 5.3.1 Ejemplo de respuesta DCC

```
<RETORNOXML>
    <CODIGO>0</CODIGO>
    <DCC>
        <moneda>826</moneda>
        <litMoneda>POUND STERLING</litMoneda>
        <litMonedaR>GBP</litMonedaR>
        <cambio>1.369986</cambio>
        <fechaCambio>2015-06-16</fechaCambio>
        <importe>1.06</importe>
        <checked>true</checked>
    </DCC>
    <DCC>
        <moneda>978</moneda>
        <litMoneda>Euros</litMoneda>
        <importe>1.45</importe>
    </DCC>
    <margenDCC>0.03</margenDCC>
    <nombreEntidad>SIN CAPTURA</nombreEntidad>

<DS_MERCHANT_SESION>0b33f991d36b28cf643769691f3e86140fa7d1e4</DS_MER
CHANT_SESION>
</RETORNOXML>
```

## 5.4 Mensaje de confirmación DCC

A continuación se describen los datos necesarios y sus características para enviar una petición de confirmación DCC (3. Confirmación moneda DCC) al TPV Virtual Webservice en el formato y que sirven como ejemplo para confirmar el anterior mensaje de petición DCC.

| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| **CAMPOS ADICIONALES REQUERIDOS EN LA SEGUNDA PETICIÓN** | | |
| *Sis_Divisa* | *16/A-N* | **Obligatorio**. Dos valores separados por #. El primero es el identificador de la moneda (ISO-4217), el segundo el importe en dicha moneda. |
| *DS_MERCHANT_SESION* | | **Obligatorio**. Identificador de la sesión para continuar la operación en operativas DCC. |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

### 5.4.1 Ejemplo de mensaje de confirmación de moneda DCC

```
<REQUEST>
    <DATOSENTRADA>
        <DS_MERCHANT_ORDER>1444904795</DS_MERCHANT_ORDER>

<DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
        <DS_MERCHANT_TERMINAL>6</DS_MERCHANT_TERMINAL>
        <Sis_Divisa>978#1.06</Sis_Divisa>

<DS_MERCHANT_SESION>0b33f991d36b28cf643769691f3e86140fa7d1e4</DS_MER
CHANT_SESION>
    </DATOSENTRADA>
    <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>

<DS_SIGNATURE>yUBgA1G0UIT6CtnwoG1PJrTtzNXEtD24vsS8nIH6KR8=</DS_SIGNA
TURE>
</REQUEST>
```

## 5.5 Mensaje de respuesta a confirmación de moneda DCC

El mensaje de respuesta (4. Respuesta de pago) posee las mismas características que lo descrito anteriormente en el apartado 3.2 de la guía del TPV Virtual WebService.

### 5.5.1 Ejemplo de respuesta a confirmación de moneda DCC

```
<RETORNOXML>
    <CODIGO>0</CODIGO>
    <OPERACION>
        <Ds_Amount>145</Ds_Amount>
        <Ds_Currency>978</Ds_Currency>
        <Ds_Order>1444904795</Ds_Order>

<Ds_Signature>EtP9XhoR0njz7QDTzu7C0FB7+KJGI68s0YlKqMsse00=</Ds_Signa
ture>
        <Ds_MerchantCode>999008881</Ds_MerchantCode>
        <Ds_Terminal>6</Ds_Terminal>
        <Ds_Response>0909</Ds_Response>
        <Ds_AuthorisationCode/>
        <Ds_TransactionType>0</Ds_TransactionType>
        <Ds_SecurePayment>0</Ds_SecurePayment>
        <Ds_Language>1</Ds_Language>
        <Ds_MerchantData/>
        <Ds_Card_Country>826</Ds_Card_Country>
    </OPERACION>
</RETORNOXML>
```

## 5.6 Mensaje de consulta DCC

El mensaje de consulta DCC se generará con los datos anteriormente descritos en un XML que se mandará al método consultaDCC. Esta consulta es solamente informativa.

### 5.6.1 Ejemplo de mensaje de consulta DCC

```
<REQUEST>
    <DATOSENTRADA>
        <DS_MERCHANT_AMOUNT>1.06</DS_MERCHANT_AMOUNT>
        <DS_MERCHANT_ORDER>1444904795</DS_MERCHANT_ORDER>

<DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
        <DS_MERCHANT_TERMINAL>6</DS_MERCHANT_TERMINAL>
        <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
    </DATOSENTRADA>
    <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>

<DS_SIGNATURE>oVGAkwOQNYHqDN8+i2oBRKYn8aZR4s7LJOcHpwnuCoU=</DS_SIGNA
TURE>
</REQUEST>
```

## 5.7 Mensaje de respuesta de consulta DCC

El mensaje de consulta DCC se generará con los datos anteriormente descritos en un XML que se mandará al método **consultaDCC**.

### 5.7.1 Ejemplo de mensaje de respuesta consulta DCC

```
<RETORNOXML>
    <CODIGO>0</CODIGO>
    <DCC>
        <moneda>978</moneda>
        <importe>0.01</importe>
    </DCC>
    <margenDCC>0.03</margenDCC>
    <nombreEntidad>SIN CAPTURA</nombreEntidad>
</RETORNOXML>
```

## 5.8 Firma del comercio

Independientemente de la petición, la firma del comercio se calcula realizando el nuevo procedimiento de firma HMAC_SHA256, el cual se especifica en el punto 3.4 de este mismo documento.

# 6. Realización de Pruebas

Existe un entorno de test que permite realizar las pruebas necesarias para verificar el correcto funcionamiento del sistema antes de hacer la implantación en el entorno real.

A continuación se proporcionarán las URL de acceso al portal de administración y el endpoint del servicio web para realizar las pruebas. Para obtener los datos de acceso para su comercio, deberá dirigirse a su entidad bancaria para que les proporcione los datos de acceso.

La URL para el envío de las órdenes de pago por entrada WebService es la siguiente:

https://sis-t.redsys.es:25443/sis/services/SerClsWSEntrada

Adicionalmente, la URL para el acceso al módulo de administración es la siguiente:

https://sis-t.redsys.es:25443/canales

*El entorno de pruebas será idéntico al entorno real, con la única diferencia que todos los pagos realizados en este entorno no tendrán validez contable.

Desde Redsys se proporcionan unos datos genéricos de prueba para todos los clientes. Como ya se ha indicado, para obtener los datos específicos de su comercio, deberá contactar con su entidad bancaria.

**DATOS GENÉRICOS DE PRUEBA**

- Número de comercio (Ds_Merchant_MerchantCode): 999008881
- Terminal (Ds_Merchant_Terminal): 001
- Clave secreta: sq7HjrUOBfKmC576ILgskD5srU870gJ7
- Tarjeta aceptada:
    - Numeración: 4548 8120 4940 0004
    - Caducidad: 12/20
    - Código CVV2: 123

    En modo de compra segura (CES), en la que se requiera autenticación del comprador, el código de identificación personal (CIP) es: 123456

# 7. Códigos de error

En este apartado se presenta un glosario de los errores que se pueden producir en el proceso de integración.

## 7.1 Glosario de errores del SIS

| ERROR | DESCRIPCIÓN | MENSAJE |
|---|---|---|
| **SIS0007** | Error al desmontar el XML de entrada | MSG0008 |
| **SIS0008** | Error falta Ds_Merchant_MerchantCode | MSG0008 |
| **SIS0009** | Error de formato en Ds_Merchant_MerchantCode | MSG0008 |
| **SIS0010** | Error falta Ds_Merchant_Terminal | MSG0008 |
| **SIS0011** | Error de formato en Ds_Merchant_Terminal | MSG0008 |
| **SIS0014** | Error de formato en Ds_Merchant_Order | MSG0008 |
| **SIS0015** | Error falta Ds_Merchant_Currency | MSG0008 |
| **SIS0016** | Error de formato en Ds_Merchant_Currency | MSG0008 |
| **SIS0017** | Error no se admiten operaciones en pesetas | MSG0008 |
| **SIS0018** | Error falta Ds_Merchant_Amount | MSG0008 |
| **SIS0019** | Error de formato en Ds_Merchant_Amount | MSG0008 |
| **SIS0020** | Error falta Ds_Merchant_MerchantSignature | MSG0008 |
| **SIS0021** | Error la Ds_Merchant_MerchantSignature viene vacía | MSG0008 |
| **SIS0022** | Error de formato en Ds_Merchant_TransactionType | MSG0008 |
| **SIS0023** | Error Ds_Merchant_TransactionType desconocido | MSG0008 |
| **SIS0024** | Error Ds_Merchant_ConsumerLanguage tiene mas de 3 posiciones | MSG0008 |
| **SIS0025** | Error de formato en Ds_Merchant_ConsumerLanguage | MSG0008 |
| **SIS0026** | Error No existe el comercio / terminal enviado | MSG0008 |
| **SIS0027** | Error Moneda enviada por el comercio es diferente a la que tiene asignada para ese terminal | MSG0008 |
| **SIS0028** | Error Comercio / terminal está dado de baja | MSG0008 |
| **SIS0030** | Error en un pago con tarjeta ha llegado un tipo de operación que no es ni pago ni preautorización | MSG0000 |
| **SIS0031** | Método de pago no definido | MSG0000 |
| **SIS0033** | Error en un pago con móvil ha llegado un tipo de operación que no es ni pago ni preautorización | MSG0000 |
| **SIS0034** | Error de acceso a la Base de Datos | MSG0000 |
| **SIS0037** | El número de teléfono no es válido | MSG0000 |

| ERROR | DESCRIPCIÓN | MENSAJE |
|---|---|---|
| **SIS0038** | Error en java | MSG0000 |
| **SIS0040** | Error el comercio / terminal no tiene ningún método de pago asignado | MSG0008 |
| **SIS0041** | Error en el cálculo de la HASH de datos del comercio. | MSG0008 |
| **SIS0042** | La firma enviada no es correcta | MSG0008 |
| **SIS0043** | Error al realizar la notificación on-line | MSG0008 |
| **SIS0046** | El bin de la tarjeta no está dado de alta | MSG0002 |
| **SIS0051** | Error número de pedido repetido | MSG0001 |
| **SIS0054** | Error no existe operación sobre la que realizar la devolución | MSG0008 |
| **SIS0055** | Error existe más de un pago con el mismo número de pedido | MSG0008 |
| **SIS0056** | La operación sobre la que se desea devolver no está autorizada | MSG0008 |
| **SIS0057** | El importe a devolver supera el permitido | MSG0008 |
| **SIS0058** | Inconsistencia de datos, en la validación de una confirmación | MSG0008 |
| **SIS0059** | Error no existe operación sobre la que realizar la confirmación | MSG0008 |
| **SIS0060** | Ya existe una confirmación asociada a la preautorización | MSG0008 |
| **SIS0061** | La preautorización sobre la que se desea confirmar no está autorizada | MSG0008 |
| **SIS0062** | El importe a confirmar supera el permitido | MSG0008 |
| **SIS0063** | Error. Número de tarjeta no disponible | MSG0008 |
| **SIS0064** | Error. El número de tarjeta no puede tener más de 19 posiciones | MSG0008 |
| **SIS0065** | Error. El número de tarjeta no es numérico | MSG0008 |
| **SIS0066** | Error. Mes de caducidad no disponible | MSG0008 |
| **SIS0067** | Error. El mes de la caducidad no es numérico | MSG0008 |
| **SIS0068** | Error. El mes de la caducidad no es válido | MSG0008 |
| **SIS0069** | Error. Año de caducidad no disponible | MSG0008 |
| **SIS0070** | Error. El Año de la caducidad no es numérico | MSG0008 |
| **SIS0071** | Tarjeta caducada | MSG0000 |
| **SIS0072** | Operación no anulable | MSG0000 |
| **SIS0074** | Error falta Ds_Merchant_Order | MSG0008 |
| **SIS0075** | Error el Ds_Merchant_Order tiene menos de 4 posiciones o más de 12 | MSG0008 |
| **SIS0076** | Error el Ds_Merchant_Order no tiene las cuatro primeras posiciones numéricas | MSG0008 |
| **SIS0077** | Error el Ds_Merchant_Order no tiene las cuatro primeras posiciones numéricas. No se utiliza | MSG0000 |
| **SIS0078** | Método de pago no disponible | MSG0005 |
| **SIS0079** | Error al realizar el pago con tarjeta | MSG0000 |

| ERROR | DESCRIPCIÓN | MENSAJE |
|---|---|---|
| **SIS0081** | La sesión es nueva, se han perdido los datos almacenados | MSG0007 |
| **SIS0084** | El valor de Ds_Merchant_Conciliation es nulo | MSG0008 |
| **SIS0085** | El valor de Ds_Merchant_Conciliation no es numérico | MSG0008 |
| **SIS0086** | El valor de Ds_Merchant_Conciliation no ocupa 6 posiciones | MSG0008 |
| **SIS0089** | El valor de Ds_Merchant_ExpiryDate no ocupa 4 posiciones | MSG0008 |
| **SIS0092** | El valor de Ds_Merchant_ExpiryDate es nulo | MSG0008 |
| **SIS0093** | Tarjeta no encontrada en la tabla de rangos | MSG0006 |
| **SIS0094** | La tarjeta no fue autenticada como 3D Secure | MSG0004 |
| **SIS0097** | Valor del campo Ds_Merchant_CComercio no válido | MSG0008 |
| **SIS0098** | Valor del campo Ds_Merchant_CVentana no válido | MSG0008 |
| **SIS0112** | Error El tipo de transacción especificado en Ds_Merchant_Transaction_Type no esta permitido | MSG0008 |
| **SIS0113** | Excepción producida en el servlet de operaciones | MSG0008 |
| **SIS0114** | Error, se ha llamado con un GET en lugar de un POST | MSG0000 |
| **SIS0115** | Error no existe operación sobre la que realizar el pago de la cuota | MSG0008 |
| **SIS0116** | La operación sobre la que se desea pagar una cuota no es una operación válida | MSG0008 |
| **SIS0117** | La operación sobre la que se desea pagar una cuota no está autorizada | MSG0008 |
| **SIS0118** | Se ha excedido el importe total de las cuotas | MSG0008 |
| **SIS0119** | Valor del campo Ds_Merchant_DateFrecuency no válido | MSG0008 |
| **SIS0120** | Valor del campo Ds_Merchant_ChargeExpiryDate no válido | MSG0008 |
| **SIS0121** | Valor del campo Ds_Merchant_SumTotal no válido | MSG0008 |
| **SIS0122** | Valor del campo Ds_Merchant_DateFrecuency o no Ds_Merchant_SumTotal tiene formato incorrecto | MSG0008 |
| **SIS0123** | Se ha excedido la fecha tope para realizar transacciones | MSG0008 |
| **SIS0124** | No ha transcurrido la frecuencia mínima en un pago recurrente sucesivo | MSG0008 |
| **SIS0132** | La fecha de Confirmación de Autorización no puede superar en más de 7 días a la de Preautorización. | MSG0008 |
| **SIS0133** | La fecha de Confirmación de Autenticación no puede superar en más de 45 días a la de Autenticación Previa. | MSG0008 |
| **SIS0139** | Error el pago recurrente inicial está duplicado | MSG0008 |
| **SIS0142** | Tiempo excedido para el pago | MSG0000 |
| **SIS0197** | Error al obtener los datos de cesta de la compra en operación tipo pasarela | MSG0000 |
| **SIS0198** | Error el importe supera el límite permitido para el comercio | MSG0000 |
| **SIS0199** | Error el número de operaciones supera el límite permitido para el comercio | MSG0008 |
| **SIS0200** | Error el importe acumulado supera el límite permitido para el comercio | MSG0008 |
| **SIS0214** | El comercio no admite devoluciones | MSG0008 |

| ERROR | DESCRIPCIÓN | MENSAJE |
|---|---|---|
| **SIS0216** | Error Ds_Merchant_CVV2 tiene más de 3 posiciones | MSG0008 |
| **SIS0217** | Error de formato en Ds_Merchant_CVV2 | MSG0008 |
| **SIS0218** | El comercio no permite operaciones seguras por la entrada /operaciones | MSG0008 |
| **SIS0219** | Error el número de operaciones de la tarjeta supera el límite permitido para el comercio | MSG0008 |
| **SIS0220** | Error el importe acumulado de la tarjeta supera el límite permitido para el comercio | MSG0008 |
| **SIS0221** | Error el CVV2 es obligatorio | MSG0008 |
| **SIS0222** | Ya existe una anulación asociada a la preautorización | MSG0008 |
| **SIS0223** | La preautorización que se desea anular no está autorizada | MSG0008 |
| **SIS0224** | El comercio no permite anulaciones por no tener firma ampliada | MSG0008 |
| **SIS0225** | Error no existe operación sobre la que realizar la anulación | MSG0008 |
| **SIS0226** | Inconsistencia de datos, en la validación de una anulación | MSG0008 |
| **SIS0227** | Valor del campo Ds_Merchant_TransactionDate no válido | MSG0008 |
| **SIS0229** | No existe el código de pago aplazado solicitado | MSG0008 |
| **SIS0252** | El comercio no permite el envío de tarjeta | MSG0008 |
| **SIS0253** | La tarjeta no cumple el check-digit | MSG0006 |
| **SIS0254** | El número de operaciones de la IP supera el límite permitido por el comercio | MSG0008 |
| **SIS0255** | El importe acumulado por la IP supera el límite permitido por el comercio | MSG0008 |
| **SIS0256** | El comercio no puede realizar preautorizaciones | MSG0008 |
| **SIS0257** | Esta tarjeta no permite operativa de preautorizaciones | MSG0008 |
| **SIS0258** | Inconsistencia de datos, en la validación de una confirmación | MSG0008 |
| **SIS0261** | Operación detenida por superar el control de restricciones en la entrada al SIS | MSG0008 |
| **SIS0270** | El comercio no puede realizar autorizaciones en diferido | MSG0008 |
| **SIS0274** | Tipo de operación desconocida o no permitida por esta entrada al SIS | MSG0008 |
| **SIS0429** | Error en la versión enviada por el comercio en el parámetro Ds_SignatureVersion | MSG0008 |
| **SIS0432** | Error FUC del comercio erróneo | MSG0008 |
| **SIS0433** | Error Terminal del comercio erróneo | MSG0008 |
| **SIS0434** | Error ausencia de número de pedido en la operación enviada por el comercio | MSG0008 |
| **SIS0435** | Error en el cálculo de la firma | MSG0008 |
| **SIS0436** | Error en la construcción del elemento padre <REQUEST> | MSG0008 |
| **SIS0437** | Error en la construcción del elemento <DS_SIGNATUREVERSION> | MSG0008 |
| **SIS0438** | Error en la construcción del elemento <DATOSENTRADA> | MSG0008 |

| ERROR | DESCRIPCIÓN | MENSAJE |
|---|---|---|
| **SIS0439** | Error en la construcción del elemento <DS_SIGNATURE> | MSG0008 |
| **SIS0444** | Error producido al acceder mediante un sistema de firma antiguo teniendo configurado el tipo de clave HMAC SHA256 | MSG0008 |
| **SIS0462** | Error, se aplica el método de pago seguro en operación Host to Host | MSG0008 |
| **SIS0469** | Error, no se ha superado el proceso de control de fraude. | MSG0008 |
| **SIS0487** | Error, el comercio no tiene el método de pago Paygold | MSG0008 |
| **SIS0488** | Error, el comercio no tiene el método de pago Pago Manual | MSG0008 |

# 8. ANEXOS

## 8.1 Peticiones de pago (con envío de datos de tarjeta)

En el presente anexo se describen los datos necesarios y sus características, para enviar una petición al Web Service de Redsys en formato XML. Así mismo se incluye un ejemplo de cómo utilizar esos datos en los mensajes de petición de pago.

| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| *DS_MERCHANT_AMOUNT* | *12 / N* | **Obligatorio**. Las dos últimas posiciones se consideran decimales, salvo en el caso de los Yenes que no tienen. |
| *DS_MERCHANT_ORDER* | *12 / A-N* | **Obligatorio**. Número de pedido. Los 4 primeros dígitos deben ser numéricos. Cada pedido es único, no puede repetirse. |
| *DS_MERCHANT_MERCHANTCODE* | *9 / N* | **Obligatorio**. Código FUC asignado al comercio. |
| *DS_MERCHANT_TERMINAL* | *3 / N* | **Obligatorio**. Número de Terminal que le asignará su banco. Por defecto valor "001". 3 se considera su longitud máxima. |
| *DS_MERCHANT_CURRENCY* | *4 / N* | **Obligatorio**. Moneda del comercio. Tiene que ser la contratada para el Terminal.<br>Valor 978 para Euros, 840 para Dólares, 826 para Libras esterlinas y 392 para Yenes. |
| *DS_MERCHANT_PAN* | *19 / N* | **Obligatorio**. Tarjeta. Su longitud depende del tipo de tarjeta. |
| *DS_MERCHANT_EXPIRYDATE* | *4 / N* | **Obligatorio**. Caducidad de la tarjeta. Su formato es AAMM, siendo AA los dos últimos dígitos del año y MM los dos dígitos del mes. |
| *DS_MERCHANT_CVV2* | *3-4 / N* | **Obligatorio**. Código CVV2 de la tarjeta. |
| *DS_MERCHANT_TRANSACTIONTYPE* | *1 / A-N* | **Obligatorio**. Campo para el comercio para indicar qué tipo de transacción es. Posibles valores:<br>0 – Autorización<br>1 – Preautorización |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

A continuación se muestra un ejemplo de un mensaje de petición de pago:

```
<DATOSENTRADA>
    <DS_MERCHANT_AMOUNT>145</DS_MERCHANT_AMOUNT>
    <DS_MERCHANT_ORDER>050911523002</DS_MERCHANT_ORDER>
    <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCO
    DE>
    <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
    <DS_MERCHANT_PAN>XXXXXXXXXXXXX</DS_MERCHANT_PAN>
    <DS_MERCHANT_CVV2>XXX</DS_MERCHANT_CVV2>
    <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
    <DS_MERCHANT_TERMINAL>999</DS_MERCHANT_TERMINAL>
    <DS_MERCHANT_EXPIRYDATE>XXXX</DS_MERCHANT_EXPIRYDATE>
</DATOSENTRADA>
```

## 8.2 Peticiones de Confirmación/Devolución

En el presente anexo se describen los datos necesarios y sus características, para enviar una petición al Web Service de Redsys en formato XML. Así mismo se incluye un ejemplo de cómo utilizar esos datos en los mensajes de petición de pago.

| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| *DS_MERCHANT_AMOUNT* | *12 / N* | **Obligatorio**. Las dos últimas posiciones se consideran decimales, salvo en el caso de los Yenes que no tienen. |
| *DS_MERCHANT_ORDER* | *12 / A-N* | **Obligatorio**. Número de pedido. Los 4 primeros dígitos deben ser numéricos. Cada pedido es único, no puede repetirse. |
| *DS_MERCHANT_MERCHANTCODE* | *9 / N* | **Obligatorio**. Código FUC asignado al comercio. |
| *DS_MERCHANT_TERMINAL* | *3 / N* | **Obligatorio**. Número de Terminal que le asignará su banco. Por defecto valor "001". 3 se considera su longitud máxima. |
| *DS_MERCHANT_CURRENCY* | *4 / N* | **Obligatorio**. Moneda del comercio. Tiene que ser la contratada para el Terminal.<br>Valor 978 para Euros, 840 para Dólares, 826 para Libras esterlinas y 392 para Yenes. |
| *DS_MERCHANT_TRANSACTIONTYPE* | *1 / A-N* | **Obligatorio**. Campo para el comercio para indicar qué tipo de transacción es. Los posibles valores son:<br>2 – Confirmación<br>3 – Devolución<br>8 – Confirmación separada<br>9 – Anulación de Preautorización |
| *DS_MERCHANT_AUTHORISATIONCODE* | *6 / Num* | **Opcional**. Representa el código de autorización necesario para identificar una transacción recurrente sucesiva en las devoluciones de operaciones recurrentes sucesivas. Obligatorio en devoluciones de operaciones recurrentes. |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

A continuación se muestra un ejemplo de un mensaje de petición de pago recurrente:

```
<DATOSENTRADA>
    <DS_MERCHANT_AMOUNT>145</DS_MERCHANT_AMOUNT>
    <DS_MERCHANT_ORDER>050911523002</DS_MERCHANT_ORDER>
    <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
    <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
    <DS_MERCHANT_TRANSACTIONTYPE>3</DS_MERCHANT_TRANSACTIONTYPE>
    <DS_MERCHANT_TERMINAL>999</DS_MERCHANT_TERMINAL>
</DATOSENTRADA>
```

## 8.3 Peticiones de Pago por Referencia (Pago 1-Clic)

En el presente anexo se describen los datos necesarios y sus características, para enviar una petición Host to Host en formato XML. Así mismo se incluye un ejemplo de cómo utilizar esos datos en los mensajes de petición de pago.

| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| *DS_MERCHANT_AMOUNT* | *12 / N* | **Obligatorio**. Las dos últimas posiciones se consideran decimales, salvo en el caso de los Yenes que no tienen. |
| *DS_MERCHANT_ORDER* | *12 / A-N* | **Obligatorio**. Número de pedido. Los 4 primeros dígitos deben ser numéricos. Cada pedido es único, no puede repetirse. |
| *DS_MERCHANT_MERCHANTCODE* | *9 / N* | **Obligatorio**. Código FUC asignado al comercio. |
| *DS_MERCHANT_TERMINAL* | *3 / N* | **Obligatorio**. Número de Terminal que le asignará su banco. Por defecto valor "001".3 se considera su longitud máxima. |
| *DS_MERCHANT_CURRENCY* | *4 / N* | **Obligatorio**. Moneda del comercio. Tiene que ser la contratada para el Terminal.<br>Valor 978 para Euros, 840 para Dólares, 826 para Libras esterlinas y 392 para Yenes. |
| *DS_MERCHANT_PAN* | *19 / N* | **Obligatorio**. Tarjeta. Su longitud depende del tipo de tarjeta. |
| *DS_MERCHANT_EXPIRYDATE* | *4 / N* | **Obligatorio**. Caducidad de la tarjeta. Su formato es AAMM, siendo AA los dos últimos dígitos del año y MM los dos dígitos del mes. |
| *DS_MERCHANT_CVV2* | *3-4 / N* | **Obligatorio**. Código CVV2 de la tarjeta. |
| *DS_MERCHANT_TRANSACTIONTYPE* | *1 / A-N* | **Obligatorio**. Campo para el comercio para indicar qué tipo de transacción es. Los posibles valores son:<br>0 – Autorización<br>1 – Preautorización |
| *DS_MERCHANT_IDENTIFIER* | *8/N* | **Obligatorio**. Su uso se especifica en los ejemplos de pago por Referencia o Pago 1-Clic |
| *DS_MERCHANT_GROUP* | *9/N* | **Opcional**. Su uso se especifica en los ejemplos de pago por Referencia o Pago 1-Clic |
| *DS_MERCHANT_DIRECTPAYMENT* | *4/N* | **Opcional**. Su uso se especifica en los ejemplos de pago por Referencia o Pago 1-Clic |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

A continuación se muestra un ejemplo de un mensaje de petición de pago:

```
<DATOSENTRADA>
    <DS_MERCHANT_AMOUNT>145</DS_MERCHANT_AMOUNT>
    <DS_MERCHANT_ORDER>050911523002</DS_MERCHANT_ORDER>
    <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCO
    DE>
    <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
    <DS_MERCHANT_PAN>XXXXXXXXXXXXX</DS_MERCHANT_PAN>
    <DS_MERCHANT_CVV2>XXX</DS_MERCHANT_CVV2>
    <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
    <DS_MERCHANT_TERMINAL>999</DS_MERCHANT_TERMINAL>
    <DS_MERCHANT_EXPIRYDATE>XXXX</DS_MERCHANT_EXPIRYDATE>
    < DS_MERCHANT_IDENTIFIER>REQUIRED</ DS_MERCHANT_IDENTIFIER>
</DATOSENTRADA>
```

## 8.4 Respuesta Host to Host

A continuación se presenta una tabla que recoge todos los parámetros que forman parte de la respuesta del Web Service.

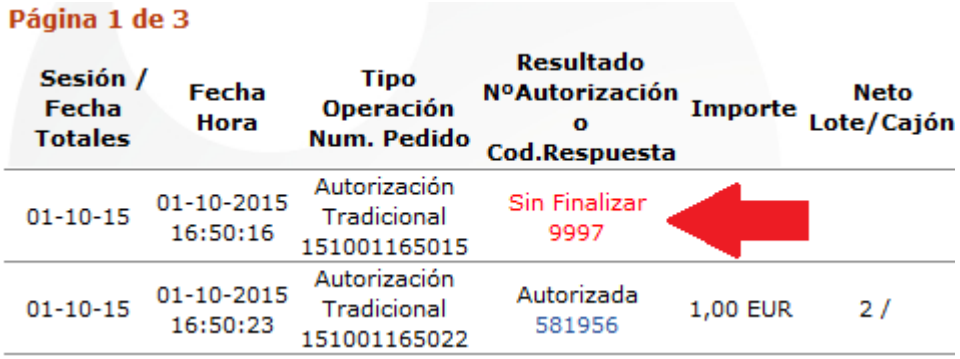
| Nombre del dato | Long. / Tipo | Descripción |
|---|---|---|
| *CODIGO* | | **Obligatorio**. Indica si la operación ha sido correcta o no, (no indica si ha sido autorizada, solo si se ha procesado). Un 0 indica que la operación ha sido correcta. En el caso de que sea distinto de 0, tendrá un código. (Ver códigos de error en apartado 5 de esta Guía) |
| *Ds_Amount* | *12 / A-N* | **Obligatorio**. Para Euros las dos últimas posiciones se consideran decimales, salvo en el caso de los Yenes que no tienen. |
| *Ds_Currency* | *4 / N* | **Obligatorio**. Moneda del comercio. |
| *Ds_Order* | *12 /A- N* | **Obligatorio**. Número de pedido. |
| *Ds_Signature* | *40 / A-N* | **Obligatorio**. Firma del comercio. |
| *Ds_MerchantCode* | *9 / N* | **Obligatorio**. Código FUC asociado al comercio. |
| *Ds_Terminal* | *3 / N* | **Obligatorio**. Número de Terminal del comercio. |
| *Ds_Response* | *4 / N* | **Obligatorio**. Valor que indica el resultado de la operación. **Indicará si ha sido autorizada o no**. Los posibles valores de este campo se describen en la siguiente tabla. |
| *Ds_AuthorisationCode* | *6 / N* | **Optativo**. Código de autorización en caso de existir para las operaciones autorizadas. |
| *Ds_TransactionType* | *1 / A-N* | **Obligatorio**. Indica qué tipo de transacción se ha realizado. Los posibles valores son:<br>0 – Autorización<br>1 – Preautorización<br>2 – Confirmación<br>3 – Devolución Automática<br>8 – Confirmación separada<br>9 – Anulación de Preautorización |
| *Ds_SecurePayment* | | **Obligatorio**. Indica si el pago ha sido seguro o no:<br>• 0: seguro (no se aplica)<br>• 1: no seguro. |
| *Ds_Language* | *1 / N* | **Obligatorio**. Idioma. |
| *Ds_Card_Brand* | *2 / N* | **Opcional.** No se debe hacer una validación cerrada sobre estos valores, pues pueden variar y/o añadise nuevos. Valores posibles:<br>1 – VISA<br>2 –MASTERCARD<br>8 – AMEX<br>9 – JCB<br>22 – UPI<br>6 – DINERS<br>22 – CUP<br>7 - PRIVADA |
| **Tipo A**: caracteres ASCII del 65 = **A** al 90 = **Z** y del 97 = **a** al 122 = **z**.<br>**Tipo N**: caracteres ASCII del 30 = **0** al 39 = **9**. | | |

Estos son los posibles valores del Ds_Response o "Código de respuesta":

| CÓDIGO | SIGNIFICADO |
|---|---|
| **0000 a 0099** | **Transacción autorizada para pagos y preautorizaciones** |
| **900** | **Transacción autorizada para devoluciones y confirmaciones** |
| **400** | **Transacción autorizada para anulaciones** |
| 101 | Tarjeta caducada |
| 102 | Tarjeta en excepción transitoria o bajo sospecha de fraude |
| 106 | Intentos de PIN excedidos |
| 125 | Tarjeta no efectiva |
| 129 | Código de seguridad (CVV2/CVC2) incorrecto |
| 180 | Tarjeta ajena al servicio |
| 184 | Error en la autenticación del titular |
| 190 | Denegación del emisor sin especificar motivo |
| 191 | Fecha de caducidad errónea |
| 202 | Tarjeta en excepción transitoria o bajo sospecha de fraude con retirada de tarjeta |
| 904 | Comercio no registrado en FUC |
| 909 | Error de sistema |
| 913 | Pedido repetido |
| 944 | Sesión Incorrecta |
| 950 | Operación de devolución no permitida |
| 9912/912 | Emisor no disponible |
| 9064 | Número de posiciones de la tarjeta incorrecto |
| 9078 | Tipo de operación no permitida para esa tarjeta |
| 9093 | Tarjeta no existente |
| 9094 | Rechazo servidores internacionales |
| 9104 | Comercio con "titular seguro" y titular sin clave de compra segura |
| 9218 | El comercio no permite op. seguras por entrada /operaciones |
| 9253 | Tarjeta no cumple el check-digit |
| 9256 | El comercio no puede realizar preautorizaciones |
| 9257 | Esta tarjeta no permite operativa de preautorizaciones |
| 9261 | Operación detenida por superar el control de restricciones en la entrada al SIS |
| 9913 | Error en la confirmación que el comercio envía al TPV Virtual (solo aplicable en la opción de sincronización SOAP) |
| 9914 | Confirmación "KO" del comercio (solo aplicable en la opción de sincronización SOAP) |
| 9915 | A petición del usuario se ha cancelado el pago |
| 9928 | Anulación de autorización en diferido realizada por el SIS (proceso batch) |

| | |
|---|---|
| 9929 | Anulación de autorización en diferido realizada por el comercio |
| 9997 | Se está procesando otra transacción en SIS con la misma tarjeta |
| 9998 | Operación en proceso de solicitud de datos de tarjeta |
| 9999 | Operación que ha sido redirigida al emisor a autenticar |

Estos códigos de respuesta, además de en la propia respuesta del Web Service, se muestran en el campo "Código de respuesta" de la consulta de operaciones, siempre y cuando la operación no está autorizada, tal y como se muestra en la siguiente imagen:



Página 1 de 3

| Sesión / Fecha Totales | Fecha Hora | Tipo Operación Num. Pedido | Resultado NºAutorización o Cod.Respuesta | Importe | Neto Lote/Cajón |
|---|---|---|---|---|---|
| 01-10-15 | 01-10-2015 16:50:16 | Autorización Tradicional 151001165015 | Sin Finalizar 9997 | | |
| 01-10-15 | 01-10-2015 16:50:23 | Autorización Tradicional 151001165022 | Autorizada 581956 | 1,00 EUR | 2 / |

## 8.5 Web Service de petición de pago - WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://webservice.sis.sermepa.es"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://webservice.sis.sermepa.es"
xmlns:intf="http://webservice.sis.sermepa.es"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <wsdl:types>
  <schema elementFormDefault="qualified"
targetNamespace="http://webservice.sis.sermepa.es"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:apachesoap="http://xml.apache.org/xml-
soap" xmlns:impl="http://webservice.sis.sermepa.es"
xmlns:intf="http://webservice.sis.sermepa.es"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
   <element name="trataPeticion">
    <complexType>
     <sequence>
      <element name="datoEntrada" nillable="true" type="xsd:string"/>
     </sequence>
    </complexType>
   </element>
   <element name="trataPeticionResponse">
    <complexType>
     <sequence>
      <element name="trataPeticionReturn" nillable="true" type="xsd:string"/>
     </sequence>
    </complexType>
   </element>
   <element name="consultaDCC">
    <complexType>
     <sequence>
      <element name="datoEntrada" nillable="true" type="xsd:string"/>
     </sequence>
    </complexType>
   </element>
   <element name="consultaDCCResponse">
    <complexType>
     <sequence>
      <element name="consultaDCCReturn" nillable="true" type="xsd:string"/>
     </sequence>
    </complexType>
   </element>
  </schema>
 </wsdl:types>
   <wsdl:message name="consultaDCCRequest">
      <wsdl:part element="intf:consultaDCC" name="parameters"/>
   </wsdl:message>
   <wsdl:message name="trataPeticionResponse">
      <wsdl:part element="intf:trataPeticionResponse" name="parameters"/>
   </wsdl:message>
   <wsdl:message name="trataPeticionRequest">
      <wsdl:part element="intf:trataPeticion" name="parameters"/>
   </wsdl:message>
   <wsdl:message name="consultaDCCResponse">
      <wsdl:part element="intf:consultaDCCResponse" name="parameters"/>
   </wsdl:message>
   <wsdl:portType name="SerClsWSEntrada">
      <wsdl:operation name="trataPeticion">
         <wsdl:input message="intf:trataPeticionRequest" name="trataPeticionRequest"/>
         <wsdl:output message="intf:trataPeticionResponse"
name="trataPeticionResponse"/>
```

```
      </wsdl:operation>
      <wsdl:operation name="consultaDCC">
         <wsdl:input message="intf:consultaDCCRequest" name="consultaDCCRequest"/>
         <wsdl:output message="intf:consultaDCCResponse" name="consultaDCCResponse"/>
      </wsdl:operation>
   </wsdl:portType>
   <wsdl:binding name="SerClsWSEntradaSoapBinding" type="intf:SerClsWSEntrada">
      <wsdlsoap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="trataPeticion">
         <wsdlsoap:operation soapAction=""/>
         <wsdl:input name="trataPeticionRequest">
            <wsdlsoap:body use="literal"/>
         </wsdl:input>
         <wsdl:output name="trataPeticionResponse">
            <wsdlsoap:body use="literal"/>
         </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="consultaDCC">
         <wsdlsoap:operation soapAction=""/>
         <wsdl:input name="consultaDCCRequest">
            <wsdlsoap:body use="literal"/>
         </wsdl:input>
         <wsdl:output name="consultaDCCResponse">
            <wsdlsoap:body use="literal"/>
         </wsdl:output>
      </wsdl:operation>
   </wsdl:binding>
   <wsdl:service name="SerClsWSEntradaService">
      <wsdl:port binding="intf:SerClsWSEntradaSoapBinding" name="SerClsWSEntrada">
         <wsdlsoap:address
location="https://sis.redsys.es/sis/services/SerClsWSEntrada"/>
      </wsdl:port>
   </wsdl:service>
</wsdl:definitions>
```