TPV-Virtual

Manual de Integración - REST

Versión: 2.1
Fecha: 12/04/2019
Referencia: RS.TE.CEL.MAN.0037

Redsys, Servicios de Procesamiento, S.L. – c/ Francisco Sancha, 12 – 28034 Madrid (España)

www.redsys.es

Autorizaciones y control de versión

AUTOR: Redsys	VALIDADO POR: Comercio Electrónico	APROBADO POR: Redsys
Empresa: Redsys	Empresa: Redsys	Empresa: Redsys
Firma:	Firma:	Firma:
Fecha: 27/03/2019	Fecha: 27/03/2019	Fecha: 27/03/2019

Versión	**Fecha**	**Afecta**	**Breve descripción del cambio**
1.0	01/10/2018	TODO	Versión Inicial
1.1	17/12/2018	Anexo 7.7	3DSecure 2.0
1.1	25/01/2019	Anexo 7.9	Integración PSP
1.2	25/01/2019	Apartado 7.2 Peticiones de Confirmación/ Devolución	Añadido soporte para anulación de autorización, al enviar tipo 9 en TRANSACTIONTYPE
1.3	06.02.2019	Varios puntos	Se incluye código de respuesta en operaciones que requieren challenge: *Ds_response=9568* En los datos 3DSecure se indica siempre la versión de protocolo en el campo *protocolVersion* Se incluyen esquemas de secuencias para aclarar el flujo. Se añade el punto 7.8 Integración PSP's
1.4	11/02/2019	Varios puntos	Se incluyen todos los parámetros 3DSecure 1.0.2 y un ejemplo.
1.5	18/02/2019	Varios puntos	MPI Externo Mover el punto 7.7 al punto 5

			Diferentes puntualizaciones en el apartado 5 de Autenticación 3DSecure Incluido apartado de timeout
2.0	27/03/2019	Varios puntos	Añadida la información sobre EMV3DS y PSD2. Añadida la información de transacciones con DCC. Añadida la información de Autenticaciones con DCC. Modificaciones del apartado Integración para PSP Modificaciones del apartado ¿Operas por PUCE? Modificaciones del apartado MPI Externo EMV3DS Añadida referencia a los nuevos documentos TPV-Virtual GuiaErroresSIS.xlsx y TPV-Virtual Parámetros Entrada-Salida.xlsx
2.1	12/04/2019	Puntos 5, 9, 10, 13	Nota importante sobre integración EMV3DS Añadidas tarjetas de pruebas para EMV3DS Puntualización para operaciones PUCE La hoja de cálculo de los errores SIS, se modifica para incluirla en la hoja de cálculo Parámetros de entrada-salida

ÍNDICE

1. Introducción

1.1 Objetivo

Este documento recoge los aspectos técnicos necesarios para que un comercio realice la integración con el TPV Virtual utilizando un interfaz REST.

Esta forma de conexión permite a los comercios tener integrado el TPV Virtual dentro de su propia aplicación Web. Además, este modo de conexión da cabida a la utilización de mecanismos de autenticación como el 3D Secure, donde el banco de la tarjeta solicita directamente al titular un dato secreto que permite dotar de más seguridad a las compras.

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

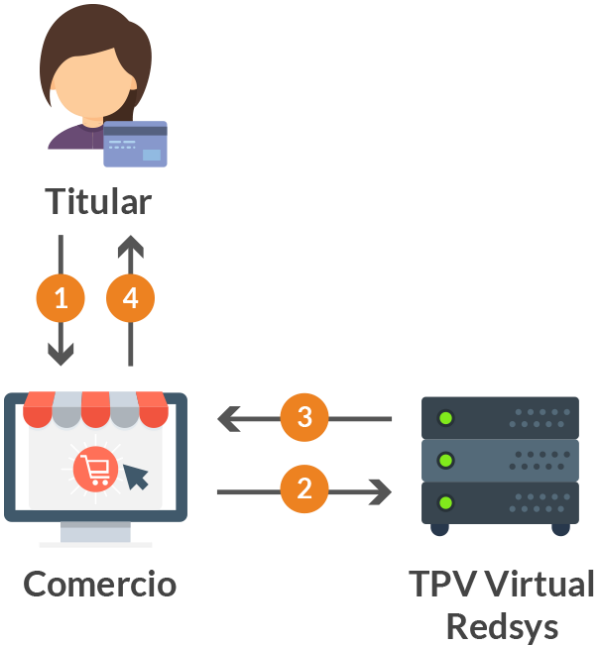
- SIS. Servidor Integrado de Redsys (Servidor del TPV Virtual).
- SCA. Strong Customer Authentication. Autenticación Fuerte del titular.
- Frictionless. Autenticación sin intervención del titular
- Challenge. Autenticación fuerte del titular (mediante OTP, contraseña estática, biometría, etc).
- PSD2. Payment Service Providers. Regulación europea en los servicios de pagos digitales.
- 3DSecure: Sistema de seguridad para los pagos online. En adelante EMV3DS
- EMV3DS: Siglas para identificar 3DSecure en la nueva versión del TPV-Virtual

1.3 Referencias

- Documentación de Integración con el SIS
- TPV-Virtual Guía SIS.
- TPV-Virtual GuiaErroresSIS.xlsx
- TPV-Virtual Parámetros Entrada-Salida.xlsx

2. Descripción general del flujo

El siguiente esquema presenta el flujo general de una operación realizada a través de la entrada REST del TPV Virtual.



1. El titular selecciona los productos que desea comprar e introduce los datos de tarjeta en un formulario mostrado por el comercio.
2. El comercio envía los datos del pago al TPV virtual.
3. Una vez realizado el pago, el TPV virtual informa del resultado de la operación.
4. El comercio devuelve la información del resultado del pago al titular.

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 en una petición REST. En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (En la sección de anexos se incluye la lista de parámetros que se pueden enviar en una solicitud de pago).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior.

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/trataPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/trataPeticionREST	Real

2.2 Recepción del resultado

Una vez gestionada la transacción, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

NOTA: El TPV Virtual envía la notificación on-line a la URL informada por el comercio en el parámetro Ds_Merchant_MerchantURL.

3. Envío de petición

El comercio deberá montar los parámetros de la petición de pago que debe hacer llegar al TPV Virtual a través de una petición Rest. Para facilitar la integración del comercio, a continuación, se explica de forma detallada los pasos a seguir para montar el formulario de petición de pago.

3.1 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 parámetro **Ds_SignatureVersion**, tal y como se puede observar en el ejemplo de formulario mostrado al inicio del apartado 3.

3.2 Montar la cadena de datos de la petición

Se debe montar una cadena con todos los datos de la petición en formato JSON. El nombre de cada parámetro debe indicarse en mayúsculas o con estructura "CamelCase" (Por ejemplo: DS_MERCHANT_AMOUNT o Ds_Merchant_Amount).

Los comercios que utilicen operativas especiales como el "Pago por referencia" (Pago 1-Clic), deberán incluir los parámetros específicos de su operativa como parte del objeto JSON.

La lista de parámetros que se pueden incluir en la petición se describe en el Anexo 11.1.

A continuación, se muestran un ejemplo del objeto JSON de una petición:

*{"DS_MERCHANT_AMOUNT":"145","DS_MERCHANT_ORDER":"1446068581","DS_MERCHANT_MERCHANTCODE":"999008881","DS_MERCHANT_CURRENCY":"978","DS_MERCHANT_TRANSACTIONTYPE":"0","DS_MERCHANT_TERMINAL":"1","DS_MERCHANT_MERCHANTURL":"http:\/\/www.prueba.com\/urlNotificacion.php","DS_MERCHANT_PAN":"454881********04","DS_MERCHANT_EXPIRYDATE":"1512","DS_MERCHANT_CVV2":"123"}*

Una vez montada la cadena JSON con todos los campos, es necesario codificarla en BASE64 sin retornos de carro para asegurarnos de que se mantiene constante y no es alterada en el proceso de envío.

A continuación, se muestra el objeto JSON codificado en BASE64:

eyJEU19NRVJDSEFOVF9BTU9VTlQiOiIxNDUiLCJEU19NRVJDSEFOVF9PUkRFUiI6IjE0NDYwNjg1ODEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVENPREUiOiI5OTkwMDg4ODEiLCJEU19NRVJDSEFOVF9DVVJSRU5DWSI6Ijk3OCIsIkRTX01FUkNIQU5UX1RSQU5TQUNUSU9OVFlQRSI6IjAiLCJEU19NRVJDSEFOVF9URVJNSU5BTCI6IjEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVFVSTCI6Imh0dHA6XC9cL3d3dy5wcnVlYmEuY29tXC91cmxOb3RpZmljYWNpb24ucGhwIiwiRFNfTUVSQ0hBTlRfUEFOIjoiNDU0ODgxMjA0OTQwMDAwNCIsIkRTX01FUkNIQU5UX0VYUElSWURBVEUiOiIxNTEyIiwiRFNfTUVSQ0hBTlRfQ1ZWMiI6IjEyMyJ9

La cadena resultante de la codificación en BASE64 será el valor del parámetro **Ds_MerchantParameters**.

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

3.3 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 la cadena de datos a firmar 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 valor del parámetro **Ds_MerchantParameters** y la clave obtenida en el paso anterior.

3. El resultado obtenido se codifica en BASE 64, y el resultado de la codificación será el valor del parámetro 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 la entrada REST 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 los campos del formulario de pago. 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_PHP_4.0.2/apiRedsys.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 RedsysAPI;
```

3. Calcular el parámetro Ds_MerchantParameters. Para llevar a cabo el cálculo de este parámetro, inicialmente se deben añadir todos los parámetros de la petición de pago que se desea enviar, tal y como se muestra a continuación:

```
$miObj->setParameter("DS_MERCHANT_AMOUNT",$amount);
$miObj->setParameter("DS_MERCHANT_ORDER",$id);
$miObj->setParameter("DS_MERCHANT_MERCHANTCODE",$fuc);
$miObj->setParameter("DS_MERCHANT_CURRENCY",$moneda);
$miObj->setParameter("DS_MERCHANT_TRANSACTIONTYPE",$trans);
$miObj->setParameter("DS_MERCHANT_TERMINAL",$terminal);
$miObj->setParameter("DS_MERCHANT_MERCHANTURL",$url);
```

Por último, para calcular el parámetro Ds_MerchantParameters, se debe llamar a la función de la librería "createMerchantParameters()", tal y como se muestra a continuación:

```
$params = $miObj->createMerchantParameters();
```

4. Calcular el parámetro Ds_Signature. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería "createMerchantSignature()" con la clave obtenida del módulo de administración, tal y como se muestra a continuación:

```
$claveModuloAdmin = 'Mk9m98IfEblmPfrpsawt7BmxObt98Jev';
$signature = $miObj->createMerchantSignature($claveModuloAdmin);
```

5. Una vez obtenidos los valores de los parámetros Ds_MerchantParameters y Ds_Signature, se debe rellenar la petición REST con dichos valores y el parámetro Ds_SignatureVersion.

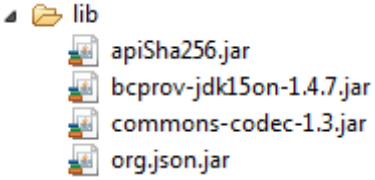
3.5.2 Librería JAVA

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.ApiMacSha256"%>
```

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:

```
ApiMacSha256 apiMacSha256 = new ApiMacSha256();
```

3. Calcular el parámetro Ds_MerchantParameters. Para llevar a cabo el cálculo de este parámetro, inicialmente se deben añadir todos los parámetros de la petición de pago que se desea enviar, tal y como se muestra a continuación:

```
apiMacSha256.setParameter("DS_MERCHANT_AMOUNT", amount);
apiMacSha256.setParameter("DS_MERCHANT_ORDER", id);
apiMacSha256.setParameter("DS_MERCHANT_MERCHANTCODE", fuc);
apiMacSha256.setParameter("DS_MERCHANT_CURRENCY", moneda);
apiMacSha256.setParameter("DS_MERCHANT_TRANSACTIONTYPE", trans);
apiMacSha256.setParameter("DS_MERCHANT_TERMINAL", terminal);
apiMacSha256.setParameter("DS_MERCHANT_MERCHANTURL", url);
```

Por último se debe llamar a la función de la librería “createMerchantParameters()”, tal y como se muestra a continuación:

```
String params = apiMacSha256.createMerchantParameters();
```

4. Calcular el parámetro Ds_Signature. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería “createMerchantSignature()” con la clave obtenida del módulo de administración, tal y como se muestra a continuación:

```
String claveModuloAdmin = "Mk9m98IfEblmPfrpsawt7BmxObt98Jev";
String signature = apiMacSha256.createMerchantSignature(claveModuloAdmin);
```

5. Una vez obtenidos los valores de los parámetros Ds_MerchantParameters y Ds_Signature, se debe rellenar la petición REST con dichos valores y el parámetro Ds_SignatureVersion

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 RedsysAPI y Newronsoft.Json en su proyecto.
2. Calcular el parámetro **Ds_MerchantParameters**. Para llevar a cabo el cálculo de este parámetro, inicialmente se deben añadir todos los parámetros de la petición de pago que se desea enviar, tal y como se muestra a continuación:

```
// New instance of RedysAPI
RedsysAPI r = new RedsysAPI();

// Fill Ds_MerchantParameters parameters
r.SetParameter("DS_MERCHANT_AMOUNT", amount);
r.SetParameter("DS_MERCHANT_ORDER", id);
r.SetParameter("DS_MERCHANT_MERCHANTCODE", fuc);
r.SetParameter("DS_MERCHANT_CURRENCY", currency);
r.SetParameter("DS_MERCHANT_TRANSACTIONTYPE", trans);
r.SetParameter("DS_MERCHANT_TERMINAL", terminal);
r.SetParameter("DS_MERCHANT_MERCHANTURL", url);
```

Por último se debe llamar a la función de la librería “createMerchantParameters()”, tal y como se muestra a continuación:

```
string parms = r.createMerchantParameters();
Ds_MerchantParameters.Value = parms;
```

3. Calcular el parámetro **Ds_Signature**. Para llevar a cabo el cálculo de este parámetro, se debe llamar a la función de la librería “createMerchantSignature()” con la clave obtenida del módulo de administración, tal y como se muestra a continuación:

```
string sig = r.createMerchantSignature(kc);
Ds_Signature.Value = sig;
```

4. Una vez obtenidos los valores de los parámetros Ds_MerchantParameters y Ds_Signature, se debe rellenar la petición REST con dichos valores y el parámetro Ds_SignatureVersion.

4. Respuesta de petición

Una vez recibida la respuesta de la petición al TPV Virtual, el comercio debe capturar y validar los parámetros de retorno para conocer el resultado de la operación

Se posibilita la utilización de las librerías de ayuda proporcionadas por Redsys para la captura y validación de los parámetros del retorno:

4.1 Utilización de librerías de ayuda

En los apartados anteriores se ha descrito la forma de acceso al SIS utilizando conexión REST. En este apartado se explica cómo se utilizan las librerías disponibles PHP, JAVA y .NET para facilitar los desarrollos para la recepción de los parámetros en la respuesta del servicio REST. El uso de las librerías suministradas por Redsys es opcional, si bien simplifican los desarrollos a realizar por el comercio.

4.1.1 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_PHP_4.0.2/apiRedsys.php';
   ```

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

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

   ```
   $miObj = new RedsysAPI;
   ```

2. Capturar los parámetros de la respuesta:

   ```
   $version = $_GET["Ds_SignatureVersion"];
   $params = $_GET["Ds_MerchantParameters"];
   $signatureRecibida = $_GET["Ds_Signature"];
   ```

3. Decodificar el parámetro **Ds_MerchantParameters**. Para llevar a cabo la decodificación de este parámetro, se debe llamar a la función de la librería “decodeMerchantParameters()”, tal y como se muestra a continuación:

   ```
   $decodec = $miObj->decodeMerchantParameters($params);
   ```

4. Una vez se ha realizado la llamada a la función “decodeMerchantParameters()”, se puede obtener el valor de cualquier parámetro que sea susceptible de incluirse en la respuesta (Anexo **¡Error! No se encuentra el origen de la referencia.**). Para llevar a cabo la obtención d el valor de un parámetro se debe llamar a la función “getParameter()” de la librería con el nombre de parámetro, tal y como se muestra a continuación para obtener el código de respuesta:

```
$codigoRespuesta = $miObj->getParameter("Ds_Response");
```

NOTA IMPORTANTE: Es importante llevar a cabo la validación de todos los parámetros que se envían en la comunicación.

5. Validar el parámetro **Ds_Signature**. Para llevar a cabo la validación de este parámetro se debe calcular la firma y compararla con el parámetro **Ds_Signature** capturado. Para ello se debe llamar a la función de la librería “createMerchantSignatureNotif()” con la clave obtenida del módulo de administración y el parámetro **Ds_MerchantParameters** capturado, tal y como se muestra a continuación:

```
$claveModuloAdmin = 'Mk9m98IfEblmPfrpsawt7BmxObt98Jev';
$signatureCalculada = $miObj->createMerchantSignatureNotif($claveModuloAdmin,
                                                           $params);
```

6. Una vez hecho esto, ya se puede validar si el valor de la firma enviada coincide con el valor de la firma calculada, tal y como se muestra a continuación:

```
if ($signatureCalculada === $signatureRecibida){
    echo "FIRMA OK. Realizar tareas en el servidor";
} else {
    echo "FIRMA KO. Error, firma inválida";
}
```

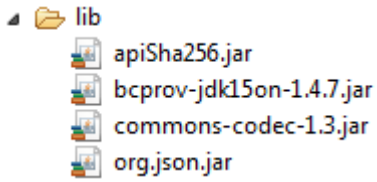
4.1.2 Librería JAVA

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.ApiMacSha256"%>
```

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:

```
ApiMacSha256 apiMacSha256 = new ApiMacSha256();
```

3. Capturar los parámetros del retorno de la petición:

```
String version = request.getParameter("Ds_SignatureVersion");
String params = request.getParameter("Ds_MerchantParameters");
String signatureRecibida = request.getParameter("Ds_Signature");
```

Decodificar el parámetro **Ds_MerchantParameters**. Para llevar a cabo la decodificación de este parámetro, se debe llamar a la función de la librería “decodeMerchantParameters()”, tal y como se muestra a continuación:

```
String decodec = apiMacSha256.decodeMerchantParameters(params);
```

Una vez se ha realizado la llamada a la función “decodeMerchantParameters()”, se puede obtener el valor de cualquier parámetro que sea susceptible de incluirse en la respuesta (Anexo **¡Error! No se encuentra el origen de la referencia.**). Para llevar a cabo la obtención d el valor de un parámetro se debe llamar a la función “getParameter()” de la librería con el nombre de parámetro, tal y como se muestra a continuación para obtener el código de respuesta:

```
String codigoRespuesta = apiMacSha256.getParameter("Ds_Response");
```

NOTA IMPORTANTE: Es importante llevar a cabo la validación de todos los parámetros que se envían en la comunicación.

4. Validar el parámetro **Ds_Signature**. Para llevar a cabo la validación de este parámetro se debe calcular la firma y compararla con el parámetro **Ds_Signature** capturado. Para ello se debe llamar a la función de la librería “createMerchantSignatureNotif()” con la clave obtenida del módulo de administración y el parámetro **Ds_MerchantParameters** capturado, tal y como se muestra a continuación:

```
String claveModuloAdmin = "Mk9m98IfEblmPfrpsawt7BmxObt98Jev";
String signatureCalculada = apiMacSha256.createMerchantSignatureNotif(claveModuloAdmin,
                                                                       params);
```

Una vez hecho esto, ya se puede validar si el valor de la firma enviada coincide con el valor de la firma calculada, tal y como se muestra a continuación:

```
if (signatureCalculada.equals(signatureRecibida)) {
    System.out.println("FIRMA OK. Realizar tareas en el servidor");
} else {
    System.out.println("FIRMA KO. Error, firma inválida");
}
```

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

```
RedsysAPI r = new RedsysAPI();
```

3. Capturar los parámetros del retorno:

```
string version = Request.QueryString["Ds_SignatureVersion"];
string parms = Request.QueryString["Ds_MerchantParameters"];
string signatureRecibida = Request.QueryString["Ds_Signature"];
```

NOTA IMPORTANTE: Es importante llevar a cabo la validación de todos los parámetros que se envían en la comunicación.

4. Validar el parámetro **Ds_Signature**. Para llevar a cabo la validación de este parámetro se debe calcular la firma y compararla con el parámetro **Ds_Signature** capturado. Para ello se debe llamar a la función de la librería "createMerchantSignatureNotif()" con la clave obtenida del módulo de administración y el parámetro **Ds_MerchantParameters** capturado, tal y como se muestra a continuación:

```
var kc = "sq7HjrUOBfKmC576ILgskD5srU870gJ7";

string signatureCalculada =  r.createMerchantSignatureNotif(kc, parms);
```

Una vez hecho esto, ya se puede validar si el valor de la firma enviada coincide con el valor de la firma calculada, tal y como se muestra a continuación:

```
if (signatureRecibida == signatureCalculada)
{
    result.InnerHtml = "FIRMA OK. Realizar tareas en el servidor";
}
else
{
    result.InnerHtml = "FIRMA KO. Error, firma invalida";
}
```

5. Transacciones con Autenticación EMV3DS

5.1 Pasos para realizar una transacción con autenticación EMV3DS

Los pagos con autenticación EMV3DS en la conexión Rest sigue los siguientes pasos:

- **Paso 1: Iniciar petición**

El comercio deberá hacer una petición al TPV Virtual para saber si la tarjeta está inscrita en EMV3DS v2 y poder iniciar el proceso de autenticación con un protocolo u otro.

- **Paso 2: 3DSMethod (Si procede)**

El comercio ejecuta el 3DSMethod para que el emisor capture la información del dispositivo.

- **Paso 3: Solicitud de autorización**

El comercio enviará la solicitud de autorización de la operación incluyendo el resultado del 3DSMethod y otros datos adicionales del protocolo EMV3DS.

El TPV Virtual iniciará la autenticación, donde se podrá obtener como resultado:

a. Autenticación OK (Frictionless): la operación ha sido autenticada y el TPV Virtual continuará el proceso de autorización.

b. Challenge Required: La entidad emisora requiere verificar la autenticidad del cliente

c. Otro resultado: autenticación no disponible, autenticación rechazada, error en la autenticación, etc.

El TPV Virtual decidirá según el caso autorizar o rechazar la operación.

- **Paso 4: Autenticación (Si procede)**

La entidad emisora requiere que el titular verifique su autenticidad (mediante OTP, contraseña estática, biometría, etc).

- **Paso 5: Confirmación de autorización**

El comercio enviará la autorización con el resultado del challenge al TPV Virtual para finalizar el proceso de autorización.

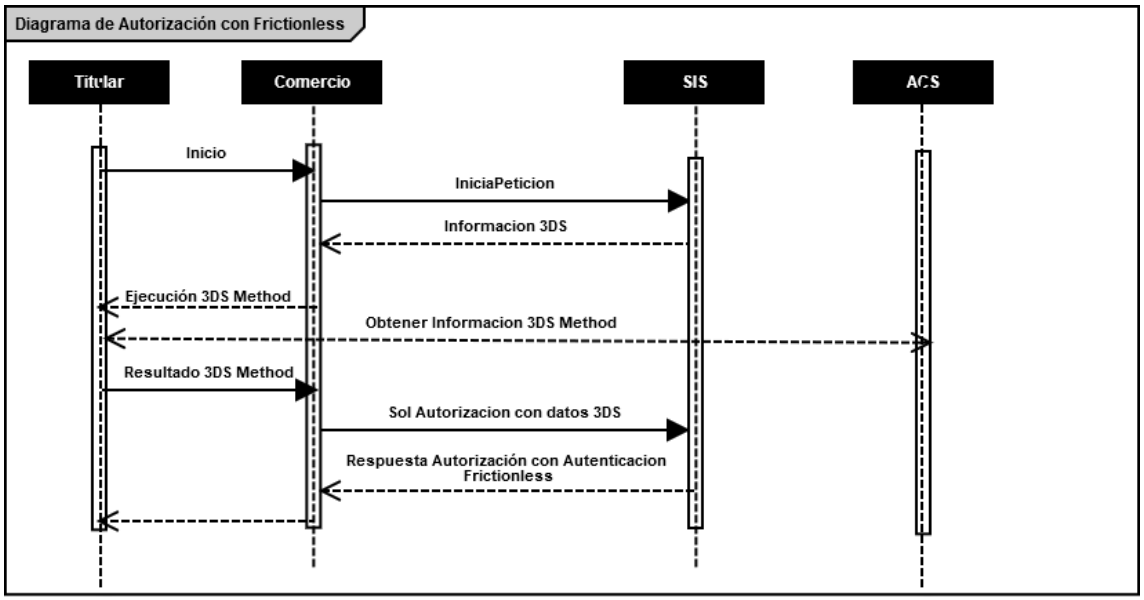
Recomendamos que en el paso 3 el comercio proporcione toda la información adicional para aumentar la probabilidad de flujo frictionless y una mayor tasa de autorización.

El tiempo máximo desde el inicio de la petición y la solicitud de autorización es de 1 hora. Pasado este tiempo la petición se da como perdida y se deberá volver a realizar el flujo desde el principio.

Nota importante: Se deberá integrar todo este apartado con los tres flujos que se describen a continuación. Según el resultado de la primera petición (inicia petición) los flujos serán de la versión 1 o la 2, puede que la tarjeta no esté inscrita en EMV3DS V2 y se debe iniciar la autenticación por el flujo de v1.

5.1.1 Ejemplo del flujo de una Autorización con autenticación EMV3DS Frictionless

El siguiente esquema presenta el flujo general de una operación con autenticación frictionless realizada a través del TPV Virtual.



1. El titular selecciona los productos que desea comprar e introduce los datos de tarjeta en un formulario mostrado por el comercio.
2. El comercio realiza un inicia petición enviando los datos al TPV Virtual.
3. El TPV Virtual comprueba la configuración de la tarjeta, y en la respuesta informará si la tarjeta soporta autenticación EMV3DS y la versión de protocolo EMV3DS que se aplica.

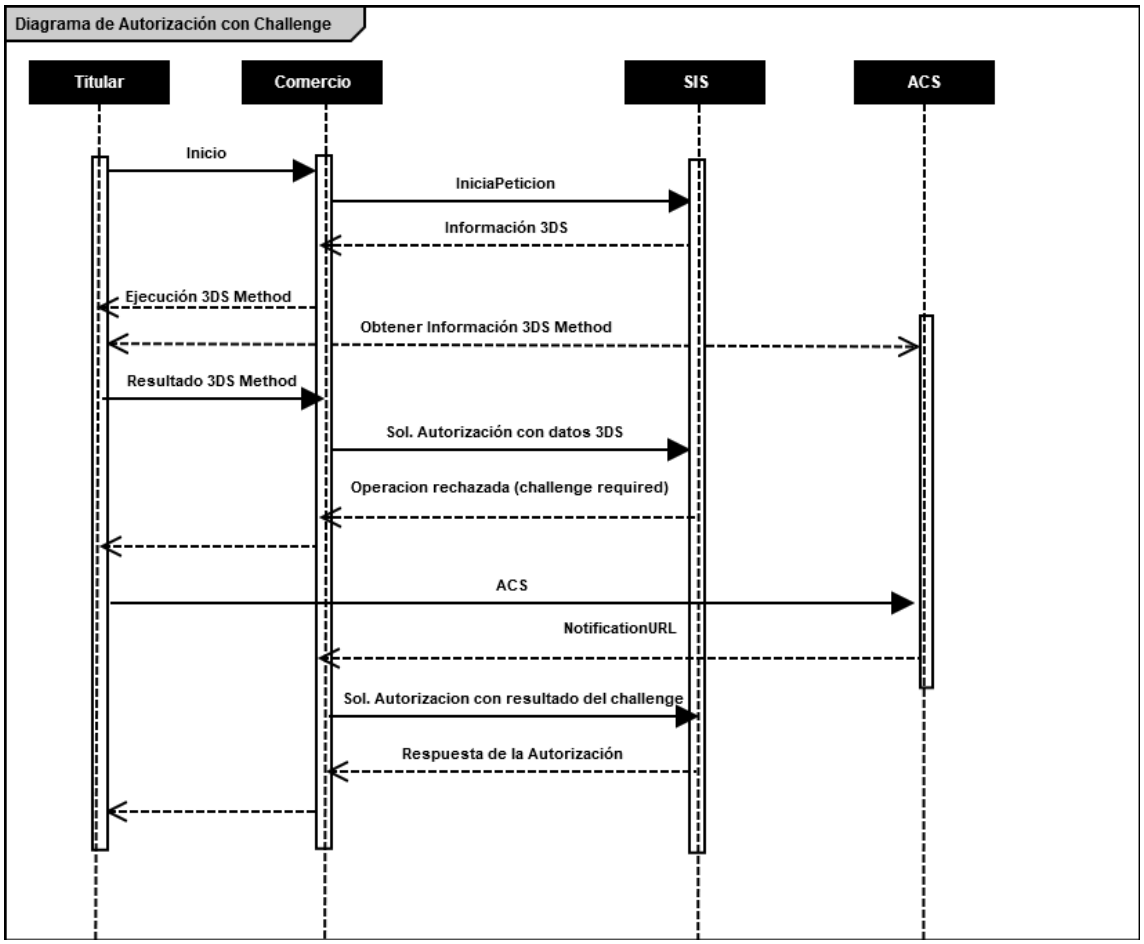
    3.1 Si la tarjeta lo requiere ejecutar el 3DSMethod: se inicia conexión desde el browser con el ACS y este devuelve el resultado de la ejecución al comercio
4. El comercio envía la solicitud de autorización con tarjeta que soporta EMV3DS v2. Además de los datos de pago es necesario enviar el resultado del 3DSMethod y los datos adicionales para la autenticación.

    El TPV Virtual inicia la autenticación, y el emisor, en base a los datos recibidos, autentica la operación sin necesidad de intervención del titular. A continuación, el TPV Virtual procesará la autorización
5. Una vez realizado el pago, el TPV virtual informa del resultado de la operación.
6. El comercio devuelve la información del resultado del pago al titular.

NOTA: en los diagramas solo se tienen en cuenta los flujos del comercio y los actores con lo que interviene en contacto directo.

5.1.2 Ejemplo del flujo de una Autorización con autenticación EMV3DS Challenge

El siguiente esquema presenta el flujo general de una operación con autenticación por challenge realizada a través del TPV Virtual.



1. El titular selecciona los productos que desea comprar e introduce los datos de tarjeta en un formulario mostrado por el comercio.
2. El comercio realiza un inicia petición enviando los datos al TPV Virtual.
3. El TPV Virtual comprueba la configuración de la tarjeta, y en la respuesta informará si la tarjeta soporta autenticación EMV3DS y la versión de protocolo EMV3DS que se aplica.

   3.1 Si la tarjeta lo requiere ejecutar el 3DSMethod: se inicia conexión desde el browser con el ACS y este devuelve el resultado de la ejecución al comercio
4. El comercio envía la solicitud de autorización con tarjeta que soporta EMV3DS v2. Además de los datos de pago es necesario enviar el resultado del 3DSMethod y los datos adicionales para la autenticación.

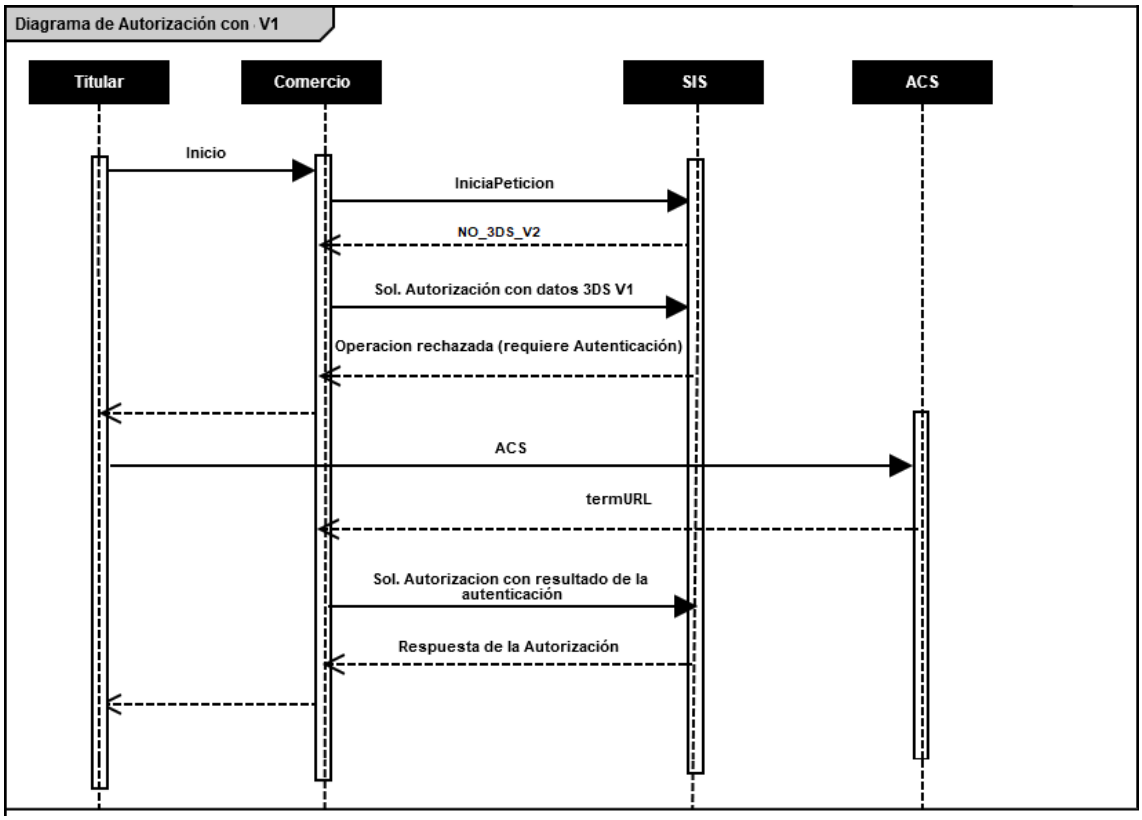
   El TPV Virtual inicia la autenticación, y el emisor en base a los datos recibidos decide que el titular debe verificar su autenticidad (challenge)

5. El TPV Virtual devuelve la información para que el titular pueda realizar el challenge con su banco emisor. Según el protocolo con el que se deba autenticar estas peticiones quedarán marcadas de la siguiente forma. Versión 2.1.0 (8210)
6. El comercio redirige al titular vía browser para que conecte con su emisor
7. El titular completa el challenge
8. La entidad emisora devuelve el resultado del challenge a la url indicada por el comercio
9. El comercio envía el resultado del challenge al TPV Virtual para finalizar el proceso de autorización
10. Una vez realizado el pago, el TPV virtual responde con el resultado de la operación.
11. El comercio responde con la información del resultado del pago al titular

NOTA: en los diagramas solo se tienen en cuenta los flujos del comercio y los actores con lo que interviene en contacto directo.

5.1.3 Ejemplo del flujo de una Autorización con autenticación EMV3DS v1

El siguiente esquema presenta el flujo general de una operación con autenticación EMV3DS v1, donde se ha determinado que es necesario realizar la autenticación.



1. El titular selecciona los productos que desea comprar e introduce los datos de tarjeta en un formulario mostrado por el comercio.
2. El comercio realiza un inicia petición enviando los datos al TPV Virtual.
3. El TPV Virtual comprueba la configuración de la tarjeta, y en la respuesta informará si la tarjeta soporta autenticación EMV3DS y la versión de protocolo EMV3DS que se aplica.

   3.2 Si la tarjeta no permite EMV3DS v2 el comercio puede solicitar la autenticación con los datos de v1. Se recibirá un valor de protocolo NO_3DS_V2
4. El comercio envía la solicitud de autorización al TPV Virtual indicando que está preparado para EMV3DS v1.
5. El TPV Virtual comprueba la configuración del comercio y la tarjeta, y en la respuesta informará que la transacción requiere autenticación y la información para que el titular pueda realizar la autenticación con su banco emisor. Estas peticiones quedarán marcadas de la siguiente forma Versión 1.0.2 (8102).
6. El comercio redirige al titular vía browser para que conecte con su emisor.
7. El titular completa la autenticación.

8. La entidad emisora devuelve el resultado de la autenticación a la URL indicada facilitada por el comercio.
9. El comercio envía el resultado de la autenticación al TPV Virtual para finalizar el proceso de autorización.
10. Una vez realizado el pago, el TPV virtual responde con el resultado de la operación.
11. El comercio responde con la información del resultado del pago al titular

NOTA: en los diagramas solo se tienen en cuenta los flujos del comercio y los actores con lo que interviene en contacto directo.

5.2 Peticiones para realizar una transacción con autenticación EMV3DS v2

5.2.1 Iniciar Petición

Esta petición permite obtener el tipo de autenticación 3D Secure que se puede realizar, además de la URL del 3DSMethod en caso de que exista.

El inicia petición se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/iniciaPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/iniciaPeticionREST	Real

Una vez gestionada la petición, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de**

validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar un inicia petición al Servicio REST:


```
{
  "DS_MERCHANT_ORDER":"1552571678",
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"999",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EMV3DS": {"threeDSInfo":"CardData"}
}
```


Como respuesta se obtendrá lo siguiente:


```
{
  "Ds_Order":"1552571678",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_TransactionType":"0",
  "Ds_EMV3DS": {
    "protocolVersion":"2.1.0",
    "threeDSServerTransID":"8de84430-3336-4ff4-b18d-f073b546ccea",
    "threeDSInfo":"CardConfiguration",
    "threeDSMethodURL":"https://sis.redsys.es/sis-simulador-web/threeDsMethod.jsp"
  }
}
```


5.2.2 Ejecución del 3DSMethod

El 3DSMethod es un proceso que permite a la entidad emisora capturar la información del dispositivo que está utilizando el titular. Esta información, junto con los datos EMV3DS que son enviados en la autorización, será utilizada por la entidad para hacer una evaluación del riesgo de la transacción. En base a esto el emisor puede determinar que la transacción es confiable y por lo tanto no requerir la intervención del titular para verificar su autenticidad (frictionless).

La captura de datos del dispositivo se realiza mediante un iframe oculto en el navegador del cliente, que establecerá conexión directamente con la entidad emisora de forma transparente para el usuario. El comercio recibirá una notificación cuanto haya terminado la captura de información. En el siguiente paso, al solicitar la autorización al TPV Virtual se enviará el parámetro threeDSCompInd indicando la ejecución del 3DSMethod

Pasos para la ejecución del 3DSMethod:

1. En la respuesta recibida con la configuración de la tarjeta (iniciaPeticion) se recibe los datos siguientes para ejecutar el 3DSMethod:
    a. threeDSMethodURL: url del 3DSMethod
    b. threeDSServerTransID: Identificador de transacción EMV3DS

    Si en la respuesta no se recibe threeDSMethodURL el proceso finaliza. En la autorización enviar threeDSCompInd = N
2. Construir el JSON Object con los parámetros:
    a. threeDSServerTransID: valor recibido en la respuesta de consulta de tarjeta
    b. threeDSMethodNotificationURL: url del comercio a la que será notificada la finalización del 3DSMethod desde la entidad
3. Codificar el JSON anterior en Base64url encode
4. Debe incluirse un iframe oculto en el navegador del cliente, y enviar un campo **threeDSMethodData** con el valor del objecto json anterior, en un formulario http post a la url obtenida en la consulta inicial **threeDSMethodURL**
5. La entidad emisora interactúa con el browser para proceder a la captura de información. Al finalizar enviará el campo **threeDSMethodData** en el iframe html del navegador por http post a la url **threeDSMethodNotificationURL** (indicada en el paso 2), y el 3DSMethod termina.
6. Si el 3DSMethod se ha completado en menos de 10 segundos se enviará **threeDSCompInd = Y** en la autorización. Si no se ha completado en 10 segundos debe detener la espera y enviar la autorización con **threeDSCompInd = N**

5.2.3 Petición de autorización con datos EMV3DS

La petición de autorización se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/trataPeticionREST	Pruebas

https://sis.redsys.es/sis/rest/trataPeticionREST	Real

Una vez gestionada la consulta, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autorización con autenticación EMV3DS v2 al Servicio REST:


```
{
  "DS_MERCHANT_ORDER":1552572812,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_EMV3DS":
          {
                  "threeDSInfo":"AuthenticationData",
                  "protocolVersion":"2.1.0",
                  "browserAcceptHeader":"text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8,ap
          plication/json",
                  "browserUserAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36",
                  "browserJavaEnabled":"false",
                  "browserLanguage":"ES-es",
                  "browserColorDepth":"24",
                  "browserScreenHeight":"1250",
                  "browserScreenWidth":"1320",
                  "browserTZ":"52",
                  "threeDSServerTransID":"8de84430-3336-4ff4-b18d-f073b546ccea",
                  "notificationURL":"https://sis-d.redsys.es/sis-simulador-
web/SisRESTCreqCres_3DSecureV2.jsp",
                  "threeDSCompInd":"Y"
          }
}
```


Como respuesta se obtendrá:

- Si se hace un ***Frictionless***, se obtendrá directamente el resultado final de la operación:


```
{
```

```
  "Ds_Amount":"1000",
  "Ds_Currency":"978",
  "Ds_Order":"1552572812",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_Response":"0000",
  "Ds_AuthorisationCode":"694432",
  "Ds_TransactionType":"0",
  "Ds_SecurePayment":"1",
  "Ds_Language":"1",
  "Ds_CardNumber":"454881******0004",
  "Ds_Card_Type":"C",
  "Ds_MerchantData":"",
  "Ds_Card_Country":"724",
  "Ds_Card_Brand":"1"
}
```


- Si se requiere ***Challenge***, se obtendrán los datos para realizar el Challenge:


```
{
  "Ds_Amount":"1000",
  "Ds_Currency":"978",
  "Ds_Order":"1552572812",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_TransactionType":"0",
  "Ds_EMV3DS":{
        "threeDSInfo":"ChallengeRequest",
        "protocolVersion":"2.1.0",
        "acsURL":"https://sis.redsys.es/sis-simulador-web/authenticationRequest.jsp",
        "CReq":"eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6IjhkZTg0NDMwLTMzMzYtNGZmNC1iMThkLWYwNzNiNTQ2Y2NlYSIsImFjc1RyYW5zSUQiOiJkYjVjOTljNC1hMmZkLTQ3ZWUtOTI2Zi1mYTBiMDk0MzUyYTAiLCJtZXNzYWdlVHlwZSI6IkNSZXEiLCJtZXNzYWdlVmVyc2lvbiI6IjIuMS4wIiwiY2hhbGxlbmdlV2luZG93U2l6ZSI6IjA1In0"
  }
}
```


5.2.4 Ejecución del Challenge

El comercio deberá montar un formulario que envíe un POST a la URL del parámetro *acsURL* obtenido en la respuesta de la petición de autorización anterior. Dicho formulario envía un único parámetro *CReq*, cuyo valor se obtiene del parámetro *CReq* obtenido en la respuesta de la petición de autorización anterior.

Este formulario enviará al titular a la página de autenticanción de su banco, donde la entidad emisora requerirá que el titular verifique su autenticidad (mediante OTP, contraseña estática, biometría, etc).

Como resultado de esta verificación, la entidad Emisora hará un POST a la URL indicada en el parámetro *notificationURL* de la petición de autorización anterior. Dicho formulario enviará un único parámetro *Cres*, que contiene el resultado de la autenticación y que deberá ser recogido por el comercio para su posterior envío en la petición de confirmación de autorización EMV3DS.

5.2.5 Confirmación de autorización EMV3DS posterior al Challenge

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de confirmación de autorización EMV3DS v2 al Servicio REST:


```
{
  "DS_MERCHANT_ORDER":1552577128,
```

```
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_EMV3DS":{
                        "threeDSInfo":"ChallengeResponse",
                        "protocolVersion":"2.1.0",
                        "CRes":"eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6IjhkZTg0NDMwLTMzMzYtNGZmNC1iMThkLWY
            wNzNiNTQ2Y2NlYSIsImFjc1RyYW5zSUQiOiJkYjVjOTljNC1hMmZkLTQ3ZWUtOTI2Zi1mYTBiMDk0MzUyYTAiL
            CJtZXNzYWdlVHlwZSI6IkNSZXMiLCJtZXNzYWdlVmVyc2lvbiI6IjIuMS4wIiwidHJhbnNTdGF0dXMiOiJZIn0="
            }
 }
```


Como respuesta se obtendrá el resultado final de la operación:


```
{
  "Ds_Amount":"1000",
  "Ds_Currency":"978",
  "Ds_Order":"1552572812",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_Response":"0000",
  "Ds_AuthorisationCode":"694432",
  "Ds_TransactionType":"0",
  "Ds_SecurePayment":"1",
  "Ds_Language":"1",
  "Ds_CardNumber":"454881******0004",
  "Ds_Card_Type":"C",
  "Ds_MerchantData":"",
  "Ds_Card_Country":"724",
  "Ds_Card_Brand":"1"
}
```


5.3 Peticiones para realizar una transacción con autenticación EMV3DS v1

5.3.1 Iniciar Petición

Esta petición permite obtener el tipo de autenticación 3D Secure que se puede realizar.

El inicia petición se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/iniciaPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/iniciaPeticionREST	Real

Una vez gestionada la petición, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar un inicia petición al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":"1552571678",
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"999",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EMV3DS": {"threeDSInfo":"CardData"}
}
```

Como respuesta se obtendrá lo siguiente:

```
{
  "Ds_Order":"1552571678",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_TransactionType":"0",
  "Ds_EMV3DS": {
    "protocolVersion":"NO_3DS_V2"
  }
}
```

5.3.2 Solicitar autorización

Esta petición permitir indicar al comercio que quiere solicitar una transacción pero que está preparado para realizar la autenticación EMV3DS v1 si procede.

La solicitud de autorización se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/trataPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/trataPeticionREST	Real

Una vez gestionada la consulta, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autenticación al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552642885,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_EMV3DS":{
                        "threeDSInfo":"AuthenticationData",
                        "protocolVersion":"1.0.2",
```

```
                "browserAcceptHeader":"text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8,ap
            plication/json",
                "browserUserAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
            (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36"
                }
}
```


Como respuesta se obtendrá lo siguiente:


```
{
  "Ds_Order":"1552642885",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_Currency":"978",
  "Ds_Amount":"1000",
  "Ds_TransactionType":"0",
  "Ds_EMV3DS": {
            "threeDSInfo":"ChallengeRequest",
            "protocolVersion":"1.0.2",
            "acsURL": "https://sis.redsys.es/sis-simulador-web/authenticationRequest.jsp",
            "PAReq":"eJxVUttygjAQ/RWG95KEooKzpkPVjj7QOpZ+QBp2KlYuDVDx77tRqS0zmdmzJ+zlnMBDXxycbzRNX
pUzV3jcdbDUVZaXHzP3LX26C90HCenOIC5eUXcGJSTYNOoDnTybuU1Rq7zPsFGlFmIU+mOfi4j7vrAfn3DOw9HYlbCJt/gl
4dpKUifPBzZAqmn0TpWtBKW/HtfPMhgFYSiAXSEUaNYL+brcJstNnCy381X8nAK7pKFUBcp5RUjnlZOhU5sO35XzZFSXIb
AzD7rqytac5MQPgA0AOnOQu7atp4wdj0fPYNacGk9XBTBLAbvNtuls1FCpPs9kso/7lzQ+Jftln6R09p88WcRHOjNg9gZkqk
U5KOIIPhVi6kfAznlQhZ1BintPcNr0gqC2TeKBsszfDJAHhiw6yWgS0hYDAuzrqkS6QbL+xkDOaEY73Cafr6zGuiXZ/lololEYW
LnPjK2WkzhBJC7lLABm/2VXJ9n1GVD073n8AOa7wW0=",
            "MD":"cd164a6d0b77c96f7ef476121acfa987a0edf602"
  }
}
```


5.3.3 Ejecución de la autenticación

El comercio deberá montar un formulario que envíe un POST a la URL del parámetro *acsURL* obtenido en la respuesta de la petición de autorización anterior. Dicho formulario envía 3 parámetros necesarios para la autenticación:

- *PaReq*, cuyo valor se obtiene del parámetro *PAReq* obtenido en la respuesta de la petición de autorización anterior.
- *MD*, cuyo valor se obtiene del parámetro *MD* obtenido en la respuesta de la petición de autorización anterior.
- *TermUrl*, que identifica la URL a la que entidad Emisora hará un POST con el resultado de autenticación. Dicho formulario enviará un único parámetro *PARes*, que contiene el resultado de la autenticación y que deberá ser recogido por el comercio para su posterior envío en la petición de confirmación de autorización.

5.3.4 Confirmación de autorización EMV3DS posterior al Challenge

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de confirmación de autorización EMV3DS v1 al Servicio REST:


```
{
  "DS_MERCHANT_ORDER":1552642885,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
```

```
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_EMV3DS":{
            "threeDSInfo":"ChallengeResponse",
            "protocolVersion":"1.0.2",
            "PARes":"eJzFWNmSo0iyfecrymoeNVVsWqBNmWPBKlaJVcAbmwBJLALE9vWDlJVZWT3VNn3vw70yyRR4u
 Dvu ESeOu8X2X0N+/dLFdZOVxctX9Dvy9UtchGWUFcnLV8vkvhFf//W6NdM6jhkjDu91/LpV4qbxk/hL …..",
            "MD":"035535127d549298f11d7d2fc1b0d4e9300f93f1"
            }
}
```


Como respuesta se obtendrá el resultado final de la operación:


```
{
  "Ds_Amount":"1000",
  "Ds_Currency":"978",
  "Ds_Order":"1552642885",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_Response":"0000",
  "Ds_AuthorisationCode":"694432",
  "Ds_TransactionType":"0",
  "Ds_SecurePayment":"1",
  "Ds_Language":"1",
  "Ds_CardNumber":"454881******0004",
  "Ds_Card_Type":"C",
  "Ds_MerchantData":"",
  "Ds_Card_Country":"724",
  "Ds_Card_Brand":"1"
}
```

6. Transacciones con DCC

6.1 Pasos para realizar una transacción con DCC

A continuación, se detallarán todas aquellas características adicionales de la operativa DCC en los comercios que utilicen la interfaz REST y cuya entidad les haya activado previamente el servicio.

Los pagos con DCC en la conexión Rest sigue los siguientes pasos:

- **Paso 1: Iniciar petición**

El comercio deberá hacer una consulta al TPV Virtual para saber si la tarjeta ofrece DCC y la información de DCC asociada a la transacción que se ha indicado.

- **Paso 2: Solicitud de autorización**

El comercio enviará la solicitud de autorización de la operación incluyendo la información de DCC obtenida en el paso anterior.

El tiempo máximo desde el inicio de la petición y la solicitud de autorización es de 1 hora. Pasado este tiempo la petición se da como perdida y se deberá volver a realizar el flujo desde el principio.

6.2 Pasos para realizar una transacción con DCC

6.2.1 Iniciar Petición

Esta petición permite obtener los datos de DCC para la petición solicitada.

El inicio de petición se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/iniciaPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/iniciaPeticionREST	Real

Una vez gestionada la consulta, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de inicia peticion al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552580496,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_PAN":"XXXXXXXXXXXXXXXXXX",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_DCC":"Y"
}
```

Como respuesta se obtendrá lo siguiente:

```
{
  "Ds_Order":"1552580496",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_TransactionType":"0",
  "Ds_DCC":{
    "InfoMonedaTarjeta":{
      "monedaDCC":"840",
      "litMonedaDCC":"DOLAR U.S.A.",
      "litMonedaRDCC":"USD",
      "importeDCC":"11.50",
      "cambioDCC":"0.869841",
      "fechaCambioDCC":"2019-01-16",
      "markUp":"0.03"
    },
    "InfoMonedaTitular":{
      "monedaCome":"978",
      "litMonedaCome":"EUR",
      "importeCome":"10.00"
    }
  }
}
```

6.2.2 Petición de autorización con DCC

Esta petición permite indicar al comercio que quiere iniciar una transacción con los datos de DCC obtenidos anteriormente.

La solicitud de autorización se hace a través de una petición REST al TPV Virtual En dicha petición deberá incluir los siguientes parámetros:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (Consultar Parámetros de entrada y salida).
- Ds_Signature: Firma de los datos enviados. Es el resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/trataPeticionREST	Pruebas
https://sis.redsys.es/sis/rest/trataPeticionREST	Real

Una vez gestionada la consulta, el TPV Virtual informará al servidor del comercio el resultado de la misma con la información del resultado incluida en un fichero JSON. En él se incluirán los siguientes campos:

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la respuesta codificada en Base 64 y sin retornos de carro.
- Ds_Signature: Firma de los datos enviados. Resultado del HMAC SHA256 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior. **El comercio es responsable de validar el HMAC enviado por el TPV Virtual para asegurarse de la validez de la respuesta. Esta validación es necesaria para garantizar que los datos no han sido manipulados y que el origen es realmente el TPV Virtual.**

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autorización con DCC al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552581014,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":"XXXXXXXXXXXXXXXXXX",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_DCC":{
                     "monedaDCC":"840",
                     "importeDCC":"11.50"
                     }
```

```
}
```

Como respuesta se obtendrá el resultado final de la operación:


```
{
  "Ds_Amount":"1000",
  "Ds_Currency":"978",
  "Ds_Order":"1552572812",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_Response":"0000",
  "Ds_AuthorisationCode":"694432",
  "Ds_TransactionType":"0",
  "Ds_SecurePayment":"1",
  "Ds_Language":"1",
  "Ds_CardNumber":"454881******0004",
  "Ds_Card_Type":"C",
  "Ds_MerchantData":"",
  "Ds_Card_Country":"724",
  "Ds_Card_Brand":"1"
}
```

7. Transacciones Autenticadas con DCC

7.1 Pasos para realizar una transacción con autenticación y DCC

A continuación, se detallarán todas aquellas características adicionales para una transacción con autenticación en la que se desee utilizar la operativa DCC para comercios que utilicen la interfaz REST y cuya entidad les haya activado previamente el servicio.

Partiendo de los pasos necesarios para la realización de una transacción con autenticación, incluiremos la parte especifica de una operativa con:

- **Paso 1: Iniciar petición**

El comercio deberá hacer una consulta al TPV Virtual para saber si la tarjeta está inscrita en 3DSecure v2 y poder iniciar el proceso de autenticación y si está tarjeta ofrece **DCC**.

- **Paso 2: 3DSMethod (Si procede)**

El comercio ejecuta el 3DSMethod para que el emisor capture la información del dispositivo.

- **Paso 3: Solicitud de autorización**

El comercio enviará la solicitud de autorización de la operación incluyendo el resultado del 3DSMethod y otros datos adicionales del protocolo EMV3DS. Además, incluyendo la información de **DCC** obtenida en el paso 1.

El TPV Virtual iniciará la autenticación, donde se podrá obtener como resultado:

d. Autenticación OK (Frictionless): la operación ha sido autenticada y el TPV Virtual continuará el proceso de autorización.

e. Challenge Required: La entidad emisora requiere verificar la autenticidad del cliente

f. Otro resultado: autenticación no disponible, autenticación rechazada, error en la autenticación, etc.

El TPV Virtual decidirá según el caso autorizar o rechazar la operación.

- **Paso 4: Autenticación (Si procede)**

La entidad emisora requiere que el titular verifique su autenticidad (mediante OTP, contraseña estática, biometría, etc).

- **Paso 5: Confirmación de autorización**

El comercio enviará la autorización con el resultado del challenge al TPV Virtual para finalizar el proceso de autorización.

Recomendamos que en el paso 3 el comercio proporcione toda la información adicional para aumentar la probabilidad de flujo frictionless y una mayor tasa de autorización.

7.2 Pasos para realizar una transacción con DCC

A continuación, se detallan solamente aquellos pasos que cambiar con respecto a una autorización sin DCC.

7.2.1 Iniciar Petición

Para iniciar la petición de una operación con autenticación y EMV3DS se deberán seguir el apartado Iniciar Petición añadiendo los datos de DCC.

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de inicia petición al Servicio REST con autenticación y 3DS:


```
{
  "DS_MERCHANT_ORDER":1552580496,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_PAN":"XXXXXXXXXXXXXXXXXX",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_AMOUNT":"1000",
   "DS_MERCHANT_DCC":"Y"
   "DS_MERCHANT_EMV3DS": {"threeDSInfo":"CardData"}
 }
```


Como respuesta se obtendrá lo siguiente:


```
{
  "Ds_Order":"1552580496",
  "Ds_MerchantCode":"999008881",
  "Ds_Terminal":"2",
  "Ds_TransactionType":"0",
  "Ds_DCC":{
    "InfoMonedaTarjeta":{
      "monedaDCC":"840",
      "litMonedaDCC":"DOLAR U.S.A.",
      "litMonedaRDCC":"USD",
      "importeDCC":"11.50",
      "cambioDCC":"0.869841",
      "fechaCambioDCC":"2019-01-16",
      "markUp":"0.03"
    },
    "InfoMonedaTitular":{
      "monedaCome":"978",
      "litMonedaCome":"EUR",
      "importeCome":"10.00"
    }
   },
   "Ds_EMV3DS": {
     "protocolVersion":"2.1.0",
     "threeDSServerTransID":"8de84430-3336-4ff4-b18d-f073b546ccea",
     "threeDSInfo":"CardConfiguration",
     "threeDSMethodURL":"https://sis.redsys.es/sis-simulador-web/threeDsMethod.jsp"
    }
}
```


7.2.2 Petición de autorización con DCC

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autorización con DCC al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552581014,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":"XXXXXXXXXXXXXXXXX",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX",
  "DS_MERCHANT_DCC":{
                        "monedaDCC":"840",
                        "importeDCC":"11.50"
                                },
   "DS_MERCHANT_EMV3DS":
            {
                        "threeDSInfo":"AuthenticationData",
                        "protocolVersion":"2.1.0",
                        "browserAcceptHeader":"text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8,ap
            plication/json",
                        "browserUserAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36",
                        "browserJavaEnabled":"false",
                        "browserLanguage":"ES-es",
                        "browserColorDepth":"24",
                        "browserScreenHeight":"1250",
                        "browserScreenWidth":"1320",
                        "browserTZ":"52",
                        "threeDSServerTransID":"8de84430-3336-4ff4-b18d-f073b546ccea",
                        "notificationURL":"https://sis-d.redsys.es/sis-simulador-
 web/SisRESTCreqCres_3DSecureV2.jsp",
                        "threeDSCompInd":"Y"
            }

}
```

8. Adaptaciones PSD2 (Septiembre 2019)

De acuerdo a la norma de PSD2 (entrada en vigor el 14 de septiembre de 2019) para la autenticación de todas las peticiones, se definen una serie de exenciones al uso de SCA que podrán venir marcadas en la petición. Para esta operativa se requiere activación por parte de la entidad (*más información sobre PSD2 en la guía de uso general del servicio "TPV Virtual Guía SIS.pdf"*).

PARÁMETRO	VALORES POSIBLES
DS_MERCHANT_EXCEP_SCA	MIT , LWV, TRA , COR

- LWV : exención por bajo importe (hasta 30 €, con máx. 5 ops. o 100 € acumulado por tarjeta, estos contadores son controlados a nivel de entidad emisora de la tarjeta)
- TRA: exención por utilizarse un sistema de análisis de riesgo (y considerarse bajo riesgo) por parte del adquirente/comercio.
- MIT: operación iniciada por el comercio (sin estar asociada a una acción o evento del cliente) que están fuera del alcance de la PSD2. Este es el caso de las operativas de pagos de subscripciones, recurrentes..etc todas las que requieren el almacenamiento de las credenciales de pago del cliente (COF) o su equivalente mediante operativas de pagos programados tokenizados (uso funcionalidad "pago por referencia" en pagos iniciados por el comercio). Toda operativa de pago iniciada por el comercio (MIT) requiere que inicialmente cuando el cliente concede el permiso al comercio de uso de sus credenciales de pago, dicho "permiso o mandato" se haga mediante operación autenticada con SCA.
- COR: exención restringida a los casos de uso de un protocolo pago corporativo seguro

*Se deberá tener en cuenta que para las exenciones LWV, TRA y COR la primera opción será marcar la exención en el paso de la autenticación, para mejorar la experiencia de usuario. Esto permite que si el emisor no quiere aceptar la propuesta de exención y requiere SCA pueda solicitar la autenticación en el mismo momento sin necesidad de rechazar la operación (challenge required EMV3DS).

8.1 Transacciones iniciadas por el comercio (MIT)

Las transacciones iniciadas por el comercio sin que haya interacción posible con el cliente. Por ejemplo, pago mensual de un recibo o cuota de subscripción. Este tipo de exención requiere el marcaje de la operativa como COF (Credencial on File) de acuerdo al uso concreto que se esté haciendo de las credenciales almacenadas.

Sin embargo no todas las operativas en las que se utilizan datos de tarjeta/credenciales almacenadas (COF) son consideradas MIT. Por ejemplo, la operativa de pago en 1 clic, donde las credenciales del

cliente están almacenadas o tokenizadas (pago por referencia) con el objetivo de facilitar al máximo el momento del pago sin tener que solicitarlas de nuevo al cliente, no se puede considerar una transacción iniciada por el comercio. En tal caso según PSD2, y mientras no se aplique otra exención, se requiere el uso de autenticación reforzada (SCA).

No obstante, con PSD2 estando en vigor, toda operativa de pago iniciada por el comercio (MIT) requiere inicialmente una operación autenticada con SCA que es aquella en la que el cliente concede el permiso y acuerda con el comercio las condiciones para que se usen sus datos de pago para cargos posteriores de acuerdo a un servicio prestado continuado en el tiempo. Esta operativa debe también marcarse debidamente siguiendo la especificación Card-on-file (COF) para indicar que los datos de tarjeta se están almacenando para pagos posteriores.

NOTA: El listado completo de todos los parámetros de entrada del SIS se presenta en la hoja de cálculo "TPV-Virtual Parámetros Entrada-Salida.xlsx".

NOTA2: Acceder a la Guía COF para más información "Especificaciones COF ECom v1.1.pdf".

8.1.1 Transacciones MIT y uso de tokenización (pago por referencia)

En muchos casos se suele utilizar la tokenización de las credenciales de pago del cliente para que Redsys se encargue del almacenamiento seguro de los mismos y asegurar el cumplimiento de los estándares de seguridad de PCI DSS, con el objetivo de generar más tarde pagos iniciados por el comercio sin estar presente el titular de la tarjeta.

En estos casos, en la transacción inicial en la que se solicitar el token o referencia, bajo PSD2 se debe utilizar 3D Secure para aplicar autenticación reforzada y además se debe marcar adecuadamente mediante los parámetros COF el uso que se dará a la misma, de forma que en usos posteriores iniciados por el comercio con el token/referencia, el propio tpv virtual SIS incorpore de forma automática la información de marcaje de uso adecuada e información adicional requerida según la marca de la tarjeta (pej: id transacción original requerido para los pagos COF en Visa "DS_MERCHANT_COF_TXNID")

9. Entorno 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 en endpoint del servicio para realizar las pruebas. Para obtener los datos de acceso, deberán dirigirse a su entidad bancaria para que ésta les proporcione los datos de acceso.

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

https://sis-t.redsys.es:25443/sis/rest/trataPeticionREST

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

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

NOTA: El entorno de pruebas será idéntico al entorno real, con la única diferencia que 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 de su comercio, deberá contactar con su entidad bancaria.

DATOS GENÉRICOS DE PRUEBA

- Número de comercio (Ds_Merchant_MerchantCode): Aquí se deberá poner el número facilitado por su entidad (ejemplo 999008881)
- Terminal (Ds_Merchant_Terminal): Aquí se deberá poner el número facilitado por su entidad (ejemplo 01)
- Clave secreta: sq7HjrUOBfKmC576ILgskD5srU870gJ7

Tarjetas Autorizadas:

- Tarjeta autorizada 1:
  - Numeración: 4548812049400004
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta aceptada 2 (EMV3DS *con "iniciapeticion"* NO_3DS_V2):
  - Numeración: 4548812049400004

- Caducidad: 12/20
- Código CVV2: 123
- Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.

- Tarjeta aceptada 3 (EMV3DS *con "iniciapeticion" con threeDSMethodURL con autenticación FRICTIONLESS)*
  - Numeración: 4918019160034602
  - Caducidad: 12/20
  - Código CVV2: 123

    Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta aceptada 4 (EMV3DS *con "iniciapeticion" sin threeDSMethodURL con autenticación FRICTIONLESS)*
  - Numeración: 4548814479727229
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta aceptada 5 (EMV3DS *con "iniciapeticion" con threeDSMethodURL con autenticación CHALLENGE)*
  - Numeración: 4918019199883839
  - Caducidad: 12/20
  - Código CVV2: 123

    Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta aceptada 6 (EMV3DS *con "iniciapeticion" sin threeDSMethodURL con autenticación CHALLENGE)*
  - Numeración: 4548817212493017
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.

Tarjetas Denegadas (Código de respuesta 190):

- Tarjeta denegada 1:
  - Numeración: 5576440022788500
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta denegada 2 (EMV3DS *con "iniciapeticion"* NO_3DS_V2):
  - Numeración: 5576440022788500
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta denegada 3 (EMV3DS *con "iniciapeticion" con threeDSMethodURL con autenticación FRICTIONLESS)*
  - Numeración: 4907277775205123
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.
- Tarjeta denegada 4 (EMV3DS *con "iniciapeticion" con threeDSMethodURL con autenticación CHALLENGE)*
  - Numeración: 4907271141151707
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.

Tarjetas Rechadas en la autenticación EMV3DS (SIS0598):

- Tarjeta rechazada 1 (EMV3DS *con "iniciapeticion" sin threeDSMethodURL)*
  - Numeración: 5410082854557833

- Caducidad: 12/20
- Código CVV2: 123
- Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.

- Tarjeta rechazada 2 (EMV3DS *con "iniciapeticion" sin threeDSMethodURL)*
  - Numeración: 5410088208000685
  - Caducidad: 12/20
  - Código CVV2: 123
  - Para compras seguras, en la que se requiere la autenticación del titular, el código de autenticación personal (CIP) es 123456.

10. Códigos de error

En este apartado se muestra la manera de informar los posibles errores que se pueden producir en el proceso de integración.

El error que se ha producido se informará en el parámetro *errorCode*, tal y como se muestra a continuación:

{"errorCode":["SIS0042"]}

NOTA: El listado completo de todos los errores del SIS se presenta en la hoja de cálculo "TPV-Virtual Parámetros Entrada-Salida.xlsx".

11. Parámetros de entrada y salida

11.1 Parámetros de la solicitud

En la petición de pago hacia el TPV Virtual SIS se tendrán que enviar una serie de datos obligatorios y otros opcionales, que irán en función del tipo de operación y operativa que se desee realizar.

NOTA: El listado completo de todos los parámetros de entrada del SIS se presenta en la hoja de cálculo "TPV-Virtual Parámetros Entrada-Salida.xlsx".

En los siguientes puntos se mostrarán algunos ejemplos de peticiones:

11.1.1 Petición de pago/preautorización (con envío de datos de tarjeta)

A continuación, se muestra un ejemplo del parámetro Ds_Merchant_Parameters, previo a ser codificado en Base 64:

```
{  "DS_MERCHANT_ORDER":"1552565870",
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"999",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":"XXXXXXXXXXXXX",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX"}
```

* DS_MERCHANT_TRANSACTIONTYPE":"0" para PAGO
* DS_MERCHANT_TRANSACTIONTYPE":"1" para PREAUTORIZACIÓN

11.1.2 Petición de Confirmación/Devolución/Anulación

A continuación, se muestra un ejemplo del parámetro Ds_Merchant_Parameters, previo a ser codificado en Base 64:

```
{  "DS_MERCHANT_ORDER":"1552565870",
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"999",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"3",
  "DS_MERCHANT_AMOUNT":"1000"}
```

* DS_MERCHANT_TRANSACTIONTYPE":"2" para CONFIRMACIÓN
* DS_MERCHANT_TRANSACTIONTYPE":"3" para DEVOLUCIÓN
* DS_MERCHANT_TRANSACTIONTYPE":"9" para ANULACIÓN

11.1.3 Petición de Tokenización (Pago por Referencia - Pago 1-Clic)

A continuación, se muestra un ejemplo del parámetro Ds_Merchant_Parameters, previo a ser codificado en Base 64:

```
{ "DS_MERCHANT_ORDER":"1552565870",
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"999",
```

"DS_MERCHANT_CURRENCY":"978",
"DS_MERCHANT_TRANSACTIONTYPE":"0",
"DS_MERCHANT_AMOUNT":"1000",
"DS_MERCHANT_PAN":"XXXXXXXXXXXX",
"DS_MERCHANT_EXPIRYDATE":"XXXX",
"DS_MERCHANT_CVV2":"XXX"
" DS_MERCHANT_IDENTIFIER ":" REQUIRED "}

11.1.4 Petición de pago con Tokenización (Pago por Referencia - Pago 1-Clic)

A continuación, se muestra un ejemplo del parámetro Ds_Merchant_Parameters, previo a ser codificado en Base 64:

{ "DS_MERCHANT_ORDER":"1552565870",
"DS_MERCHANT_MERCHANTCODE":"999008881",
"DS_MERCHANT_TERMINAL":"999",
"DS_MERCHANT_CURRENCY":"978",
"DS_MERCHANT_TRANSACTIONTYPE":"0",
"DS_MERCHANT_AMOUNT":"1000",
" DS_MERCHANT_IDENTIFIER ":"XXXXXXXXXXXXXXXXXXXXXXX"}

11.2 Parámetros de la respuesta

Las peticiones REST generan una respuesta síncrona a la petición enviada por el comercio, en la cual se informará el resultado de la operación.

El resultado de la operación se informará mediante el parámetro Ds_Response o "Código de respuesta". Además, se informará dicho código de respuesta en la consulta de operaciones, siempre y cuando la operación no está autorizada, tal y como se muestra en la siguiente imagen:

Fecha	Tipo operación	Número de pedido	Resultado operación y código	Importe
29/06/2018 10:01:44	Autorización	290618100053	Autorizada 101311	1,00 EUR
29/06/2018 10:46:55	Autorización	5674	Sin Finalizar 9998	1,45 EUR
29/06/2018 10:54:06	Autorización	7907vPBMh	Sin Finalizar 9998	1,45 EUR

Un ejemplo de respuesta de una operación de pago autorizada sería la siguiente:

{
"Ds_Amount":"1000",
"Ds_Currency":"978",
"Ds_Order":"1552568529",
"Ds_MerchantCode":"999008881",
"Ds_Terminal":"2",
"Ds_Response":"0000",

"Ds_AuthorisationCode":"842841",
"Ds_TransactionType":"0",
"Ds_SecurePayment":"0",
"Ds_Language":"1",
"Ds_CardNumber":"XXXXXXXXXXXXXXXX",
"Ds_Card_Type":"C",
"Ds_MerchantData":"",
"Ds_Card_Country":"724",
"Ds_Card_Brand":"1"
}

NOTA2: El listado completo de todos los parámetros de salida del SIS se presenta en la hoja de cálculo "TPV-Virtual Parámetros Entrada-Salida.xlsx".

12. Integración para PSP

*Se requiere activación por parte de la Entidad.

Si eres un agregador de comercio o PSPs hay una integración específica para que, de esta forma, con clave secreta para PSP puedan operar en nombre de los comercios a nivel de terminal.

Para estas peticiones los parámetros a enviar son los mismos que en la petición habitual de comercios, pero estos tendrán una versión de firma y firma en ANSI X9.19.

No se definen flujos específicos para PSP. Los parámetros de entrada y salida así como los códigos de error se podrán ver en Parámetros de entrada y salida y Códigos de error.

12.1 Configuración

Los comercios-terminales deberán estar asociados y configurados para poder enviar peticiones desde un PSP. Esta solicitud la debe realizar a la entidad el propio comercio.

12.2 Solicitud y recepción de claves

Se recibirán dos claves privadas con el protocolo establecido por Redsys.

- Clave para realizar cifrado 3DES del campo DS_MERCHANT_PAN
- Clave para firmar la petición (DS_MERCHANTPARAMETERS) según la normal X9.19

12.3 Envío de petición al TPV Virtual

Al igual que en la petición de pago enviada por un comercio, el PSP tiene los campos siguientes variando como hemos comentado antes la firma y su versión de firma.

- Ds_SignatureVersion: Constante que indica la versión de firma que se está utilizando.
- Ds_MerchantParameters: Cadena en formato JSON con todos los parámetros de la petición codificada en Base 64 y sin retornos de carro (En la sección de anexos se incluye la lista de parámetros que se pueden enviar en una solicitud de pago). Si se envía el campo DS_MERCHANT_PAN este debe estar firmado con 3DES.
- Ds_Signature: Firma de los datos enviados. Es el resultado del X9.19 de la cadena JSON codificada en Base 64 enviada en el parámetro anterior.

Dichos parámetros deben enviarse a los siguientes endpoints dependiendo de si se quiere realizar una petición en el entorno de prueba u operaciones reales:

URL Conexión	Entorno
https://sis-t.redsys.es:25443/sis/rest/trataPeticionPSPREST	Pruebas

https://sis.redsys.es/sis/rest/trataPeticionPSPREST	Real

12.4 Recepción del resultado

La recepción del resultado será firmada de la misma forma que la petición de envío, según la norma ANSI x9.19

12.5 Envío de petición

El PSP deberá montar los parámetros de la petición de pago que debe hacer llegar al TPV Virtual a través de una petición Rest al igual que en la integración de comercios.

12.5.1 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 para esta integración de PSP el valor a utilizar será **T25V1** en el parámetro **Ds_SignatureVersion**

12.5.2 Montar la cadena de datos de la petición

Se debe montar una cadena con todos los datos de la petición en formado JSON, para más información de como montar la cadena accede al apartado de la integración para comercios Montar la cadena de datos de la petición

Importante:¿Vas a enviar el parámetro DS_MERCHANT_PAN? Para aquellas integraciones en las que sea necesario enviar el PAN de la tarjeta, este deberá estar cifrado en 3DES con la clave facilitada para tal caso al PSP.

12.5.3 Firmar los datos de la petición

Para calcular la firma es necesario utilizar la clave específica que se ha facilitado al PSP independientemente del terminal con el que se realice la operación.

1. Una vez se tiene montada la cadena de datos a firmar codificada en base 64 (Ds_MerchantParameters) se genera una Mac sobre esta cadena de datos de acuerdo a la norma ANSI X9.19.
2. El campo Ds_Signature se obtiene convirtiendo a hexadecimal la Mac obtenida en el paso anterior y cogiendo los 4 primeros bytes (8 caracteres) iniciando por la izquierda del resultado.

12.5.4 Ejemplo de peticiones

Cadena en JSON

*{"DS_MERCHANT_AMOUNT":"145","DS_MERCHANT_ORDER":"1446068581","DS_MERCHANT_MERCHANTCODE":"999008881","DS_MERCHANT_CURRENCY":"978","DS_MERCHANT_TRANSACTIONTYPE":"0","DS_MERCHANT_TERMINAL":"1","DS_MERCHANT_MERCHANTURL":"http:\/\/www.prueba.com\/urlNotificacion.php","DS_MERCHANT_PAN":"454881********04","DS_MERCHANT_EXPIRYDATE":"1512","DS_MERCHANT_CVV2":"123"}*

Ciframos en 3DES el parámetro DS_MERCHANT_PAN

Número de tarjeta	*454881********04*
Número de tarjeta hexadecimal cifrado en 3DES modo CBC sin vector, clave F180E06B7A89A88F4A2A52C8EC1C5D1C	*377f34989cf0ae79857a70aa45f3e4c3*

Cadena en JSON con PAN cifrado:

{"DS_MERCHANT_AMOUNT":"145","DS_MERCHANT_ORDER":"1446068581","DS_MERCHANT_MERCHANTCODE":"999008881","DS_MERCHANT_CURRENCY":"978","DS_MERCHANT_TRANSACTIONTYPE":"0","DS_MERCHANT_TERMINAL":"1","DS_MERCHANT_MERCHANTURL":"http:\/\/www.prueba.com\/urlNotificacion.php","DS_MERCHANT_PAN":"377f34989cf0ae79857a70aa45f3e4c3","DS_MERCHANT_EXPIRYDATE":"1512","DS_MERCHANT_CVV2":"123"}

A continuación, se muestra el objeto JSON codificado en BASE64, campo DS_MERCHANT_PARAMETERS:

eyJEU19NRVJDSEFOVF9BTU9VTlQiOiIxNDUiLCJEU19NRVJDSEFOVF9PUkRFUiI6IjE0NDYwNjg1ODEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVENPREUiOiI5OTkwMDg4ODEiLCJEU19NRVJDSEFOVF9DVVJSRU5DWSI6Ijk3OCIsIkRTX01FUkNIQU5UX1RSQU5TQUNUSU9OVFlQRSI6IjAiLCJEU19NRVJDSEFOVF9URVJNSU5BTCI6IjEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVFVSTCI6Imh0dHA6XC9cL3d3dy5wcnVlYmEuY29tXC91cmxOb3RpZmljYWNpb24ucGhwIiwiRFNfTUVSQ0hBTlRfUEFOIjoiMzc3ZjM0OTg5Y2YwYWU3OTg1N2E3MGFhNDVmM2U0YzMiLCJEU19NRVJDSEFOVF9FWFBJUllEQVRFIjoiMTUxMiIsIkRTX01FUkNIQU5UX0NWVjIiOiIxMjMifQ==

Firmar los datos de la petición

Sobre toda la cadena obtenida en el paso anterior (DS_MERCHANT_PARAMETERS) se calcula la firma completa en base a la norma x9.19.

Obtenemos la MAC *5D823A402DE70705*con la clave de cifrado *269289DA6EAD0B20928C8F2F2F6BC752* y el ds_merchant_parameters.

El campo ds_signature será cogiendo los 4 primeros bytes (8 caracteres) iniciando por la izquierda del resultado del MAC del paso anterior ***5D823A40***

Formar el mensaje de la petición

- Ds_SignatureVersion: *T25V1*
- Ds_MerchantParameters:
  eyJEU19NRVJDSEFOVF9BTU9VTlQiOiIxNDUiLCJEU19NRVJDSEFOVF9PUkRFUiI6IjE0NDYwNjg1ODEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVENPREUiOiI5OTkwMDg4ODEiLCJEU19NRVJDSEFOVF9DVVJSRU5DWSI6Ijk3OCIsIkRTX01FUkNIQU5UX1RSQU5TQUNUSU9OVFlQRSI6IjAiLCJEU19NRVJDSEFOVF9URVJNSU5BTCI6IjEiLCJEU19NRVJDSEFOVF9NRVJDSEFOVFVSTCI6Imh0dHA6XC9cL3d3dy5wcnVlYmEuY29tXC91cmxOb3RpZmljYWNpb24ucGhwIiwiRFNfTUVSQ0hBTlRfUEFOIjoiMzc3ZjM0OTg5Y2YwYWU3OTg1N2E3MGFhNDVmM2U0YzMiLCJEU19NRVJDSEFOVF9FWFBJUllEQVRFIjoiMTUxMiIsIkRTX01FUkNIQU5UX0NWVjIiOiIxMjMifQ==
- Ds_Signature: *5D823A40*

13. ¿Operas por PUCE? Protocolo unificado de comercios

*Se requiere activación por parte de la Entidad.

Para comercios/integradores que envían las autorizaciones de comercio electrónico por protocolo PUC (también conocida como PRICE Comercios) y además desean poder aplicar autenticación EMV3DS pueden:

1. Utilizar el protocolo REST descrito en esta guía para gestionar la autenticación 3D Secure del titular (EMV3DSv2 y EMV3DSv1)
2. Posteriormente enviar la autorización por una conexión PUC/PRICE existente. El envío de la autorización financiera a través de PUC no es estrictamente necesario (ya que con este mismo REST es posible realizar también este paso) pero está disponible para los comercios/instalaciones que ya tienen este protocolo implantado en sus sistemas contables.

En este punto se detallan las particularidades (ej: envío del dato de case) a incorporar en el mensaje de solicitud de autenticación REST para el envío con control de clave de PSP y poder utilizar los datos de la autenticación 3DS obtenida para ligarlos a una autorización enviada por la conexión PUC/E

NOTA: Al operar con protocolo PUCE/PUC solo se realizará por el TPV-Virtual la autenticación (EMV3DS), teniendo que usar el PUC para DCC, tokenización, COF y la exencion(MIT) de PSD que van en los mensajes de autorización.

13.1 Mensaje de solicitud de autenticación

El mensaje de solicitud de autenticación seguirá la misma estructura que los mensajes definidos en este documento. Los parámetros de entrada se definen en el siguiente apartado Parámetros de entrada y salida.

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autenticación PUCE al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552572812,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"17",
  "DS_MERCHANT_MATCHINGDATA ":" 35VSTCh0dHA6XC9cL3d3d",
  "DS_ACQUIRER_IDENTIFIER ":"185187",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
  "DS_MERCHANT_CVV2":"XXX
}
```

Para las operaciones con autenticación EMV3DS se tendrán en cuenta las peticiones descritas en el apartado Transacciones con Autenticación EMV3DS. Los diagramas serán simulares a los expuestos,

teniendo en cuenta no se realiza la autorización por el TPV- Virtual, en los casos de Frictionless se deberá realizar una petición más con la autorización al PUC y en los casos de Challenge la autorización final no se deberá enviar al TPV-Virtual si no al PUC.

13.2 Mensaje de respuesta a solicitud de autenticación

El mensaje de respuesta de autenticación seguirá la misma estructura que los mensajes definidos en este documento. Los parámetros se definen en apartado Parámetros de entrada y salida.

13.3 Mensaje de solicitud de autorización existente por protocolo PUC

Este mensaje no sufrirá cambios. Revisar la Guía de PUCE.

14. MPI Externo EMV3DS

*Se requiere activación por parte de la Entidad.

En el caso de que se disponga de un MIP certificado externo a Redsys, se podrán mandar las peticiones y en estas indicar que la operación viene autenticada. Este MPI para poder realizar operaciones EMV3DS deberá estar certificado con EMVCO y con las marcas.

Las peticiones se realizarán de la misma forma que se ha explicado en el documento y añadiendo en el DS_MERCHANT_PARAMETERS, el parámetro DS_MERCHANT_MPIEXTERNAL, del tipo JSON Object. Acceder al apartado Parámetros de entrada y salida para conocer el detalle del parámetro de los campos a enviar

Los valores que debe tener el parámetro DS_MERCHANT_MPIEXTERNAL son los siguientes:

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autenticación con MPI en versión 1.0.2 al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552572812,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
 "DS_MERCHANT_CVV2":"XXX",
 " DS_MERCHANT_MPIEXTERNAL ":{
  "TXID":" VFVSTCI6Imh0dHA6XC9cL3d3dy5wcnVlYmEuY23456gh67j789",
  "CAVV":" Y23456gh67j7Imh0dHA6XC9cL3d3dy5VlYmEu89VFVSTCI6wcn" ,
  "ECI":"5"
  }
}
```

A continuación, se describen los datos de debe incluir el Ds_MerchantParameters para enviar una petición de autenticación con MPI en versión 2 al Servicio REST:

```
{
  "DS_MERCHANT_ORDER":1552572812,
  "DS_MERCHANT_MERCHANTCODE":"999008881",
  "DS_MERCHANT_TERMINAL":"2",
  "DS_MERCHANT_CURRENCY":"978",
  "DS_MERCHANT_TRANSACTIONTYPE":"0",
  "DS_MERCHANT_AMOUNT":"1000",
  "DS_MERCHANT_PAN":" XXXXXXXXXXXXXXXXX ",
  "DS_MERCHANT_EXPIRYDATE":"XXXX",
 "DS_MERCHANT_CVV2":"XXX"",
 " DS_MERCHANT_MPIEXTERNAL ":{
  "threeDSServerTransID":" FVSTCI6Imh0A6XC9cL3d3dy5wcnVlYmEuY23",
  "authenticacionValue":" 7Imh0dHA6XC9cL3d3dy5VlYmEu81" ,
  "protocolVersion":"2.1.0"
  }
}
```

15. Timeout

Que hacer en el caso de que el TPV Virtual no responda a una petición solicitada.

Este problema puede tener dos posibles causas:

- No se ha recibido la petición, con lo que TPV Virtual no responderá al mensaje de petición.
- TPV Virtual ha recibido el mensaje de petición, pero no puede contactar con el Centro Autorizador. Esta conexión tiene definido un timeout de 30 segundos, por lo que si transcurrido ese tiempo, no se recibe respuesta del Centro Autorizador, se devolverá un mensaje de respuesta con código 9912/912 "Emisor no disponible". La aplicación cliente deberá por tanto establecer un timeout mayor (unos 40 o 50 segundos), para asegurar que TPV Virtual siempre le va a responder.

15.1 Que hacer en caso de timeout del TPV Virtual

Para las peticiones de un pago o una preautorización se deberá mandar una anulación, si la petición es de una confirmación se deberá mandar una devolución.

En el caso de operaciones de devoluciones u operaciones de anulaciones se podrá volver a realizar la petición.

16. Errores frecuentes

Error de firma

Cuando hay un error de firma el comercio ha de verificar:

- Que los datos que se han utilizado para hacer la firma son iguales a los que se envían en el formulario, teniendo en cuenta, que cualquier modificación del valor o formato de un campo posterior al cálculo de la firma, hace que ésta sea incorrecta.
- Que la clave secreta empleada por el comercio coincide con la clave que tiene cargada el comercio en el módulo de administración (apartado comercios).
- Se debe revisar que los comercios no están enviando espacios en blanco en la firma. Si la petición se hace mediante cURL o mediante el navegador Safari, puede que se conviertan los símbolos "+" en espacio en blanco. Para que esto no ocurra se deben sustituir los símbolos "+" de la firma por "%2B" (Valor URL encoded).
- Si el comercio no consigue localizar qué parámetro es el erróneo, debe contactar con el Centro de Atención al Cliente de Redsys, o con el departamento de Soporte a la Integración de Redsys, si su entidad le ha facilitado el contacto.

Tengo en mi comercio denegaciones por número de repetido, pero no tengo constancia de haberlos repetido.

Esto ocurre habitualmente porque la plataforma del comercio está generando números de pedido repetido únicamente cuando recibe denegaciones o autorizaciones, pero los está repitiendo cuando las transacciones se quedan a medias. Ante esto hay dos opciones:

- Solicitar al servicio de Soporte que el TPV se configure para que pueda repetir números de pedidos. Máximo de una operación autorizada al día y sin límite para las denegadas.
- Generar siempre números de pedido distintos, no solo para las operaciones autorizadas y denegadas, sino para aquellas que no hayan finalizado trascurrido un tiempo.

Necesito hacer una devolución de una operación, pero no me aparece la opción de devolución en el módulo de administración.

Se debe a que el usuario con el que se está accediendo a Canales no tiene permiso para hacer devoluciones. Si necesita este permiso debe ponerse en contacto con su entidad.

17. Preguntas Frecuentes

Soy un comercio y necesito conocer la clave de encriptación de mi TPV Virtual

Para ver la clave del TPV Virtual hay que seguir los siguientes pasos:

Acceda a su módulo de administración de su TPV virtual.

Seleccione la opción "comercio" y pulse “ver clave”

Introduzca su contraseña de su usuario del TPV virtual y pulse aceptar. -Tendrá acceso a ver la clave del comercio durante 10 segundos.

Mi usuario de comercio de acceso al módulo de administración del Canales está bloqueado. ¿Cómo puedo desbloquearlo?

Bajo las casillas de usuario y contraseña existe un link de “He olvidado mi contraseña”. Tras pulsarlo deberá escribir su usuario y confirmar la dirección de envío de la nueva contraseña.