COLLECTOR DE PAGOS

Lograr la integración del Formulario API

Guía de implementación

Versión del documento 1.9

Contenido

1. HISTORIAL DEL DOCUMENTO

Versión	Autor	Fecha	Comentario
1.9	Lyra Collect	20/11/2020	• Actualización del capítulo *Enviar un formulario de pago en POST*.
1.8	Lyra Collect	30/07/2020	• Corrección del formato del campo **vads_trans_date**. • Actualización del capítulo *Configurar la notificación al final del pago*.
1.7	Lyra Collect	09/12/2019	• Actualización del procedimiento de configuración de IPN. • Adición del capítulo **Implementar la IPN**. • Corrección del formato del campo **vads_product_label**. • Corrección del formato del campo **vads_trans_id**
1.6	Lyra Collect	17/06/2019	Ahora el algoritmo hash está disponible en el menú Configuración Tienda, pestaña Claves.
1.5	Lyra Collect	23/01/2019	• Actualización del capítulo **Identificarse durante los intercambios** • Sustitución del termino “Certificado” por “Clave” en todos los menús
1.4	Lyra Collect	01/10/2018	Versión inicial

2. OBTENER AYUDA

¿Necesita ayuda? Consulte las preguntas frecuentes en nuestro sitio web

https://docs.lyra.com/es/collect/faq/sitemap.html

Para cualquier pregunta técnica o solicitud de asistencia, contacte *el soporte técnico*.

Para facilitar el procesamiento de sus solicitudes, se le pedirá que informe su código cliente (ejemplo: **CLXXXXX**, **MKXXXXX** o **AGXXXXX**).

Esta información está disponible en el Back Office Vendedor (en la parte superior del menú).

3. ESTABLECER DIÁLOGO CON LA PLATAFORMA DE PAGO

El diálogo entre el sitio web vendedor y la plataforma de pago se realiza mediante un intercambio de datos.

Para crear un pago, estos datos se envían a través de un formulario HTML por el navegador del comprador.

Al final del pago, el resultado se transmite al sitio web vendedor de dos maneras:

- por el navegador cuando el comprador hace clic en el botón para volver al sitio del comerciante.
- automáticamente mediante notificaciones denominadas URL de notificación instantánea (también llamadas IPN para Instant Payment Notification) véase capítulo **Configurar notificaciones**.

Para garantizar la seguridad de los intercambios, los datos se firmarán mediante una clave conocida solamente por el vendedor y la plataforma de pago.

3.1. Redirección del comprador hacia la página de pago

El sitio web vendedor se comunica con la plataforma de pago redirigiendo al comprador a la URL a continuación.

https://secure.lyra.com/vads-payment/

3.2. Identificarse durante los intercambios

Para interactuar con la plataforma de pago, el vendedor necesita dos datos:

- **El identificador de la tienda**: identifica el sitio web vendedor durante los intercambios. Su valor se transmite en el campo **vads_site_id.**
- **La clave**: permite calcular la firma alfanumérica transmitida en el campo de **firma**.

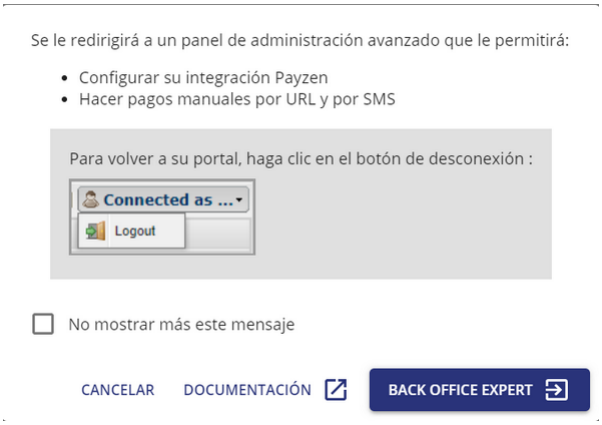
Para recuperar estos valores:

1. Conéctese a su **Back Office Lyra Collect** : *https://secure.lyra.com/portal/*
2. Ingrese su usuario.
3. Ingrese su clave.
4. Haga clic en **Iniciar sesión**.

   En el caso de un error al ingresar el usuario y/o clave, aparece el mensaje de error "*Nombre de usuario o clave inválido*".

   Puede corregir su entrada o hacer clic en el enlace **Olvidé mi clave o la cuenta está bloqueada**.
5. Haga clic en **Más acciones**.

   Se abre la siguiente ventana:



6. Haga clic en **Back Office Expert** para acceder a su Expert Back Office.

7. Haga clic en **Configuración > Tienda**.

8. Seleccione la pestaña **Claves**.

Figura 1: Pestaña Claves

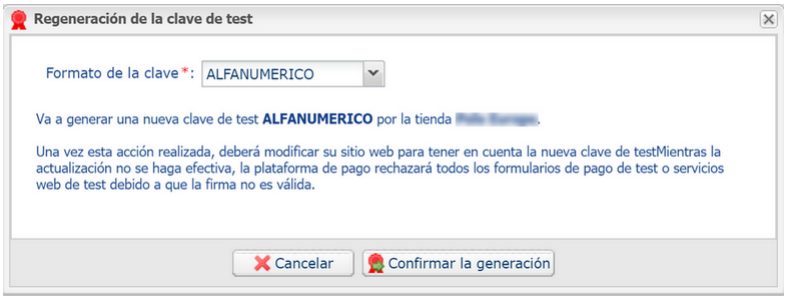
Hay dos tipos de clave disponibles:

- La **clave de test** que genera la firma de un formulario en modo de test.
- La **clave de producción** que genera la firma de un formulario en modo de producción.

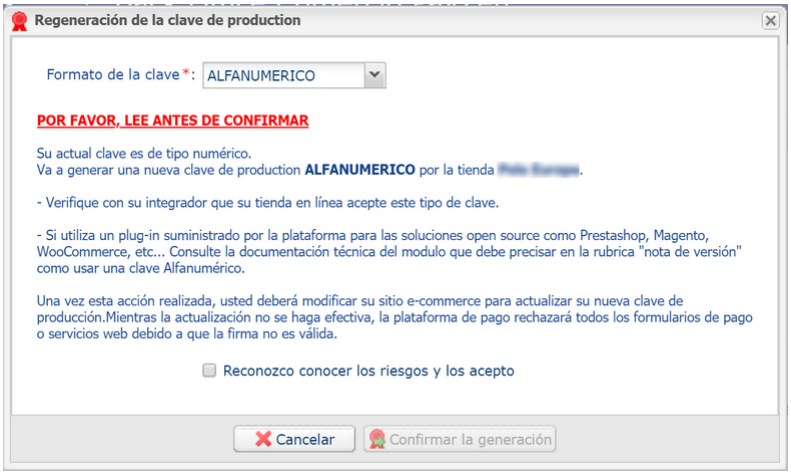
Estas claves pueden ser numéricas o alfanuméricas.

Para máxima seguridad, se recomienda utilizar una clave alfanumérica.

Para cambiar el formato de su clave de prueba, haga clic en el botón **Regenerar clave de prueba** y seleccione el formato ("ALFANUMÉRICO" o "NUMÉRICO").



Para cambiar el formato de su clave de producción, haga clic en el botón **Regenerar clave de producción** y seleccione el formato ("ALFANUMÉRICO" o "NUMÉRICO").



3.3. Gestionar el diálogo al sitio web vendedor

La gestión del diálogo hacia el sitio web vendedor ocurre mediante dos tipos de URL:

- **URL de notificación instantánea**, también llamada IPN (Instant Payment Notification),
- **URL de retorno** al sitio web vendedor.

URL de notificación instantánea - IPN (Instant Payment Notification)

La **URL de notificación** es la URL de una página dedicada en el sitio del comerciante llamada **automáticamente** por la plataforma de pago cuando ocurren eventos particulares.

Por defecto se crean reglas para gestionar los siguientes eventos:

- fin de un pago (aceptado o rechazado),
- abandono o cancelación durante el pago,
- creación o actualización de un token,
- creación de una recurrencia,
- nueva cuota de una recurrencia,
- autorización realizada en el caso de un pago diferido,
- modificación del estado de una transacción por el adquiriente,
- operación realizada desde el Back Office Expert (cancelación, reembolso, duplicación, pago manual, etc.).

Estas reglas deben activarse y configurarse correctamente en función de las necesidades del vendedor.

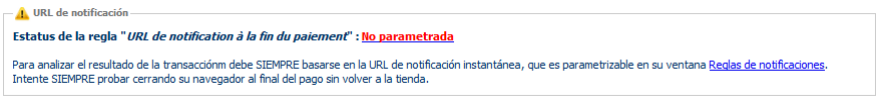
En cada llamado, la plataforma de pago transmite al sitio del comerciante los datos relativos a una transacción. Esto se llama notificación instantánea (o **IPN** para Instant Payment Notification).

Para garantizar la seguridad de los intercambios, los datos se firmarán mediante una clave conocida solamente por el vendedor y la plataforma de pago.

URL de retorno hacia el sitio web vendedor

En el Back Office Expert, el vendedor puede definir las URL de retorno "predeterminadas" desde el menú **Configuración** > **Tienda** > pestaña **Configuración**:

Figura 2: Especificación de las URL de retorno



Puede configurar una URL de retorno a la tienda diferente según el modo.

De forma predeterminada, se redirige el comprador a la URL de retorno, independientemente del resultado del pago.

Sin embargo, si no se configura una URL en este nivel, entonces la redireccion usará la URL principal de la tienda (parámetro **URL** definido en el cuadro **Detalles** de la tienda).

El vendedor tiene la posibilidad de sobrecargar esta configuración en su formulario de pago (véase capítulo **Definir URL de retorno**).

El estado de la regla “URL de notificación al final del pago” (IPN) aparece en esta pantalla. Si no está configurada, tiene que definirla (véase capítulo **Configurar notificaciones**).

3.4. Gestión de la seguridad

Se utilizan diversos métodos con el fin de garantizar la seguridad de las transacciones de pago en línea.

3.4.1. Garantizar la integridad de los intercambios

La integridad de la información intercambiada está garantizada por un intercambio de firmas alfanuméricas entre la plataforma de pago y el sitio web vendedor.

El diálogo entre la plataforma de pago y el sitio web vendedor se realiza mediante el envío de formularios HTML.

Un formulario contiene una lista de campos específicos (ver capítulo **Generar un formulario de pago**) utilizados para generar una cadena.

Este string se convierte luego en un string más pequeño mediante una función hash (SHA-1, HMAC-SHA-256).

*El vendedor puede elegir el algoritmo hash en su Back Office Expert (consulte el capítulo **Seleccionar el algoritmo hash**).*

El string resultante se llama el **resumen** (*digest* en inglés) del string inicial.

El resumen debe transmitirse en el campo de **firma** (consulte el capítulo **Calcular la firma**).

Modelado de mecanismos de seguridad:

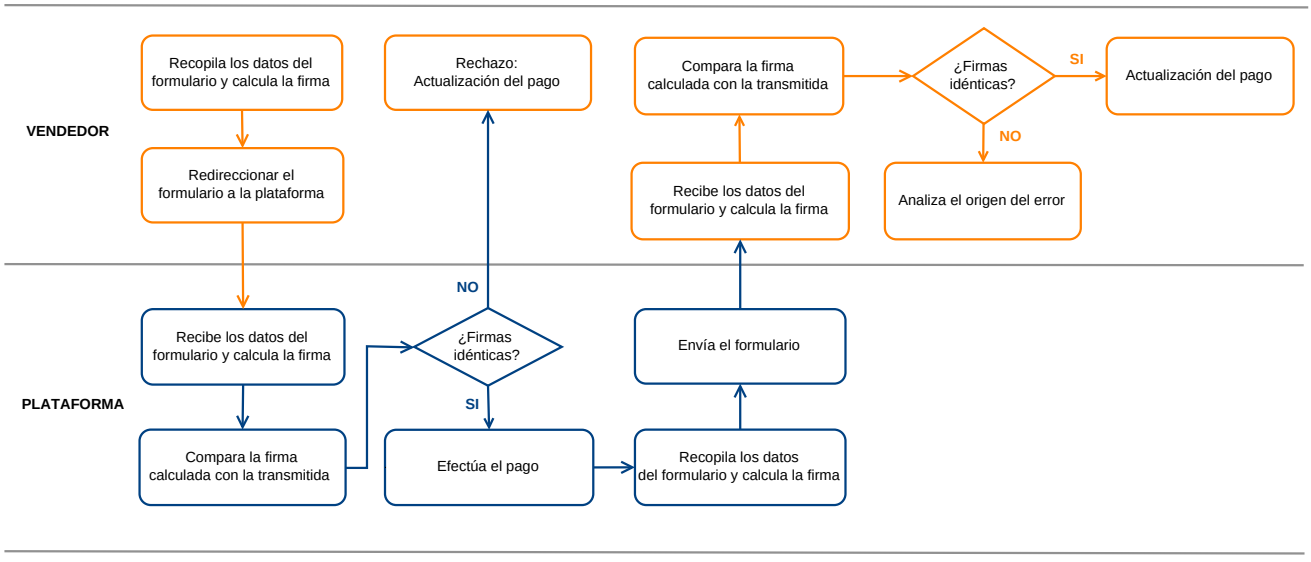


Figura 3: Diagrama mecanismo de seguridad

1. El sitio web vendedor construye los datos del formulario y calcula la firma.
2. El sitio web vendedor envía el formulario a la plataforma.
3. La plataforma recibe los datos del formulario y calcula la firma con los datos recibidos.
4. La plataforma compara la firma calculada con la firma transmitida por el sitio web vendedor.
5. Si las firmas difieren, la solicitud de pago es rechazada.

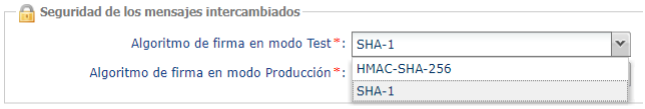
   De lo contrario, la plataforma procede al pago.

6. La plataforma construye los datos de respuesta y calcula la firma de la respuesta.
7. Según la configuración de la tienda (ver capítulo **Configurar notificaciones**), la plataforma transmite el resultado del pago al sitio del comerciante.
8. El sitio web vendedor recibe los datos y calcula la firma. Compara la firma calculada con la firma transmitida por la plataforma.
9. Si las firmas difieren, el vendedor analiza el origen del error (error en el cálculo, intento de fraude, etc.)

   De lo contrario, el sitio web vendedor actualiza su base de datos (estado del stock, estado del pedido, etc.).

3.4.2. Seleccionar el algoritmo hash

Desde el Back Office Expert (menú **Configuración** > **Tienda** > **Claves**), el vendedor tiene la opción de elegir la función de hash que se usará para generar las firmas.



Por defecto, se aplicará el algoritmo HMAC-SHA-256.

Puede seleccionar un algoritmo diferente para el modo de Prueba y para el modo de Producción.

Sin embargo, asegúrese de utilizar el mismo método para generar sus formularios de pago y analizar los datos transmitidos por la plataforma de pago durante las notificaciones.

Para facilitar el cambio de algoritmo, se aceptarán firmas en SHA-1 o HMAC-SHA-256 sin generar rechazo por error de firma durante 24 horas.

3.4.3. Conservar la clave de producción

Desde el primer pago realizado con una tarjeta real, la clave de producción se oculta por razones de seguridad.

Le recomendamos encarecidamente que guarde esta clave en un lugar seguro (archivo cifrado, base de datos, etc.).

En caso de pérdida, el vendedor tendrá la posibilidad de generar uno nuevo desde el Back Office Expert.

Recuerde que puede consultar la clave de producción en el Back Office Expert desde el menú **Configuración** > **Tienda** > pestaña **Claves**.

3.4.4. Gestionar datos sensibles

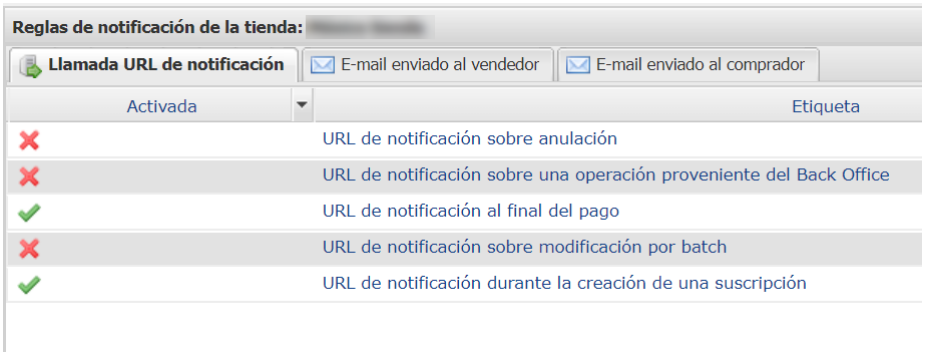
Las reglas estrictas rigen las transacciones de pago en línea (Certificación PCI-DSS).

Como vendedor, debe asegurarse de no transcribir nunca en datos claros que puedan parecerse a un número de tarjeta de crédito. Su formulario será rechazado (código 999 - Datos sensibles detectados).

En particular, evite los números de orden entre 13 y 16 caracteres numéricos que comienzan con 3, 4 o 5.

4. CONFIGURAR NOTIFICACIONES

Para acceder a la gestión de las reglas de notificación, abra el menú: **Configuración** > **Reglas de notificaciones**.



Se muestra la pestaña de configuración de las reglas tipo "Llamada URL de notificación".

4.1. Configurar la notificación al final del pago

Esta regla permite notificar al sitio del comerciante en los siguientes casos:

- Pago aceptado
- Pago rechazado
- Creación o actualización de un token
- Registro de una recurrencia

El evento **Pago aceptado** corresponde a la creación de una transacción en uno de los estados (**vads_trans_status**) a continuación:

- **ACCEPTED**
- **AUTHORISED**
- **AUTHORISED_TO_VALIDATE**
- **CAPTURED**
- **INITIAL**
- **UNDER_VERIFICATION**
- **WAITING_AUTHORISATION**
- **WAITING_AUTHORISATION_TO_VALIDATE**
- **WAITING_FOR_PAYMENT**

Esta notificación es indispensable para comunicar el resultado de una solicitud de pago.

Esta informará al sitio web vendedor el resultado del pago incluso si el cliente no ha hecho clic en el botón Volver a la tienda.

1. Haga clic derecho en la línea **URL de notificación al final del pago**.
2. Seleccione **Gestionar la Regla**.

3. En la sección **Configuración general**, ingrese el campo **Dirección(es) e-mail(s) a notificar en caso de falla**.

   Para especificar varias direcciones de e-mail, sepárelas con un punto y coma.

4. Marque la casilla **Reenvío automático en caso de falla** si desea autorizar a la plataforma a reenviar automáticamente la notificación hasta 4 veces en caso de falla.

   Para más información, consulte el capítulo *Reejecutar automáticamente en caso de fallo* en la página 13.

5. En la sección **URL de notificación de la API formulario V1, V2**, ingrese la URL de su página en los campos **URL a llamar en modo PRUEBA** y **URL a llamar en modo PRODUCCIÓN**.

6. Guarde sus cambios.

4.2. Reejecutar automáticamente en caso de fallo

El reenvío automático no se aplica a las notificaciones activadas manualmente desde el Back Office Expert.

El vendedor puede activar un mecanismo que permita a la plataforma de pago reenviar automáticamente las notificaciones cuando el sitio del comerciante es realmente inalcanzable **hasta 4 veces**.

Una notificación se considerará infructuosa si el código de retorno HTTP devuelto por el sitio del comerciante no se encuentra en la siguiente lista: **200**, **201**, **202**, **203**, **204**, **205**, **206**, **301**, **302**, **303**, **307**, **308**.

Los intentos de llamada se programan a horas fijas cada 15 minutos (00, 15, 30, 45).

Tras cada tentativa infructuosa, se enviará un e-mail de alerta a la dirección especificada en la configuración de la regla de notificación correspondiente.

El asunto del e-mail de alerta contiene el número del intento de enviar la notificación. Se presenta en la forma `attempt #` seguida del número de intento.

- Ejemplo de asunto de un correo electrónico de alerta recibido después de la primera notificación fallida al final de un pago:

```
[MODE TEST] Mi Tienda - Tr. Ref. 067925 / FALLO al invocar a su URL de notificación
 [unsuccessful attempt #1]
```

- Ejemplo de asunto de e-mail recibido en el segundo error:

```
[MODE TEST] Mi Tienda - Tr. Ref. 067925 / FALLO al invocar a su URL de notificación
 [unsuccessful attempt #2]
```

- Ejemplo de asunto de e-mail recibido en el tercer error:

```
[MODE TEST] Mi Tienda - Tr. Ref. 067925 / FALLO al invocar a su URL de notificación
 [unsuccessful attempt #3]
```

- Ejemplo de asunto de e-mail recibido en el último intento:

```
[MODE TEST] Mi Tienda - Tr. Ref. 067925 / FALLO al invocar a su URL de notificación
 [unsuccessful attempt #last]
```

  Para notificar al sitio del comerciante el fallo del último intento de notificación, el asunto del e-mail incluirá `attempt #last`.

Cuando hay reenvío automático, parte de la información no se guarda en la base de datos o se modifica.

Ejemplos de campos no disponibles / no registrados en la base de datos:

Nombre del campo	Descripción
vads_page_action	Operación realizada
vads_payment_config	Tipo de pago (al contado o en vencimientos)
vads_action_mode	Modo de adquisición de la información del medio de pago

Ejemplos de campos enviados con diferentes valores:

Nombre del campo	Nuevo valor
vads_url_check_src	Se asignará el valor **RETRY** en el caso de un reenvío automático.
vads_trans_status	El estado de la transacción puede variar entre la llamada inicial y el reenvío automático (cancelación del vendedor, remesa al banco de la transacción, etc.).
vads_hash	El valor de este campo se regenera en cada llamada.
firma	El valor de la firma depende de los diferentes estados que pueden variar entre la llamada inicial y el reenvío automático.

Estos e-mails detallan:

- el problema encontrado
- los elementos de análisis en función del error
- sus consecuencias
- el procedimiento a seguir desde el Back Office Expert para activar la notificación de forma manual.

Después del cuarto intento, todavía es posible reenviar la URL de notificación **manualmente** desde su Back Office Expert.

Atención, durante el período de reenvío automático, cualquier llamada manual a la URL de notificación afectará el número de reintentos automáticos:

- una llamada manual exitosa detendrá el reenvío automático
- una llamada manual fallida no tendrá ningún impacto en el reenvío automático actual.

4.3. Otros casos de notificación

En función de las opciones comerciales contratadas, la plataforma de pago puede efectuar una llamada a la URL de notificación en los siguientes casos:

- abandono o cancelación de la página de pago por parte del comprador
- reembolso efectuado a través del Back Office Expert
- cancelación de una transacción a través del Back Office Expert
- validación de una transacción a través del Back Office Expert
- modificación de una transacción a través del Back Office Expert
- etc.

Para más información sobre la configuración de las reglas, consulte el manual del usuario *Centro de notificaciones*.

5. ENVIAR UN FORMULARIO DE PAGO EN POST

El sitio web vendedor redirecciona al comprador hacia la plataforma de pago mediante un formulario HTML POST en HTTPS.

Este formulario contiene:

Los siguientes elementos técnicos:

- Las etiquetas `<form>` y `</form>` que permiten crear un formulario HTML.
- El atributo `method="POST"` que especifica el método utilizado para enviar los datos.
- El atributo `action="`*https://secure.lyra.com/vads-payment/*`"` que especifica a dónde enviar los datos del formulario.

Los datos del formulario:

Todos los datos del formulario deben estar codificados en **UTF-8.**

De esta forma, los caracteres especiales (acentos, puntuación, etc.) serán interpretados correctamente por la plataforma de pago. En el caso contrario, el cálculo de la firma será erróneo y el formulario será rechazado.

Lo invitamos a consultar la siguiente tabla para comprender mejor la codificación de formatos.

Notación	Descripción
a	Caracteres alfabéticos (de 'A' a 'Z' y de 'a' a 'z')
n	Caracteres numéricos
s	Caracteres especiales
an	Caracteres alfanuméricos
ans	Caracteres alfanuméricos y especiales (excepto "<" y ">")
3	Longitud fija de 3 caracteres
..12	Longitud variable hasta 12 caracteres
json	JavaScript Object Notation. Un objeto que contiene pares clave/valor separados por comas. Comienza con un refuerzo izquierdo " `{` " y termina con un refuerzo derecho " `}` ". Cada par de clave/valor contiene el nombre de la clave entre comillas dobles seguido de " `:`", y un valor. El nombre de la clave debe ser alfanumérico. El valor puede ser: • una cadena de caracteres (en este caso debe estar encuadrada entre comillas dobles) • un número • un objeto • un tablero • un booleano • vacío Ejemplo: `{"name1":45,"name2":"value2", "name3"=false}`
enum	Caracteriza un campo con un número finito de valores. La lista de valores posibles se da en la definición del campo.
liste d'enum	Lista de valores separados por un " `;` ". La lista de valores posibles se da en la definición del campo. Ejemplo: `vads_payment_cards=VISA;MASTERCARD`
map	Lista de pares clave/valores separados por un " `;` ". Cada par de clave/valor contiene el nombre de la clave seguido de "`=`", y un valor. El valor puede ser: • una cadena de caracteres

Notación	Descripción
	• un booleano • un objeto json • un objeto xml La lista de valores posibles para cada par de clave / valor se proporciona en la definición del campo. Ejemplo: vads_theme_config=SIMPLIFIED_DISPLAY=true;RESPONSIVE_MODEL=Model_1

- Los campos obligatorios:

Nombre del campo	Descripción	Formato	Valor
signature	Firma que garantiza la integridad de las solicitudes intercambiadas entre el sitio web vendedor y la plataforma de pago.	ans	Ex : **ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPy9/rpesfS0=**
vads_action_mode	Modo de adquisición de la información del medio de pago	enum	**INTERACTIVE**
vads_amount	Monto del pago en su unidad monetaria más pequeña(el centavo para el euro)	n..12	Ejemplo: 4525 para 45,25 EUR
vads_ctx_mode	Adquisición de los datos en la plataforma de pago	enum	**TEST** o **PRODUCTION**
vads_currency	Código numérico de la moneda que se utilizará para el pago, según la norma ISO 4217 (código numérico)	n3	Ejemplo: 978 para el euro (EUR)
vads_page_action	Acción a realizar	enum	**PAYMENT**
vads_payment_config	Tipo de pago	enum	**SINGLE** para un pago único **MULTI** para un pago en vencimientos
vads_site_id	Identificador de la tienda	n8	Ejemplo: 12345678
vads_trans_date	Fecha y hora del formulario de pago en el huso horario UTC	n14	Respete el formato AAAAMMDDHHMMSS Ejemplo: 20200101130025
vads_trans_id	Número de la transacción. Debe ser único en un mismo día (de 00:00:00 a 23:59:59 UTC). **Atención: Este campo no distingue entre mayúsculas y minúsculas.**	an6	Ejemplo: xrT15p
vads_version	Versión del protocolo de intercambio con la plataforma de pago	enum	**V2**

- Los campos muy recomendados:
  - El medio de pago a utilizar

Nombre del campo	Descripción	Formato	Valor
vads_payment_cards	Permite forzar el tipo de tarjeta que se utilizará. Se recomienda proponer en el sitio del comerciante un botón de pago diferente para cada medio de pago. **No se recomienda dejar el campo vacío**. Consulte el capítulo *Administrar los medios de pago propuestos al comprador de la* Guía de implementación - API Formulario para más informaciones.	enum	Ejemplo: • CB • CVCONNECT • MASTERCARD • VISA • SDD

  - Los datos del pedido

Nombre del campo	Descripción	Formato	Valor
vads_order_id	Número del pedido Puede estar compuesto de letras en mayúsculas o minúsculas, dígitos o guiones ([A-Z] [a-z], 0-9, _, -).	ans..64	Ejemplo: 2-XQ001
vads_order_info	Información adicional sobre el pedido	ans..255	Ejemplo: Código intercomunicación 3125
vads_order_info2	Información adicional sobre el pedido	ans..255	Ejemplo: Sin ascensor
vads_order_info3	Información adicional sobre el pedido	ans..255	Ejemplo: Exprés
vads_ext_info_xxxx	Información complementaria necesaria al vendedor, que aparecerá en el e-mail de confirmación de pago destinado al vendedor y en el Back Office Expert (pestaña **Extra** del detalle de la transacción). **xxxx** corresponde al nombre del dato transmitido. Por ejemplo: vads_ext_info_departure_city	ans..255	Ejemplo: LHR

- Los datos del comprador

Nombre del campo	Descripción	Formato	Valor
vads_cust_email	Dirección de correo electrónico del comprador	ans..150	Ej.: abc@example.com
vads_cust_id	Referencia del comprador en el sitio web vendedor	an..63	Ejemplo: C2383333540
vads_cust_national_id	Número de identificación Tributaria	ans..255	Ejemplo: 940992310285
vads_cust_title	Estado civil del comprador	an..63	Ejemplo: M
vads_cust_status	Estado	enum	**PRIVATE**: para un particular **COMPANY**: para una empresa
vads_cust_first_name	Nombre	ans..63	Ejemplo: Laurent
vads_cust_last_name	Apellido	ans..63	Ejemplo: Durant
vads_cust_legal_name	Razón social del comprador	an..100	Ejemplo: D. & Cie
vads_cust_phone	Número de teléfono	an..32	Ejemplo: 0467330222
vads_cust_cell_phone	Número de teléfono móvil	an..32	Ejemplo: 06 12 34 56 78
vads_cust_address_number	Número de vía	ans..64	Ejemplo: 109
vads_cust_address	Dirección postal	ans..255	Ejemplo: Rue de l'innovation
vads_cust_address2	Segunda línea de dirección	ans..255	Ejemplo:
vads_cust_district	Barrio	ans..127	Ejemplo: Centre ville
vads_cust_zip	Código Postal	an..64	Ejemplo: 31670
vads_cust_city	Ciudad	an..128	Ejemplo: Labège
vads_cust_state	Estado / región	ans..127	Ejemplo: Occitanie
vads_cust_country	Código del país según ISO 3166 alpha-2	a2	Ejemplo: "FR" para Francia, "PF" para la Polinesia Francesa, "NC" para la Nueva Caledonia, "US" para Estados Unidos.

- Los campos recomendados:
  - Los datos de entrega

Nombre del campo	Descripción	Formato	Valor
vads_ship_to_city	Ciudad	an..128	Ejemplo: Bordeaux
vads_ship_to_country	Código del país según la norma ISO 3166 (obligatorio para activar una o varias acciones si el perfil **Control del país de entrega** está activado).	a2	Ejemplo: FR
vads_ship_to_district	Barrio	ans..127	Ejemplo: La Bastide
vads_ship_to_first_name	Nombre	ans..63	Ejemplo: Albert
vads_ship_to_last_name	Apellido	ans..63	Ejemplo: Durant
vads_ship_to_legal_name	Razón social	an..100	Ejemplo: D. & Cie
vads_ship_to_phone_num	Número de teléfono	ans..32	Ejemplo: 0460030288
vads_ship_to_state	Estado / región	ans..127	Ejemplo: Nouvelle aquitaine
vads_ship_to_status	Define el tipo de dirección de entrega	enum	**PRIVATE**: para entrega a un particular **COMPANY**: para entrega a una empresa
vads_ship_to_street_number	Número de vía	ans..64	Ejemplo: 2
vads_ship_to_street	Dirección postal	ans..255	Ejemplo: Rue Sainte Catherine
vads_ship_to_street2	Segunda línea de dirección	ans..255	
vads_ship_to_zip	Código Postal	an..64	Ejemplo: 33000

  - Los datos del carrito:

Nombre del campo	Descripción	Formato	Valor
vads_nb_products	Número de artículos que se encuentran en el carrito de compras	n..12	Ejemplo: 2
vads_product_ext_idN	Código de barras del producto en el sitio web vendedor. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	an..100	Ejemplo: vads_product_ext_id0 = "0123654789123654789" vads_product_ext_id1 = "0223654789123654789" vads_product_ext_id2 = "0323654789123654789"
vads_product_labelN	Descripción del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	ans..255	Ejemplo: vads_product_label0 = "tee-shirt" vads_product_label1 = "Galleta" vads_product_label2 = "sandwich"
vads_product_amountN	Precio del artículo IVA incluida. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	n..12	Ejemplo: vads_product_amount0 = "1200" vads_product_amount1 = "800" vads_product_amount2 = "950"
vads_product_typeN	Tipo del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	enum	Ejemplo: vads_product_type0 = "CLOTHING_AND_ACCESSORIES" vads_product_type1 = "FOOD_AND_GROCERY" vads_product_type2 = "FOOD_AND_GROCERY"
vads_product_refN	Referencia del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	an..64	Ejemplo: vads_product_ref0 = "CAA-25-006" vads_product_ref1 = "FAG-B5-112" vads_product_ref2 = "FAG-S9-650"
vads_product_qtyN	Cantidad del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	n..12	Ejemplo: vads_product_qty0 = "1" vads_product_qty1 = "2" vads_product_qty2 = "2"

Nota:

*Al completar el campo **vads_nb_products**, se mostrará la pestaña **Carrito** en los detalles de una transacción desde el Back Office Expert.*

*Sin embargo, si los otros campos que comienzan con **vads_product_** no se completan, la pestaña no contendrá ninguna información. Por este motivo, al completar el campo **vads_nb_products**, se vuelve obligatorio llenar los otros campos que comienzan con vads_product_. par **vads_product_**.*

- Los campos facultativos:

  Se pueden utilizar campos facultativos suplementarios.

  Consultar el capítulo **Diccionario de datos** de la guía de implementación Formulario API en nuestro sitio web para visualizar la lista de los campos disponibles

El botón **Pagar** que permitirá el envío de los datos:

```
<input type="submit" name="pagar" value="Pagar"/>
```

6. CALCULAR LA FIRMA

Para poder calcular la firma debe disponer:

- de los campos cuyos nombres comienzan con **vads_**
- del tipo de algoritmo elegido en la configuración de la tienda;
- de la **clave**.

El valor de la clave está disponible en el Back Office Expert en el menú **Configuración** > **Tienda** > pestaña **Claves**.

El tipo de algoritmo se define en su Back Office Expert en el menú **Configuración** > **Tienda** > pestaña **Configuración**.

Para máxima seguridad, se recomienda utilizar el algoritmo HMAC-SHA-256 además de una clave alfanumérica.

El algoritmo SHA-1 está en desuso, pero se mantiene por razones de compatibilidad.

Para calcular la firma:

1. Ordene los campos cuyo nombre comienza con **vads_** en orden alfabético.
2. Asegúrese de que todos los campos estén codificados en UTF-8.
3. Concatene los valores de estos campos separándolos con el carácter "**+**".
4. Concatene el resultado con la clave de prueba o de producción separándolos con el carácter "**+**".
5. De acuerdo con el algoritmo de firma definido en la configuración de su tienda:
   a. si su tienda está configurada para usar "SHA-1", aplique la función de hash **SHA-1** en el string obtenido en el paso anterior. **Depreciado.**
   b. si su tienda está configurada para usar "HMAC-SHA-256", calcule y codifique en formato Base64 la firma del mensaje usando el algoritmo **HMAC-SHA-256** con los siguientes parámetros:
      - la función hash SHA-256,
      - la clave de prueba o de producción (según el valor del campo **vads_ctx_mode**) como clave compartida,
      - el resultado del paso anterior como mensaje a autenticar.
6. Guarde el resultado del paso anterior en el campo **signature**.

Ejemplo de parámetros enviados a la plataforma de pago:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="5124" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20170129130025" />
<input type="hidden" name="vads_trans_id" value="123456" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPy9/rpesfS0="/>

<input type="submit" name="pagar" value="Pagar"/>
</form>
```

Este ejemplo de formulario se desglosa de la siguiente manera:

1. Se organizan en orden **alfabética** los campos cuyo nombre comienza con **vads_**:
   - vads_action_mode
   - vads_amount
   - vads_ctx_mode
   - vads_currency
   - vads_page_action
   - vads_payment_config
   - vads_site_id
   - vads_trans_date
   - vads_trans_id
   - vads_version
2. Se concatena el valor de estos campos con el carácter "**+**":

   ```
   INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2
   ```

3. Se agrega el valor de la clave de prueba al final del string, separado por el carácter "**+**". En este ejemplo, la clave de prueba es **1122334455667788**

   ```
   INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2+1122334455667788
   ```

4. Si usa el algoritmo SHA-1, aplíquelo al string resultante.

   El resultado a transmitir en el campo firma es: **59c96b34c74b9375c332b0b6a32e6deeec87de2b**

5. Si su tienda está configurada para usar "HMAC-SHA-256", calcule y codifique en formato Base64 la firma del mensaje usando el algoritmo **HMAC-SHA-256** con los siguientes parámetros:
   - la función hash SHA-256,
   - la clave de prueba o de producción (según el valor del campo **vads_ctx_mode**) como clave compartida,
   - el resultado del paso anterior como mensaje a autenticar.

   El resultado a transmitir en el campo firma es:

   ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPy9/rpesfS0=

6.1. Ejemplo de implementación en JAVA

Definición de una clase de utilidad Sha utilizando el algoritmo HMAC-SHA-256 para calcular la firma:

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.io.UnsupportedEncodingException;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.util.Base64;
import java.util.TreeMap;

public class VadsSignatureExample {
 /**
  * Build signature (HMAC SHA-256 version) from provided parameters and secret key.
  * Parameters are provided as a TreeMap (with sorted keys).
  */
 public static String buildSignature(TreeMap<String, String> formParameters, String
  secretKey) throws NoSuchAlgorithmException, InvalidKeyException, UnsupportedEncodingException
 {
   // Build message from parameters
   String message = String.join("+", formParameters.values());
   message += "+" + secretKey;
   // Sign
   return hmacSha256Base64(message, secretKey);
  }
    /**
    * Actual signing operation.
    */
 public static String hmacSha256Base64(String message, String secretKey) throws
  NoSuchAlgorithmException, InvalidKeyException, UnsupportedEncodingException {
   // Prepare hmac sha256 cipher algorithm with provided secretKey
   Mac hmacSha256;
   try {
    hmacSha256 = Mac.getInstance("HmacSHA256");
   } catch (NoSuchAlgorithmException nsae) {
    hmacSha256 = Mac.getInstance("HMAC-SHA-256");
   }
   SecretKeySpec secretKeySpec = new SecretKeySpec(secretKey.getBytes("UTF-8"), "HmacSHA256");
   hmacSha256.init(secretKeySpec);
   // Build and return signature
   return Base64.getEncoder().encodeToString(hmacSha256.doFinal(message.getBytes("UTF-8")));
  }
}
```

Definición de una clase de utilidad Sha utilizando el algoritmo SHA-1 para calcular la firma:

```
import java.security.MessageDigest;
import java.security.SecureRandom;

public class Sha {
    static public final String SEPARATOR = "+" ;
    public static String encode(String src) {
        try {
            MessageDigest md;
            md = MessageDigest.getInstance( "SHA-1" );
            byte bytes[] = src.getBytes( "UTF-8" );
            md.update(bytes, 0, bytes. length );
            byte[] sha1hash = md.digest();
            return convertToHex(sha1hash);
        }
        catch(Exception e){
            throw new RuntimeException(e);
        }
    }
    private static String convertToHex(byte[] sha1hash) {
        StringBuilder builder = new StringBuilder();
            for (int i = 0; i < sha1hash. length ; i++) {
               byte c = sha1hash[i];
               addHex(builder, (c >> 4) & 0xf);
               addHex(builder, c & 0xf);
        }
        return builder.toString();
    }
    private static void addHex(StringBuilder builder, int c) {
        if (c < 10)
            builder.append((char) (c + '0' ));
        else
            builder.append((char) (c + 'a' - 10));
    }
}
```

Función que calcula la firma:

```
public ActionForward performCheck(ActionMapping actionMapping, BasicForm form,
    HttpServletRequest request, HttpServletResponse response){
        SortedSet<String> vadsFields = new TreeSet<String>();
        Enumeration<String> paramNames = request.getParameterNames();

        // Recupera y ordena los nombres de los campos vads_ * en orden alfabético
        while (paramNames.hasMoreElements()) {
            String paramName = paramNames.nextElement();
            if (paramName.startsWith( "vads_" )) {
               vadsFields.add(paramName);
            }
        }
        // Calcula la firma
        String sep = Sha.SEPARATOR;
        StringBuilder sb = new StringBuilder();
        for (String vadsParamName : vadsFields) {
            String vadsParamValue = request.getParameter(vadsParamName);
            if (vadsParamValue != null) {
               sb.append(vadsParamValue);
            }
            sb.append(sep);
        }
        sb.append( shaKey );
        String c_sign = Sha.encode(sb.toString());
        return c_sign;
}
```

6.2. Ejemplo de implementación en PHP

Ejemplo de cálculo de firma utilizando el algoritmo HMAC-SHA-256:

```
function getSignature ($params,$key)
{
    /**
     *Function that computes the signature.
     * $params : table containing the fields to send in the payment form.
     * $key : TEST or PRODUCTION key
     */
    //Initialization of the variable that will contain the string to encrypt
    $signature_content = "";

    //sorting fields alphabetically
    ksort($params);
    foreach($params as $name=>$value){

        //Recovery of vads_ fields
        if (substr($name,0,5)=='vads_'){

            //Concatenation with "+"
            $signature_content .= $value."+";
         }
    }
    //Adding the key at the end
    $signature_content .= $key;

    //Encoding base64 encoded chain with SHA-256 algorithm
    $signature = base64_encode(hash_hmac('sha256',$signature_content, $key, true));
    return $signature;
 }
```

Ejemplo de cálculo de firma utilizando el algoritmo SHA-1:

```
function getSignature($params, $key)
{
    /**
     * Function that computes the signature.
     * $params : table containing the fields to send in the payment form.
     * $key : TEST or  PRODUCTION key
     */
    //Initialization of the variable that will contain the string to encrypt
    $signature_content = "" ;

    // Sorting fields alphabetically
    ksort($params);
        foreach ($params as $name =>$value)
    {
    // Recovery of vads_ fields
        if (substr($name,0,5)=='vads_') {
            // Concatenation with "+"
            $signature_content .= $value."+";
        }
    }
    // Adding the key at the end
    $signature_content .= $key;

    // Applying SHA-1 algorithm
    $signature = sha1($signature_content);
    return $signature ;
}
```

7. IMPLEMENTAR LA IPN

El script debe incluir al menos los siguientes pasos:

- Recuperar la lista de campos presentes en la respuesta enviada en POST
- Calcular la firma tomando en cuenta los datos recibidos
- Comparar la firma calculada con la recibida.
- Analizar la naturaleza de la notificación
- Recuperar el resultado del pago

El script puede, por ejemplo, probar el estado del pedido (o la información de su elección) para verificar que no se haya actualizado.

Una vez que se han completado estos pasos, el script puede actualizar la base de datos (nuevo estado del pedido, actualización del stock, registro de la información de pago, etc.).

A fin de facilitar el soporte y el diagnóstico por el vendedor en caso de error durante una notificación, se recomienda escribir mensajes que permitan conocer en qué etapa del procesamiento se produjo el error.

La plataforma lee y guarda los primeros 256 bytes del cuerpo de la respuesta HTTP.

Usted puede escribir mensajes durante todo el procesamiento. Aquí tiene un ejemplo de mensaje que puede utilizar:

Mensaje	Casos de uso
Data received	Mensaje que se mostrará durante la recuperación de los datos. Permite confirmar que el sitio del comerciante ha recibido correctamente la notificación.
POST is empty	Mensaje que se mostrará durante la recuperación de los datos. Permite indicar una eventual redirección que ha perdido los parámetros publicados por la plataforma de pago.
An error occurred while computing the signature.	Mensaje que se mostrará cuando haya fracasado la verificación de la firma.
Order successfully updated.	Mensaje que se mostrará al final del archivo una vez que sus procesamientos se hayan terminado con éxito.
An error occurred while updating the order.	Mensaje que se mostrará al final del archivo si se produjo un error durante sus procesamientos.

7.1. Preparar su entorno

Las notificaciones de tipo Llamada URL de notificación son las más importantes, pues representan el único medio confiable para que el sitio del comerciante pueda obtener el resultado de un pago.

Por lo tanto, es fundamental controlar que las notificaciones funcionen correctamente.

A continuación le presentamos algunas recomendaciones:

- Para que el diálogo entre la plataforma de pago y su sitio comerciante funcione, usted debe comprobar con sus equipos técnicos que el rango de la dirección IP **194.50.38.0/24** esté autorizada en los diferentes dispositivos de su arquitectura (firewalls, servidor apache, servidor proxy, etc.).

  Las notificaciones se envían desde una dirección IP dentro del rango 194.50.38.0/24 **en modo TEST y en modo PRODUCTION**.

- Los redireccionamientos dan como resultado la pérdida de datos en POST.

  Este caso se da si existe una configuración en sus dispositivos o en su proveedor que redirige las URL de tipo "http://**www.**example.com" vers "http://example.com" o "**http**://example.com" hacia "**https**://example.com".

- La página no debe tener una vista HTML. El acceso a recursos como imágenes o hojas de estilo ralentizan los intercambios entre la plataforma de pago y el sitio web vendedor.

- Evite las tareas que consumen tanto tiempo como generar facturas PDF o enviar e-mails en su script.

  El tiempo de procesamiento tiene un efecto directo en el plazo de la visualizacion de la página de resumen de pago.

  Cuanto mayor sea el procesamiento de la notificación, más se demora la visualización. Si el tiempo de procesamiento supera los 35 segundos, la plataforma considera que la llamada ha fallado (timeout).

- Si a su página solo se puede acceder por https, pruebe su URL en el sitio deQualys SSL Labs (*https://www.ssllabs.com/ssltest/* y modifique su configuración, si fuera necesario, a fin de obtener un grado A.

  Su certificado SSL debe firmarlo una autoridad de certificación conocida y reconocida en el mercado.

- Asegúrese de utilizar las últimas versiones del protocolo TLS a fin de mantener un alto nivel de seguridad.

7.2. Recuperar los datos devueltos en la respuesta

Los datos devueltos en la respuesta dependen de los parámetros enviados en la solicitud de pago, el tipo de pago realizado y las opciones de su tienda y del formato de la notificación.

Los datos siempre son enviados en **POST** por la plataforma de pago.

Por lo tanto, el primer paso es recuperar el contenido recibido en el modo POST.

Ejemplos:

- En PHP, los datos se almacenarán en la variable superglobal **$_POST**.
- En ASP.NET (C #), debe usar la propiedad **Form** de la clase **HttpRequest**.
- En java, debe usar el método **getParameter** de la clase **HttpServletRequest**.

La respuesta constituye una lista de campos. Cada campo contiene un valor de respuesta. La lista de campos puede cambiar.

El script tendrá que hacer un bucle para recuperar todos los campos transmitidos.

Se recomienda probar la presencia del campo **vads_hash**, presente solo durante una notificación.

```
if (empty ($_POST)){
 echo 'POST is empty';

}else{
 echo 'Data Received ';
 if (isset($_POST['vads_hash'])){

  echo 'Form API notification detected';
  //Signature computation
  //Signature verification
  //Order Update
 }
}
```

7.3. Calcular la firma de la IPN

La firma se calcula de acuerdo con la misma lógica utilizada al solicitar el pago.

Los datos transmitidos por la plataforma de pago están codificados en UTF-8. Cualquier alteración de los datos recibidos dará lugar a un cálculo de firma errónea.

Debe calcular la firma con los campos recibidos en la notificación y no con los que transmitió en la solicitud de pago.

1. Considere todos los campos cuyos nombres comienzan con **vads_**.
2. Ordene estos campos alfabéticamente.
3. Concatene los valores de estos campos separándolos con el carácter "**+**".
4. Concatene el resultado con la clave de prueba o de producción separándolos con el carácter "**+**".
5. De acuerdo con el algoritmo de firma definido en la configuración de su tienda:
   a. si su tienda está configurada para usar "SHA-1", aplique la función de hash **SHA-1** en el string obtenido en el paso anterior. **Depreciado.**
   b. si su tienda está configurada para usar "HMAC-SHA-256", calcule y codifique en formato Base64 la firma del mensaje usando el algoritmo **HMAC-SHA-256** con los siguientes parámetros:
      - la función hash SHA-256,
      - la clave de prueba o de producción (según el valor del campo **vads_ctx_mode**) como clave compartida,
      - el resultado del paso anterior como mensaje a autenticar.

Ejemplos en PHP:

```
función getSignature ($params,$key)
{
    /**
     * Función que calcula la firma.
     * $ params: matriz que contiene los campos que se enviarán en la IPN.
     * $key : clave de TEST o PRODUCTION
     */
    //Inicialización de la variable que contendrá el string a cifrar
    $contenu_signature = "";

    //Ordenar los campos alfabéticamente
    ksort($params);
    foreach($params as $nom=>$valeur){

        //Recuperación de los campos vads_
        if (substr($nom,0,5)=='vads_'){

            //Concatenación con el separador  "+"
            $contenu_signature .= $valeur."+";
         }
    }
    //Añadir la clave al final del string
    $contenu_signature .= $key;

    //Codificación base64 del string cifrada con el algoritmo HMAC-SHA-256
    $sign = base64_encode(hash_hmac('sha256',$contenu_signature, $key, true));
    return $sign;
 }
```

7.4. Comparar firmas

Para garantizar la integridad de la respuesta, debe comparar el valor de la firma contenida en la IPN con el valor calculado en el paso anterior.

No se debe comparar la firma de la IPN con la firma que transmitió en su solicitud de pago.

Si las firmas coinciden,

- luego puede considerar la respuesta como segura y proceder como resultado del análisis.
- de lo contrario, el script lanzará una excepción y advertirá al vendedor de la anomalía.

Ejemplo PHP:

```
if ($_POST['signature'] == $sign){

    //Processing data

}else{
    throw new Exception('An error occurred while computing the signature');
}
```

Las firmas no coinciden en el caso:

- error de implementación (error en su cálculo, problema de codificación UTF-8, etc.),
- un error en el valor de la clave utilizada o en el campo **vads_ctx_mode** (un problema frecuente al entrar en producción),
- intento de corromper los datos.

7.5. Analizar la naturaleza de la notificación

Durante una notificación, el campo **vads_url_check_src** permite diferenciar las notificaciones según su evento desencadenante:

- creación de una transacción.
- Referencia de la notificación en el Back Office Expert por el vendedor.

Especifica la regla de notificación aplicada:

Valor	Regla aplicada
PAY	El valor PAY se envía en los siguientes casos: • pago inmediato (pago al contado o primer vencimiento de un pago en vencimientos) • pago diferido dentro de 7 días solo si el vendedor ha configurado la regla de **URL de notificación al final del pago**. • pago abandonado o cancelado por el comprador. solo si el vendedor ha configurado la regla **URL de notificación al abandonar (comprador)**.
BO	Ejecución de la notificación desde el Back Office Expert (haga clic con el botón derecho en una transacción > **Ejecutar la URL de notificación**).
BATCH	El valor BATCH se envía al actualizar el estado de una transacción tras la sincronización con el adquirente. Este es el caso de los pagos redirigidos al adquirente. Solo si el vendedor ha configurado la regla **URL de notificación al modificar por batch**.
BATCH_AUTO	El valor BATCH_AUTO se envía en los siguientes casos: • pago diferido a más de 7 días • vencimientos para un pago en vencimientos (excepto el primero). solo si el vendedor ha configurado la regla **URL de notificación al autorizar por batch**. La notificación se enviará cuando se solicite autorización para un pago con el estado "autorización pendiente".
REC	El valor REC solo se enviará para los pagos por suscripción si el vendedor ha establecido la regla **URL de notificación para crear un pago recurrente**.
MERCH_BO	El valor MERCH_BO se enviará: • durante una operación realizada desde el Back Office Expert (cancelación, reembolso, modificación, validación, duplicación, creación o actualización de token), si el vendedor ha configurado la regla de notificación: **URL de notificación al modificar una transacción en el Back Office (vendedor)**
RETRY	Repetición automática de la URL de notificación.

Tabla 1: Valores asociados al campo vads_url_check_src

Al probar su valor, el script puede realizar un procesamiento diferente según la naturaleza de la notificación.

Por ejemplo:

Si **vads_url_check_src** tiene asignado el valor **PAY** o **BATCH_AUTO** entonces el script actualizará el estado del pedido, ...

Si **vads_url_check_src** tiene asignado el valor **REC** entonces el script recuperará la referencia de recurrencia e incrementará el número de vencimientos vencidas en caso de pago aceptado...

7.6. Tratamiento de los datos de la respuesta

A continuación hay un ejemplo de análisis para guiarle paso a paso durante el tratamiento de los datos de la respuesta.

1. Identifique el modo (TEST o PRODUCTION) en el que fue creada la transacción analizando el valor del campo **vads_ctx_mode.**

2. Identifique el pedido tomando el valor del campo **vads_order_id** si lo indicó en el formulario de pago.
   Verifique que el estado de la orden no haya sido actualizado.

3. Tome el resultado del pago indicado en el campo **vads_trans_status**.
   Su valor le permite definir el estado de la orden.

Valor	Descripción
ABANDONED	**Abandonado** Pago abandonado por el comprador. La transacción no se creó y **por lo tanto no es visible en el Back Office Expert.**
ACCEPTED	**Aceptado**. Estado de una transacción de tipo VERIFICATION cuya autorización o solicitud de información ha sido aceptada. Este estado no puede cambiar. Las transacciones con estado "**ACCEPTED**" no se capturan nunca.
AUTHORISED	**En espera de captura** La transacción es aceptada y será remitida al banco automáticamente en la fecha prevista.
AUTHORISED_TO_VALIDATE	**Por validar** La transacción, creada en validación manual, está autorizada. El vendedor debe validar manualmente la transacción para que sea capturada al banco. La transacción puede ser validada siempre y cuando la fecha de expiración de la solicitud de autorización no se haya vencida. Si esta fecha se supera, el pago toma el estado **EXPIRED**. El estado **Vencido** es definitivo.
CANCELLED	**Anulado** La transacción es anulada por el vendedor.
CAPTURED	**Capturada** La transacción se ha remitido al banco.
CAPTURE_FAILED	**La remesa de la transacción ha fallado.** Contacte al Soporte.
EXPIRED	**Vencido** Este estado interviene en el ciclo de vida de un pago con captura diferida. La fecha de caducidad de la solicitud de autorización se alcanzó y el vendedor no validó la transacción. No se realizará el débito al portador.
REFUSED	**Rechazado** La transacción se ha rechazado.
SUSPENDED	**Suspendido** La captura de la transacción está bloqueada temporalmente por el adquirente (AMEX GLOBAL o SECURE TRADING). Una vez que la captura se procesa correctamente, el estado de la transacción se **CAPTURED**.
UNDER_VERIFICATION	**Comprobación en curso** En espera de la respuesta del adquirente. Este estado es temporal.

Valor	Descripción
	Para transacciones CB o PPRO, este estado indica que se ha solicitado un reembolso. Los controles están en curso para validar el reembolso. Se enviará una notificación al sitio del comerciante para advertirle sobre el cambio de estado. Requiere la activación de la regla de notificación URL de notificación al modificar por batch.
WAITING_AUTHORISATION	**En espera de autorización** El plazo de captura al banco es superior a la duración de validez de la autorización.
WAITING_AUTHORISATION_TO_VALIDATE	**Para validar y autorizar** El plazo de captura al banco es superior a la duración de validez de la autorización. Se aceptó una autorización 1 EUR (o solicitud de información en la red de CB si el adquirente lo admite). El vendedor debe validar manualmente la transacción para que se realice la orden de autorización y la captura.

4. Analice el campo **vads_occurrence_type** para determinar si se trata de un pago unitario o de un pago que forma parte de una serie (suscripción o pago en N veces).

Valor	Descripción
UNITAIRE	Pago unitario (pago al contado).
RECURRENT_INITIAL	Primer pago de una serie.
RECURRENT_INTERMEDIAIRE	Enésimo pago de una serie.
RECURRENT_FINAL	último pago de una serie.

5. Analice el campo **vads_payment_config** para determinar si se trata de un **pago en N veces**.

Nombre del campo	Valor para un pago al contado	Valor para un pago en varias veces
vads_payment_config	SINGLE	MULTI (donde la sintaxis exacta es MULTI:first=X;count=Y;period=Z)

Si se trata de un pago en N veces, identifique el número del vencimiento tomando el valor del campo **vads_sequence_number**.

Atención: con la aplicación del Soft Decline, el campo **vads_sequence_number** ya no permite identificar con facilidad el primer pago de un pago en N veces. El primer pago que puede tomar un número de secuencia diferente 1, el número de secuencia del segundo pago no será forzosamente 2.

6. Tome el valor del campo **vads_trans_date** para identificar la fecha de pago.

7. Analice el campo **vads_payment_option_code** para determinar si se trata de un pago en varias cuotas:

Valor	Descripción
1	Pago en 1 cuota
2	Pago en 2 cuotas
3	Pago en 3 cuotas
n	Pago en n cuotas

8. Tome el valor del campo **vads_capture_delay** para identificar el número de días antes de la entrega al banco.

   Esto le permitirá identificar si se trata de un pago inmediato o diferido.

9. Tome el monto y la moneda utilizada. Para esto, tome los valores de los siguientes campos:

Nombre del campo	Descripción
vads_amount	Monto del pago en su unidad monetaria más pequeña.
vads_currency	Código de la moneda utilizada para el pago.
vads_change_rate	Tasa de cambio utilizada para calcular el monto real del pago (vervads_effective_amount).
vads_effective_amount	Monto del pago en la moneda realmente utilizada para efectuar el depósito.
vads_effective_currency	Moneda en la que será efectuado el depósito.

10. Tome el valor del campo **vads_auth_result** para conocer el resultado de la solicitud de autorización:

La lista completa de los códigos enviados se puede consultar en el diccionario de datos.

Para ayudarle a entender el motivo del rechazo, a continuación está una lista de los códigos frecuentemente devueltos:

Valor	Descripción
03	**Aceptador invalido** Este código es emitido por el adquirente. Corresponde a un problema de configuración en los servidores de autorización. (p. ej., contrato cerrado, código MCC declarado incorrecto, etc.). **Para conocer la razón precisa del rechazo, el vendedor debe contactar a su banco.**
05	**No honrar** Este código es emitido por el banco emisor de la tarjeta. Es utilizado en los siguientes casos: • fecha de expiración no válida, • código de seguridad CVV no válido, • crédito sobrepasado, • saldo insuficiente (etc.) **Para conocer la razón precisa del rechazo, el comprador debe contactar a su banco.**
51	**Saldo insuficiente o crédito rebasado** Este código es emitido por el banco emisor de la tarjeta. Puede ser obtenido si el comprador no cuenta con un saldo suficiente para realizar su compra. **Para conocer la razón precisa del rechazo, el comprador debe contactar a su banco.**
56	**Tarjeta ausente del archivo** Este código es emitido por el banco emisor de la tarjeta. El número de la tarjeta ingresado es erróneo o el número de tarjeta y la fecha de expiración no existen.
57	**Transacción no permitida a este portador** Este código es emitido por el banco emisor de la tarjeta. Es utilizado en los siguientes casos: • el comprador trata de realizar un pago por internet con una tarjeta de retiro, • el límite autorizado de la tarjeta se superó. **Para conocer la razón precisa del rechazo, el comprador debe contactar a su banco.**
59	**Sospecha de fraude** Este código es emitido por el banco emisor de la tarjeta. Puede ser enviado después de ingresar varias veces el CVV o la fecha de expiración erróneos. **Para conocer la razón precisa del rechazo, el comprador debe contactar a su banco.**
60	**El aceptador de la tarjeta debe contactar al adquirente** Este código es emitido por el adquirente. Corresponde a un problema de configuración en los servidores de autorización. Se utiliza cuando el contrato del vendedor no corresponde al canal de venta utilizado. (p. ej., una transacción de comercio electrónico con un contrato VAD-de ingreso manual). **Contacte al servicio al cliente para regularizar la situación.**
81	**El emisor no acepta pagos sin autenticación Safekey** Este código es emitido por el banco emisor de la tarjeta. Al recibir este código, la plataforma de pago realiza automáticamente un nuevo intento de pago con autenticación 3D Secure cuando esto es posible.

11. Tome el resultado de la autenticación del titular. Para esto:

a. Tome el valor del campo **vads_threeds_enrolled** para determinar el estado de la inscripción de la tarjeta.

Valor	Descripción
Vacío	Proceso 3DS no realizado (3DS desactivado en la solicitud, vendedor no inscrito o medio de pago no elegible para 3DS).
Y	Autenticación disponible, portador inscrito.
N	Titular no enrolado.
U	Imposible identificar al portador o tarjeta no elegible para las tentativas de autenticación (p. ej., tarjetas comerciales o prepagadas).

b. Tome el resultado de la autenticación del titular recuperando el valor del campo **vads_threeds_status**.

Valor	Descripción
Vacío	Autenticación 3DS no realizada (3DS desactivado en la solicitud, vendedor no inscrito o medio de pago no elegible para 3DS).
Y	Portador autenticado correctamente.
N	Error de autenticación del portador.
U	Autenticación imposible.
A	Tentativa de autenticación, pero no se realizó la autenticación.

12. Tome el resultado de los controles asociados con el fraude identificando el valor del campo **vads_risk_control**. Este campo es enviado únicamente si el vendedor:

- se suscribe al servicio "**Ayuda con la decisión**"
- activó al menos un control desde su Back Office Expert (menú **Configuración** > **Control de riesgos**).

Esto toma como valor una lista de valores separados por “;” cuya sintaxis es: **vads_risk_control = control1=result1;control2=result2**

Los valores posibles para **control** son:

Valor	Descripción
CARD_FRAUD	Control de la presencia del número de la tarjeta del comprador en la lista negra de tarjetas.
SUSPECT_COUNTRY	Controla la presencia del país emisor de la tarjeta del comprador dentro de la lista de países prohibidos.
IP_FRAUD	Controlar la presencia de la dirección IP del comprador en la lista negra de IP.
CREDIT_LIMIT	Control de la frecuencia y el monto de compras de un mismo número de tarjeta, o del monto máximo de un pedido.
BIN_FRAUD	Controla la presencia del código BIN de la tarjeta dentro de la lista gris de códigos BIN.
ECB	Controla si la tarjeta del comprador es de tipo débito.
COMMERCIAL_CARD	Controla si la tarjeta del comprador es una tarjeta comercial.
SYSTEMATIC_AUTO	Controla si la tarjeta del comprador es una tarjeta con autorización sistemática.
INCONSISTENT_COUNTRIES	Controla si el país de la dirección IP, el país emisor de la tarjeta de pago y el país de la dirección del comprador son coherentes entre ellos.
NON_WARRANTY_PAYMENT	Transferencia de responsabilidad.
SUSPECT_IP_COUNTRY	Controla la presencia del país del comprador, identificado mediante su dirección IP, dentro de la lista de países prohibidos.

Los valores posibles para **result** son:

Valor	Descripción
OK	OK.
WARNING	Control informativo no exitoso.
ERROR	Control de bloqueo no exitoso.

13. Tome el tipo de tarjeta utilizada para el pago.

Se pueden presentar dos casos:

- Para un pago realizado con **una sola tarjeta**. Los campos correspondientes son los siguientes:

Nombre del campo	Descripción
vads_card_brand	Marque la tarjeta utilizada para el pago. P. ej.: CB, VISA, VISA_ELECTRON, MASTERCARD, MAESTRO, VPAY
vads_card_number	Número de la tarjeta utilizada para realizar el pago.
vads_expiry_month	Mes de expiración entre 1 y 12 (p. ej.: 3 para marzo, 10 para octubre).
vads_expiry_year	Año de expiración de 4 cifras (p. ej.: 2023).
vads_bank_code	Código del banco emisor
vads_bank_label	Apellido del banco emisor
vads_bank_product	Código de producto de la tarjeta
vads_card_country	Código de país del país de emisión de la tarjeta (Código alfa ISO 3166-2 ej.:"FR" para Francia, "PF" para la Polinesia Francesa, "NC" para la Nueva Caledonia, "US" para Estados Unidos.).

- Para un **pago fraccionado** (es decir, una transacción que utiliza varios medios de pago), los campos correspondientes son los siguientes:

Nombre del campo	Valor	Descripción
vads_card_brand	MULTI	Se utilizan varios tipos de tarjeta para el pago.
vads_payment_seq	En formato json, ver detalles a continuación.	Detalles de las transacciones realizadas.

El campo **vads_payment_seq** (formato json) describe la secuencia de pago fraccionado. Contiene los elementos:

1. "trans_id": identificador de la transacción global en la secuencia de pago.
2. "transaction": cuadro de las transacciones de la secuencia. Los elementos que lo componen son los siguientes:

Nombre del parámetro	Descripción
amount	Monto de la secuencia de pago.
operation_type	Operación de débito.
auth_number	Número de autorización. No se devolverá si no se aplica al medio de pago en cuestión. Ejemplo: 949478
auth_result	Código de retorno de la solicitud de autorización.
capture_delay	Plazo antes de la captura (en días). • Para un pago con tarjeta bancaria, el valor de este parámetro tiene en cuenta el plazo en número de días antes del captura. Si este parámetro no es transmitido en el formulario de pago, se utilizará el valor predeterminado definido en Back Office Expert.
card_brand	Medio de pago utilizado. Para un pago con tarjeta bancaria (por ejemplo CB o tarjetas CB de marca compartida Visa o Mastercard), este parámetro tiene el valor **"CB"**. Consultar la guía de integración del formulario de pago disponible en nuestro sitio de documentación para visualizar la lista completa de los tipos de tarjeta.
card_number	Número del medio de pago.
expiry_month	Mes de caducidad del medio de pago.
expiry_year	Año de caducidad del medio de pago.
payment_certificate	Certificado de pago.
contract_used	Afiliación utilizada para el pago.
identifier	Identificador único (token/alias) asociado a un medio de pago.
identifier_status	Presente solo si la acción solicitada es la creación o actualización de un alias. Valores posibles:

<table>
<tr><th>Nombre del parámetro</th><th>Descripción</th></tr>
<tr><td></td><td>
<table>
<tr><th>Valor</th><th>Descripción</th></tr>
<tr><td>CREATED</td><td>La solicitud de autorización fue aceptada.
El token (o RUM para un pago SEPA) se crea con éxito.</td></tr>
<tr><td>NOT_CREATED</td><td>La solicitud de autorización fue denegada.
El token (o RUM para un pago SEPA) no se crea y no aparecerá en el Back Office Expert.</td></tr>
<tr><td>UPDATED</td><td>El token (o RUM para un pago SEPA) se actualiza con éxito.</td></tr>
<tr><td>NOT_UPDATED</td><td>El token (o RUM para un pago SEPA) no se ha actualizado.</td></tr>
<tr><td>ABANDONED</td><td>Acción abandonada por el comprador (deudor).
El token (o RUM para un pago SEPA) no se crea y no aparecerá en el Back Office Expert.</td></tr>
</table>
</td></tr>
<tr><td>presentation_date</td><td>Para un pago con tarjeta bancaria, este parámetro corresponde a la fecha de captura deseada (en el formato ISO 8601).</td></tr>
<tr><td>trans_id</td><td>Número de transacción.</td></tr>
<tr><td>ext_trans_id</td><td>Parámetro ausente para el pago con tarjeta bancaria.</td></tr>
<tr><td>trans_uuid</td><td>Referencia única generada por la plataforma de pago después de la creación de una transacción de pago.
Ofrece una garantía de singularidad para cada transacción.</td></tr>
<tr><td>extra_result</td><td>Código numérico del resultado de los controles de riesgo.
<table>
<tr><th>Code</th><th>Descripción</th></tr>
<tr><td>Vacío</td><td>No se hace ningún control.</td></tr>
<tr><td>00</td><td>Todos los controles fueron exitosos.</td></tr>
<tr><td>02</td><td>La tarjeta ha superado el saldo autorizado.</td></tr>
<tr><td>03</td><td>La tarjeta pertenece a la lista gris del vendedor.</td></tr>
<tr><td>04</td><td>El país de emisión de la tarjeta pertenece a la lista gris del vendedor.</td></tr>
<tr><td>05</td><td>La dirección IP pertenece a la lista gris del vendedor.</td></tr>
<tr><td>06</td><td>El código bin pertenece a la lista gris del vendedor.</td></tr>
<tr><td>07</td><td>Detección de una tarjeta electrónica de crédito.</td></tr>
<tr><td>08</td><td>Detección de una tarjeta comercial nacional.</td></tr>
<tr><td>09</td><td>Detección de una tarjeta comercial extranjera.</td></tr>
<tr><td>14</td><td>Detección de una tarjeta con autorización sistemática.</td></tr>
<tr><td>20</td><td>Verificación de consistencia: no hay coincidencias de país (IP del país, mapa del país, país del comprador).</td></tr>
<tr><td>30</td><td>El país de la dirección IP pertenece a la lista gris.</td></tr>
<tr><td>99</td><td>Problema técnico encontrado por el servidor al procesar uno de los controles locales.</td></tr>
</table>
</td></tr>
<tr><td>sequence_number</td><td>Número de secuencia.</td></tr>
<tr><td>trans_status</td><td>Estado de la transacción.</td></tr>
</table>

Las transacciones anuladas también se encuentran en el cuadro.

14. Registre el valor del campo **vads_trans_uuid**. Este le permitirá identificar de manera única la transacción si utiliza la API Web Services.

15. Tome toda la información sobre el detalle del pedido, el detalle del comprador y el detalle de la entrega.

    Estos datos solo están presentes en la respuesta si se enviaron en el formulario de pago.

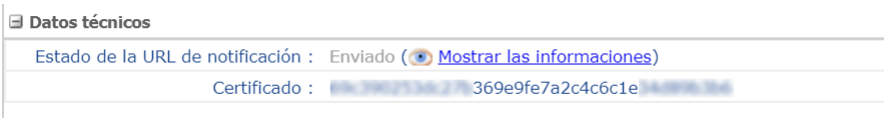
    Sus valores son iguales a los enviados en el formulario.

16. Proceda a actualizar el pedido.

7.7. Test y troubleshooting

Para probar las notificaciones, siga las siguientes etapas:

1. Realice un pago (en modo TEST o en modo PRODUCTION).
2. Una vez finalizado el pago, busque la transacción en su Back Office (Menú **Gestión** > **Transacciones** o **Transacciones de TEST** si realizó el pago en modo TEST).
3. Haga doble clic en la transacción para ver el **detalle de la transacción**.
4. En el detalle de la transacción, busque la sección **Datos técnicos**.
5. Compruebe el estado de la URL de notificación:



La lista de los estados posibles se presenta a continuación:

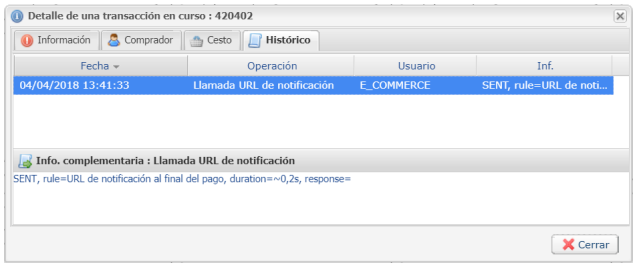
Estado	Descripción
N/A	La transacción no dio lugar a una notificación o no se activó ninguna regla de notificación.
URL no definido	Un evento activó la regla de notificación de fin de pago, pero la URL no está configurada.
Llamada en curso	La notificación está en curso Este estado es temporal.
Enviado	La notificación se ha enviado correctamente y un equipo distante respondió con un código HTTP 200, 201, 202, 203, 204, 205 ou 206.
Enviado (redirección permanente)	El sitio del comerciante ha devuelto un código HTTP 301 o 308 con una nueva URL para contactar. Una nueva llamada en modo POST se realiza hacia la nueva URL.
Enviado (redirección temporal)	El sitio del comerciante ha devuelto un código HTTP 302 o 307 con una nueva URL para contactar. Una nueva llamada en modo POST se realiza hacia la nueva URL.
Enviado (redirección a otra página)	El sitio del comerciante ha devuelto un código HTTP 301 con una nueva URL para contactar. Una nueva llamada en modo GET se realiza hacia la nueva URL.
Fallido	Error genérico diferente de los códigos descritos a continuación.
Servidor inalcanzable	La notificación duró más de 35 s.
Error con SSL handshake	La configuración de su servidor no es correcta. Realice un diagnóstico en el sitio de Qualys (*https://www.ssllabs.com/ssltest/*) y corrija los errores.
Conexión interrumpida	Error de comunicación.
Conexión rechazada	Error de comunicación.
Error servidor 300	Caso de redirección no aceptado por la plataforma.
Error servidor 304	Caso de redirección no aceptado por la plataforma.
Error servidor 305	Caso de redirección no aceptado por la plataforma.
Error servidor 400	El sitio del vendedor ha devuelto un código HTTP 400 Bad Request.
Error servidor 401	El sitio del vendedor ha devuelto un código HTTP 401 Unauthorized. Asegúrese de que el recurso no esté protegido por un archivo .htaccess.
Error servidor 402	El sitio del vendedor ha devuelto un código HTTP 402 Payment Required.
Error servidor 403	El sitio del vendedor ha devuelto un código HTTP 403 Forbidden. Asegúrese de que el recurso no esté protegido por un archivo .htaccess.
Error servidor 404	El sitio del vendedor ha devuelto un código HTTP 404 Not Found. Verifique que el ingreso de la URL esté correcto en la configuración de la regla. También verifique que el archivo esté presente en su servidor.
Error servidor 405	El sitio del vendedor ha devuelto un código HTTP 405 Method Not allowed.
Error servidor 406	El sitio del vendedor ha devuelto un código HTTP 406 Not Acceptable.
Error servidor 407	El sitio del vendedor ha devuelto un código HTTP 407 Proxy Authentication Required.
Error servidor 408	El sitio del vendedor ha devuelto un código HTTP 408 Request Time-out.

Estado	Descripción
Error servidor 409	El sitio del vendedor ha devuelto un código HTTP 409 Conflict.
Error servidor 410	El sitio del vendedor ha devuelto un código HTTP 410 Gone.
Error servidor 411	El sitio del vendedor ha devuelto un código HTTP 411 Length Required.
Error servidor 412	El sitio del vendedor ha devuelto un código HTTP 412 Precondition Failed.
Error servidor 413	El sitio del vendedor ha devuelto un código HTTP 413 Request Entity Too Large.
Error servidor 414	El sitio del vendedor ha devuelto un código HTTP 414 Request-URI Too long.
Error servidor 415	El sitio del vendedor ha devuelto un código HTTP 415 Unsupported Media Type.
Error servidor 416	El sitio del vendedor ha devuelto un código HTTP 416 Requested range unsatisfiable.
Error servidor 417	El sitio del vendedor ha devuelto un código HTTP 417 Expectation failed.
Error servidor 419	El sitio del vendedor ha devuelto un código HTTP 419 Authentication Timeout.
Error servidor 421	El sitio del vendedor ha devuelto un código HTTP 421 Misdirected Request.
Error servidor 422	El sitio del vendedor ha devuelto un código HTTP 422 Unprocessable Entity.
Error servidor 423	El sitio del vendedor ha devuelto un código HTTP 423 Locked.
Error servidor 424	El sitio del vendedor ha devuelto un código HTTP 424 Failed Dependency.
Error servidor 425	El sitio del vendedor ha devuelto un código HTTP 425 Too Early.
Error servidor 426	El sitio del vendedor ha devuelto un código HTTP 426 Upgrade Required.
Error servidor 429	El sitio del vendedor ha devuelto un código HTTP 431 Request Header Fields Too Large.
Error servidor 431	El sitio del vendedor ha devuelto un código HTTP 415 Unsupported Media Type.
Error servidor 451	El sitio del vendedor ha devuelto un código HTTP 451 Unavailable For Legal Reasons.
Error servidor 500	El sitio del vendedor ha devuelto un código HTTP 500 Internal Server Error. Se ha producido un error aplicativo en el servidor de su tienda. Consulte los registros de su servidor HTTP (generalmente apache). El problema solo puede corregirse al intervenir en su servidor.
Error servidor 501	El sitio del vendedor ha devuelto un código HTTP 501 Not Implemented.
Error servidor 502	El sitio del vendedor ha devuelto un código HTTP 502 Bad Gateway / Proxy Error.
Error servidor 503	El sitio del vendedor ha devuelto un código HTTP 503 Service Unavailable.
Error servidor 504	El sitio del vendedor ha devuelto un código HTTP 504 Gateway Time-out. El servidor del vendedor no ha aceptado la llamada dentro del tiempo de espera establecido de 10 s.
Error servidor 505	El sitio del vendedor ha devuelto un código HTTP 505 HTTP Version not supported.

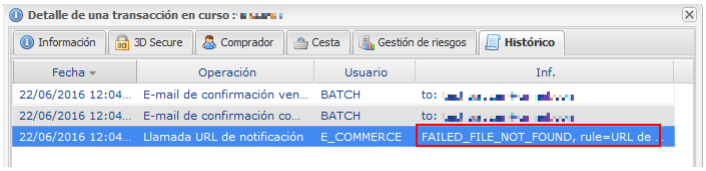
Para obtener más información sobre una notificación, haga clic en el enlace **Mostrar la información** o haga clic en la pestaña **Historial** y busque la línea **Llamada URL de notificación**.

Para ayudar al vendedor a identificar el origen del error, la plataforma analiza sistemáticamente los primeros 512 caracteres que devuelve el sitio del comerciante y los muestra en la columna **Información**.

- Ejemplo de notificación procesada con éxito:



- Ejemplo de notificación incorrecta



Si la plataforma no logra conectarse a la URL de su página, se enviará un e-mail de alerta a la dirección especificada.

Este contiene:

- El código HTTP del error encontrado
- Elementos de análisis en función del error
- Sus consecuencias
- El procedimiento a seguir desde el Back Office Expert para reenviar la solicitud a la URL definida en la configuración de la regla.

8. PROCESAR EL REGRESO A LA TIENDA

De forma predeterminada, cuando el comprador vuelve al sitio web vendedor, su navegador no transmite ningún parámetro.

No obstante, si el campo **vads_return_mode** se ha transmitido en el formulario de pago (véase el capítulo **Gestionar el retorno al sitio web vendedor** de la guía de implementación Formulario API disponible en nuestro sitio web) será posible recuperar los datos:

- ya sea en GET: datos presentes en la url en la forma: ?param1=valeur1&param2=valeur2.
- en POST: datos enviados en un formulario POST.

Los datos transmitidos al navegador son los mismos que en las notificaciones (IPN).

Solo los campos **vads_url_check_src** y **vads_hash** solo se enviarán en la notificación instantánea.

Puede consultar el capítulo **Análisis de los resultados de pago** para analizar estos datos.

Nota: El retorno a la tienda solo debe permitirle mostrar un contexto visual al comprador. No utilice los datos recibidos para realizar el procesamiento de la base de datos.

