

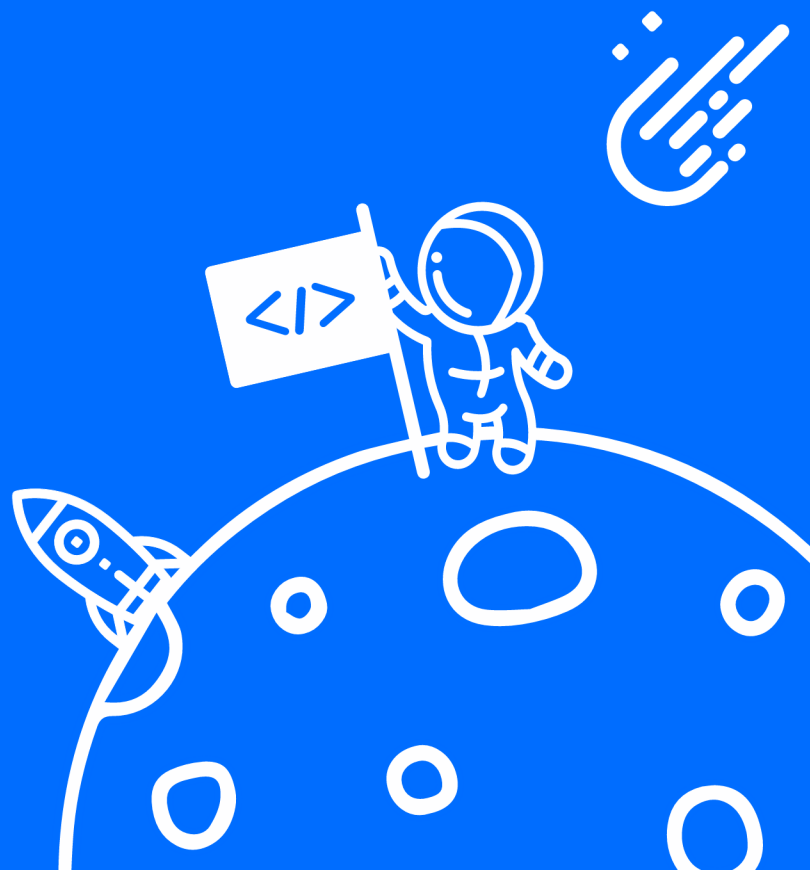
PAYCOMET

Proxy Wubook v1.3

24-02-2020

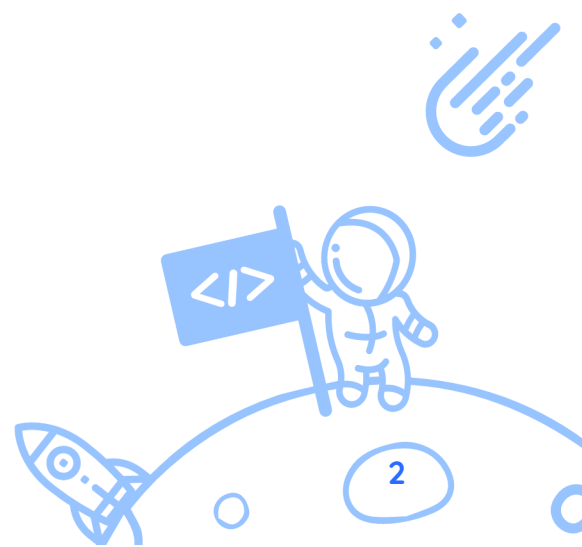
<https://www.paycomet.com>
info@paycomet.com

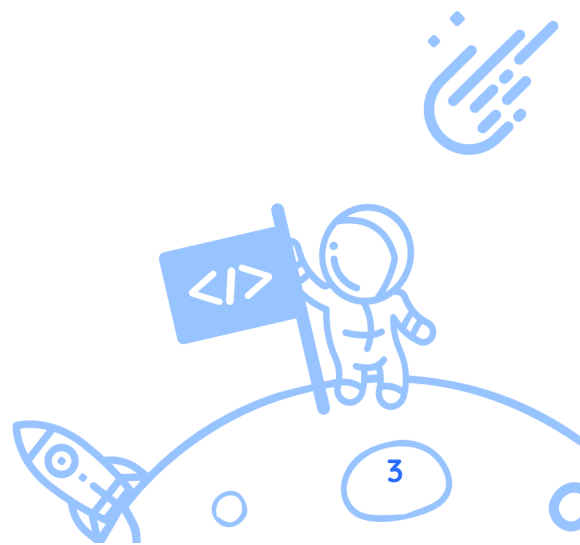
Paycomet © 2020



Control de versiones

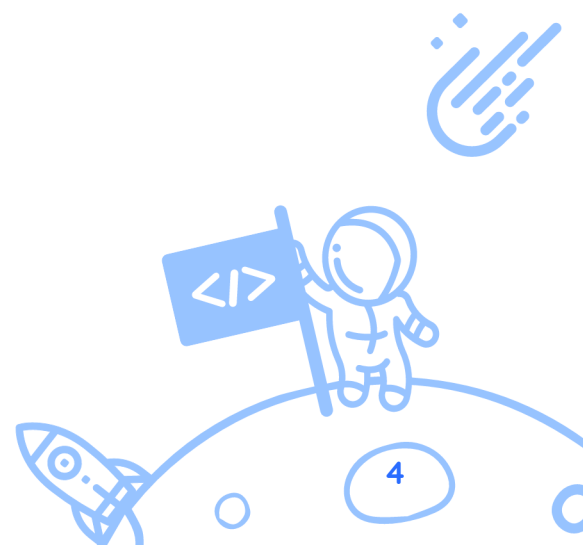
Autor	Fecha	Versión	Modificaciones
W.E.D	02/10/2017	V 1.0	Versión inicial
Juan Sánchez	12/03/2018	V 1.1	Se permite tokenizar tarjetas sin cvc.
Jose Ramón García	26/06/2019	V 1.2	Cambio Marca y Firma por SHA512
Diego Marcos	24/02/2020	V 1.3	Cambio EndPoints y API-KEYS





Contenido

Control de versiones	2
Introducción	5
Tipos de peticiones.....	5
Tokenización de la información de tarjeta de una reserva	7
Transformación de los datos de tarjeta	7
Peticiones acquire_token y release_token	9
Endpoint y datos de la llamada	10
Respuesta error.....	11
Envío de API-KEYS.....	13
Entorno de desarrollo.....	14
Contacto	14



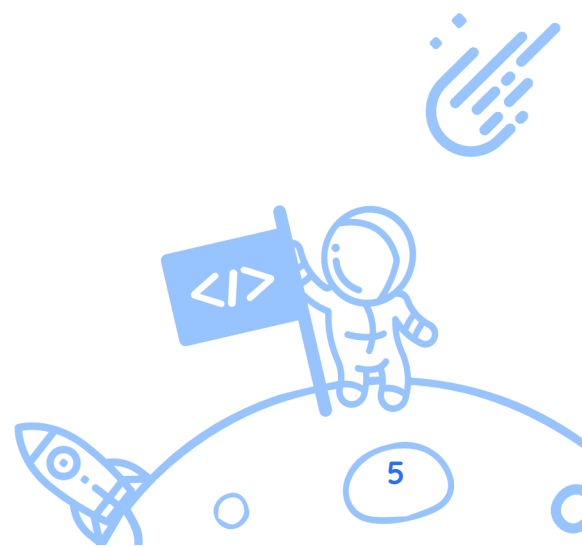
Introducción

El servicio Proxy de PAYCOMET con WuBook permite tokenizar las tarjetas registradas en las reservas hoteleras y verificar que los formatos de los datos de tarjeta recibidos en las mismas son correctos. Para poder acceder a este tipo de integración, se deberá contar previamente con la integración de la API de WuBook (<http://tdocs.wubook.net/wired.html>), en la que se envían los datos identificadores de la tarjeta incluida en la reserva seleccionada y se devuelve la información de tarjeta modificada y tokenizada.

Tipos de peticiones

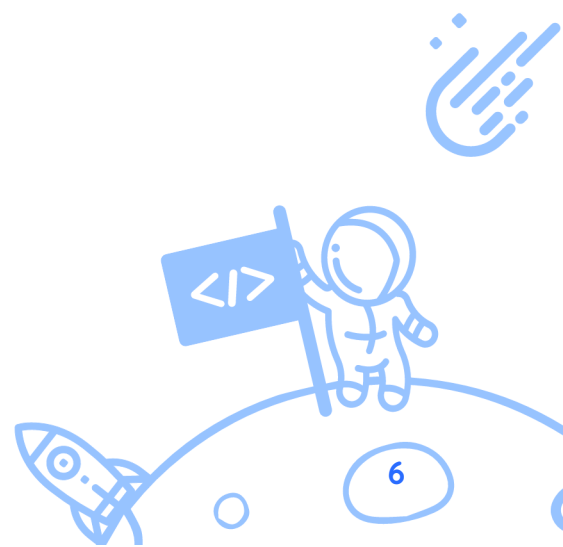
Los tipos de peticiones que se tratan en esta integración son aquellas que pueden recuperarse mediante la llamada al método `fetch_cc` (`token`, `building_id`, `booking_code`, `building_password`).

Como es preceptivo, también las propias para obtención / liberación de token previo / posterior, métodos `acquire_token` y `release_token`.



Ejemplo de información contenida en una petición fetch_cc

```
<?xml version='1.0'?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>cc_type</name>
            <value>
              <int>1</int>
            </value>
          </member>
          <member>
            <name>cc_number</name>
            <value>
              <string>4111111111111111</string>
            </value>
          </member>
          <member>
            <name>cc_cvv</name>
            <value>
              <string>123</string>
            </value>
          </member>
          <member>
            <name>cc_code</name>
            <value>
              <nil/>
            </value>
          </member>
          <member>
            <name>cc_owner</name>
            <value>
              <string>John Doe </string>
            </value>
          </member>
          <member>
            <name>cc_expiring</name>
            <value>
              <string>01/2020</string>
            </value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodResponse>
```



Tokenización de la información de tarjeta de una reserva

Al recibir una respuesta de petición `fetch_cc`, si esta contiene datos de tarjeta (es preciso que están presentes `cc_number`, `cc_expiring`, `cc_cvv`), se intenta realizar una tokenización de los mismos. Si todos los datos necesarios están presentes y se consigue realizar correctamente, se añadirán los elementos

- `id_user`
- `token_user`

que representan la tarjeta tokenizada y que serán necesarios para operaciones posteriores. A partir de la versión 1.1 se permite tokenizar aunque no llegue cvc. Por favor, solícitanos la habilitación de esta opción.

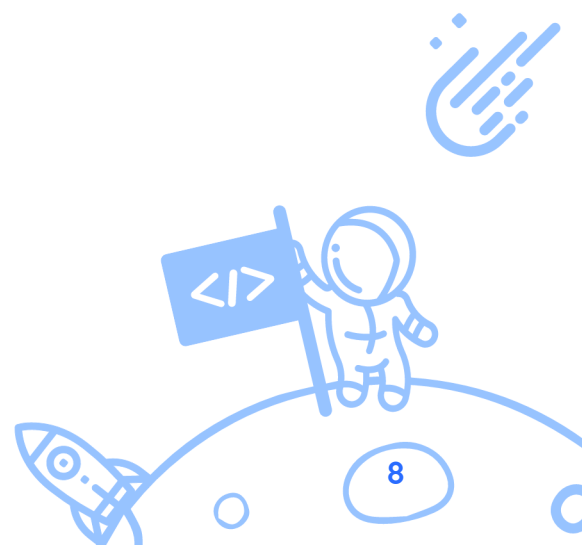
Transformación de los datos de tarjeta

En la respuesta devuelta, la tarjeta aparecerá enmascarada, `411111*****1111`. Le fecha de caducidad se devolverá tal y como aparece en la respuesta a la petición `fetch_cc` y el código CVC2 se eliminará y no llegará. Por lo tanto, la reserva anterior, una vez tokenizada la información de la tarjeta, quedará de la siguiente manera:

```
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
          <member>
            <name>cc_type</name>
            <value>
              <int>1</int>
            </value>
          </member>
          <member>
            <name>cc_number</name>
            <value>
              <string>411111*****1111</string>
            </value>
          </member>
          <member>
            <name>cc_cvv</name>
```



```
        <value>
            <nil/>
        </value>
    </member>
    <member>
        <name>cc_code</name>
        <value>
            <nil/>
        </value>
    </member>
    <member>
        <name>cc_owner</name>
        <value>
            <string>John Doe</string>
        </value>
    </member>
    <member>
        <name>cc_expiring</name>
        <value>
            <string>01/2020</string>
        </value>
    </member>
    <member>
        <name>id_user</name>
        <value>
            <int>99999999</int>
        </value>
    </member>
    <member>
        <name>token_user</name>
        <value>
            <string>WWXXwwXWWXxWWXxWWxxWW</string>
        </value>
    </member>
</struct>
</value>
</param>
</params>
</methodResponse>
```

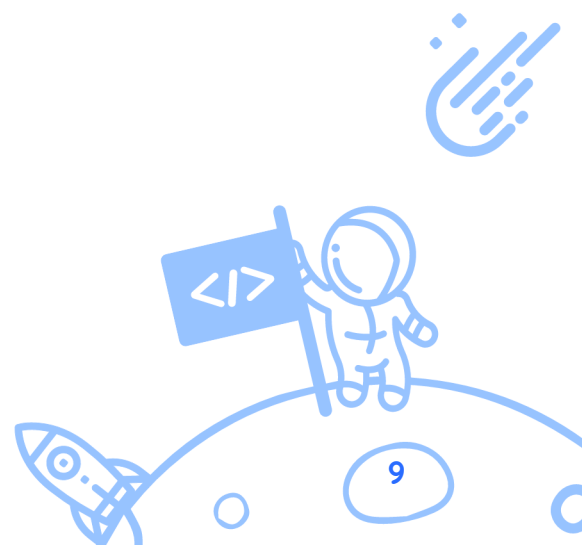


En caso de error en la tokenización, no aparecerán los elementos iduser / tokenuser y se añadirá un elemento ds_error_id

```
<struct>
  ...
  <member>
    <name>ds_error_id</name>
    <value>
      <int>1195</int>
    </value>
  </member>
  ...
</struct>
```

Peticiones acquire_token y release_token

Estas peticiones son necesarias en base a la integración WuBook y el proxy las permite ambas. Se especifican las tres llamadas permitidas en el punto siguiente.



Endpoint y datos de la llamada

El script al que hay que realizar la petición es:

- <https://eclipse.paycomet.com/api/v1/wubook>

El tipo de petición que espera es un post con un parámetro 'xml' con la siguiente estructura:

```
<root>
  <booking>
    <xml>
      <request>
        <method>nombre_metodo</method>
        <params>

1.- acquire_token
        <user>usuario</user>
        <password>password</password>
        <provider_key>provider_key</provider_key>

2.- fetch_cc
        <token>token</token>
        <building_id>building_id</building_id>
        <booking_code>booking_code</booking_code>
        <building_password>building_password</building_password>

3.- release_token
        <token>token</token>

        </params>
      </request>
    </xml>
    <url>URL Wubook a la que se debe realizar la llamada</url>
  </booking>
  <paytpv>
    <ds_merchant_code>código de comercio en PAYTPV</ds_merchant_code>
    <ds_merchant_terminal>número de terminal en PAYTPV</ds_merchant_terminal>
  </paytpv>
```

```
</root>
```

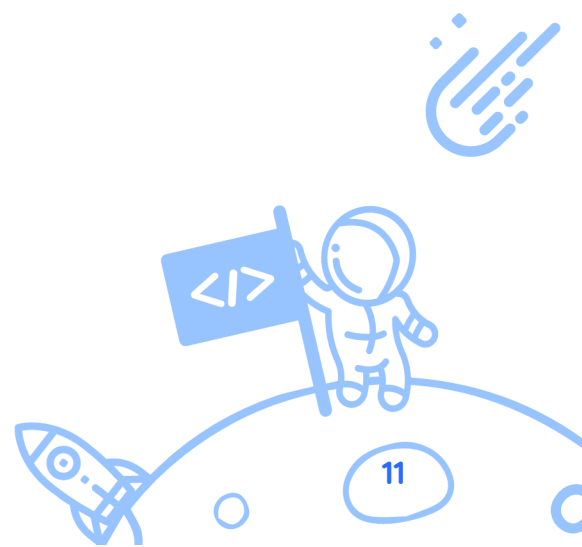
Si se quiere realizar la llamada `acquire_token`, debe indicarse en el elemento `<method>` y enviar los parámetros/elementos necesarios (`user`, `password`, `provider_key`).

Similar en los demás métodos. Para `fetch_cc` indicadlo en el elemento `<method>` y enviad `token`, `building_id`, `booking_code` y `building_password`.

Para liberar el token, indicad `release_token` en el elemento `method` y enviar su valor en el elemento `token`.

Respuesta error

Si se produce un error en los datos de la petición o al acceder a WuBook, se enviará un mensaje de error igual al que devuelve WuBook actualmente, con la consideración siguiente: si el error se produce en la llamada a PAYCOMET (credenciales, formato) o en la llamada desde PAYCOMET a WuBook (timeout, sin respuesta, etc) el identificador del error será el de PAYCOMET y la descripción del error será la de PAYCOMET indicando que no ha habido establecimiento de conexión con WuBook. Una vez superado ese punto, si se produce un error se devolverá exactamente lo devuelto por WuBook:

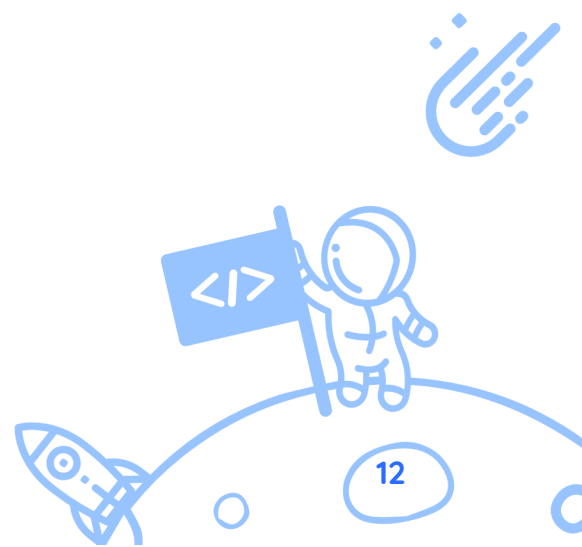


Error PAYCOMET anterior a la llamada a Wubook

```
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <int>1121</int>
            </value>
            <value>
              <string>Client not found</string>
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```

Error devuelto por WuBook

```
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value>
        <array>
          <data>
            <value>
              <int>-2</int>
            </value>
            <value>
              <string>Invalid Token</string>
            </value>
          </data>
        </array>
      </value>
    </param>
  </params>
</methodResponse>
```



Los códigos de error PAYCOMET pueden consultarse en esta url:
<https://docs.paycomet.com/es/documentacion/codigos-de-error>

Envío de API-KEYS

En el nuevo Endpoint el método de autenticación ha sido actualizado de una restricción por IP al envío una cabecera específica con el API-KEY generado para el producto/productos en cuestión.

La cabecera HTTP debe tener el siguiente nombre:

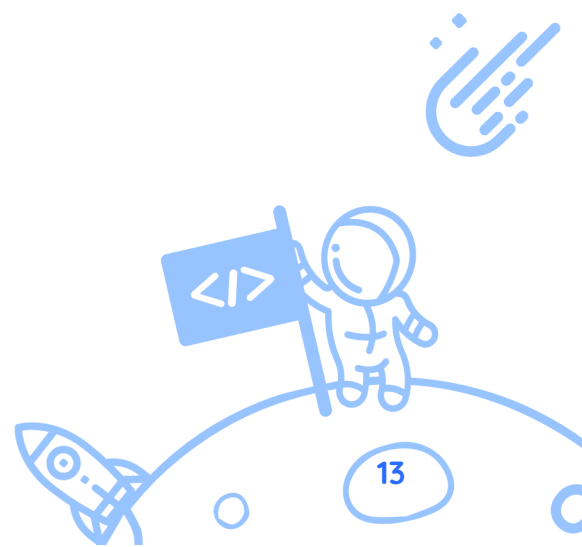
PAYCOMET-API-TOKEN

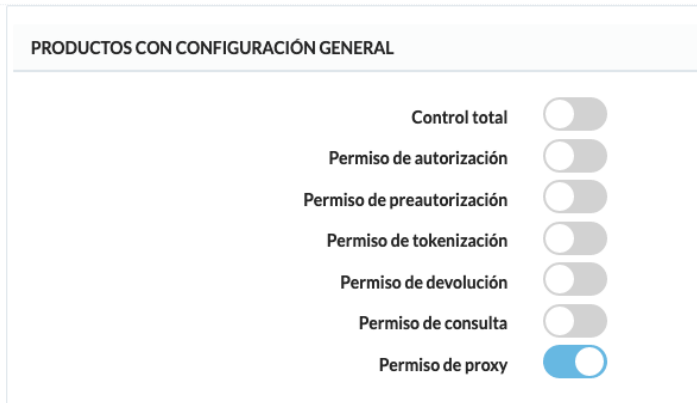
El valor de la cabecera debe ser la clave API-KEY generada a través del panel de control de cliente: "Mis productos" -> "API Keys".

Una vez en el panel de gestión de API Keys debe crearse una nueva "API KEY" en el botón superior derecho y asignarle un nombre.

En el siguiente panel debe copiarse la API KEY ya que solo aparecerá en una ocasión. Por motivos de seguridad, no volverá a mostrarse.

Los permisos de las API KEYS pueden asignarse a uno o varios productos, e incluso a todos los de una cuenta mediante los selectores:





Para el producto que vaya a utilizarse para presentar las reservas de la OTA (Online Travel Agency) y donde quedarán tokenizadas las tarjetas será necesario asignarle permisos específicos de "**Permiso de proxy**".

Una vez finalizadas las gestiones de API Keys, el valor generado a la API Key deberá enviarse en la cabecera como método de autenticación.

Entorno de desarrollo

Este desarrollo no cuenta con un sandbox en el que puedan realizarse llamadas con datos de prueba. Las llamadas al script son todas reales. Lo que debe hacerse para poder probar la integración es crear en WuBook hoteles y reservas de prueba que permitan verificar el proceso completo.

Contacto

Para ampliar información o resolver cualquier duda acerca de esta integración, puedes ponerte en contacto con nosotros en esta dirección de correo tecnico@paycomet.com

