hola@efevoopay.com

Manual de Integración

EfevooPay

Versión 1.02 Julio 2023

ÍNDICE

INTRODUCCIÓN

Las APIS de EfevooPay te permiten hacer cobros en tu plataforma de Ecommerce de tus productos y/o servicios. Mediante las APIS es muy sencillo conectarse con nuestra plataforma, generar y cobrar órdenes utilizando cualquier lenguaje de programación.

Nuestra API REST cuenta con URLs independientes para cada función. Estas llamadas esperan una solicitud en formato JSON y responden los datos correspondientes en este mismo formato.

Existe un tipo de integración para poder implementar el procesador de pagos:

EfevooPay Checkout: Ventana externa hosteada en EfevooPay.com donde el cliente final llena sus datos personales, de tarjeta y el monto a pagar.

Información General

- Es necesario utilizar los métodos de encriptación definidos más adelante para enviar peticiones al API.
- La API Key, API Secret y TOTP Secret se obtienen del dashboard de EfevooPay Agregador (https://admin.efevoopay.com) en la sección de Integraciones.
- La API de EfevooPay solamente responderá a comunicaciones seguras hechas sobre https. Las solicitudes de http regresaran mensajes de error.
- Las respuestas a las solicitudes arrojaran el resultado en formato JSON.
- Las llaves de prueba las pueden solicitar a soporte EfevooPay para realizar pruebas en ambiente testnet .

Acuerdo de desarrollador

Al utilizar las APIs de EfevooPay estás de acuerdo con los términos y condiciones de EfevooPay.com

AUTENTICACIÓN

Las APIs de EfevooPay utilizan llaves para autenticar el acceso y es necesario utilizar un HASH de autorización dentro del json de acuerdo a la funcionalidad requerida.

Llaves API

Las llaves se obtienen en el dashboard de EfevooPay en la sección de Integraciones.

Todos los comercios dentro de EfevooPay pueden crear una serie de llaves de acceso API/WSS.

- **API Key:** Llave principal de acceso al API del comercio.
- **API Secret:** Llave secreta para la encriptación de los mensajes a firmar.
- **API Totp:** Llave para generar un código basado en tiempo.

REST: TOTP: Valor en el mensaje.

El valor del TOTP es una “Time-based One Time Password”, una contraseña de 6 caracteres de un único uso basada en el tiempo, cambiando cada 30 segundos y único para cada cliente de EfevooPay. Toda petición deberá contener el parámetro de "totp" en la concatenación del mensaje. Este valor es calculado usando la llave TOTP obtenida desde la plataforma de EfevooPay.

REST: Authorization HASH.

El usuario deberá crear un HASH de autorización (firma) generado al encriptar el body del mensaje (en cada petición) usando el algoritmo HMAC-SHA256.

La llave para generar la autorización (firma) : APISECRET que se obtiene desde la plataforma EfevooPay.

Al generar la firma asegúrese de que el body se utilice como un String de caracteres, ejemplo:

body = API_Key + TOTP

body = "ABCDEFG123456" + "000000"

Nota: Evite usar separadores de línea.

Una vez obtenida la concatenación del mensaje, se realizará la encriptación del ApiSecret y el mensaje, dicho se deberá integrar como **token** dentro del json de acuerdo a la funcionalidad requerida.

ENCRIPTACIONES

Para hacer las llamadas a la API de EfevooPay es necesario encriptar el mensaje utilizando HMAC-SHA256. Te presentamos la forma de encriptar los mensajes utilizando diferentes lenguajes de programación.

Python

```
import hmac
import hashlib

string_body = "apiKey":"ABCDEFG123456" + "totp":"000000"
apiSecret = "7ab134dfe1cb4c60a72abba2e0cec88178a466b2622e45159a1233b3f917dd4c"

AuthorizationHash =
hmac.new(apiSecret.encode('utf8'),string_body.encode('utf8'),hashlib.sha256).hexdige
st() print(AuthorizationHash) #
eebb4cc712fd1fe9ac0ae6613b91a1dba2a048c77.....
```

Ruby

```
require 'openssl'

string_body = "apiKey":"ABCDEFG123456" + "totp":"000000"
apiSecret = "7ab134dfe1cb4c60a72abba2e0cec88178a466b2622e45159a1233b3f917dd4c"
AuthorizationHash = OpenSSL::HMAC.hexdigest("SHA256", apiSecret, string_body)

puts AuthorizationHash #eebb4cc712fd1fe9ac0ae6613b91a1dba2a048c77.....
```

Php

```
<?php
$string_body = "apiKey":"ABCDEFG123456" + "totp":"000000";
$apiSecret = "7ab134dfe1cb4c60a72abba2e0cec88178a466b2622e45159a1233b3f917dd4c";

$AuthorizationHash = hash_hmac('sha256',$string_body,$apiSecret,false);
echo $AuthorizationHash; //eebb4cc712fd1fe9ac0ae6613b91a1dba2a048c77.....
?>
```

Nodejs

```
import { createHmac } from 'crypto';

const string_body = "apiKey":"ABCDEFG123456" + "totp":"000000"
 const apiSecret    = "7ab134dfe1cb4c60a72abba2e0cec88178a466b2622e45159a1233b3f917dd4c"
const AuthorizationHash = createHmac('sha256',
apiSecret).update(string_body).digest('hex'); console.log(AuthorizationHash);
//eebb4cc712fd1fe9ac0ae6613b91a1dba2a048c77…….
```

C#

```
using
System;
using
System.IO;
using
System.Security.Cryptography;
using System.Text;

class AuthorizationHash
{
   static void Main(string[] args)
   {
      string string_body = "apiKey":"ABCDEFG123456" + "totp":"000000”;
      string apiSecret = "7ab134dfe1cb4c60a72abba2e0cec88178a466b2622e45159a1233b3f917dd4c";
      //
      HMACSHA256 hmac = new
      HMACSHA256(Encoding.UTF8.GetBytes(apiSecret)); Byte[] hash =
      hmac.ComputeHash(Encoding.UTF8.GetBytes(string_body)); string
      authorizationHash = BitConverter.ToString(hash).Replace("-",
string.Empty).ToLower();
      Console.WriteLine(authorizationHash); //eebb4cc712fd1fe9ac0ae6613b91a1dba2a048c77..
   }
}
```

ESTRUCTURA MENSAJES

La estructura general de los mensajes será la siguiente:

```
{
        "group": "{{grupo de funciones}}",
        "api_key": "{{apiKey}}",
        "token" : "{{token encriptado}}",
        "method" : "{{metodo/evento}}",
        "data"{
                "website" : "https://comercio-registrado.com",
                "data1" : "-",
                "data2" : "-"
        }
}
```

Todos los mensajes de respuesta del API y del WSS tendrán la siguiente estructura.

```
{
        "status": {
                "code": "0",
                "description": "Message ok."
        },
        "payload": {}
}
```

El sistema de EfevooPay siempre regresará como status['code'] el valor de "0" cuando los mensajes sean correctos y contengan datos dentro del payload. En caso de que el sistema de su comercio detecte un valor diferente a "0", podrá procesar el error.

DEFINICIÓN DE LLAMADAS

Eventos WSS.

- sales_monitor.
  Notificación de todo los eventos de venta que se realicen a través de la plataforma ecommerce.

- session_monitor.
  Monitoreo del estado de la venta a través de EfevooPay.

API.

- /sessionToken.
  Solicitud de un nuevo token para venta en EfevooPay.

- /createOrder
  Crea una orden nueva y genera el URL del checkout en ventana externa.

EFEVOOPAY REST APIS.

Los comercios electrónicos podrán utilizar el formato y branding deseado para hacer su carrito de compras y pagar mediante la plataforma de EfevooPay utilizando las siguientes llamadas REST API.

Paso 1. Create Token.

En esta llamada se obtiene el TOKEN único de la orden a crear.

Listado de Variables

Variable	Descripción	Formato	¿Es Requerido?
group	Consulta el grupo de funciones.	String	Si
method	Método que se consultará.	String	Si
token	Token obtenido de la encriptación.	String	Si
apikey	Llave unica de identificación	String	Si
web_site	Ruta del comercio que se registró.	String	Si

Enlaces

Producción: https://ecommapi.efevoopay.com/ecommerce
Prueba: https://wmx.ecommapi.testflight-efevoopay.com/ecommerce

Ejemplo.

BODY

```
{
        "group": "wmx_api",
        "method": "get_token",
        "token": “{{generated_token}}”,
        "api_key": “{{apikey}}”,
        "data": {
                "web_site": “{{comercio_registrado}}”
        }
}
```

Nota: Evitar usar espacios o separadores de línea en los mensajes a enviar.

RESPUESTA EXITOSA.

```
{
        "status": {
                "code": "0",
                "description": "Message ok."
        },
        "payload": [
                {
                   "token":  "auJ0eXAiOiJKV1.QiLCJhbGciOT8xcqgtUclOfinthechat.PrnRUxqg7FCjYpk6NI",
                }
        ]
}
```

RESPUESTA NO EXITOSA.

```
{
        "status": {
                "code": "99999",
                "description": "ERROR: Mensaje descriptivo de error"
        },
        "payload": { }
}
```

Paso 2. getCheckOut

Se obtiene la ruta para realizar el checkout :

Producción: https://efevoopay.com/CheckOut?ApiKey={api_key}&Token={token}

Test : https://wmx.testflight-efevoopay.com/CheckOut?ApiKey={api_key}&Token={token}

ERRORES.

Status code	Descripción	Solución
7001	Esta compra no es de el comercio especificado	Por favor revisar que la url parámetros y web_site es el comercio que se registro.
7002	Esta compra ya ha sido cancelada.	Por favor revise que es una compra sin cancelar.
7005	No fue posible generar la transacción correctamente.	Por favor revisar el body del request.
7006	No se encontraron datos del carro de compras.	Por favor revisar el body del request de la información enviada.
7007	No se encontró precio total en el carro.	Por favor revisar el body del request los parámetros enviados.

VERSIONES.

Versión	Fecha de elaboración.	Descripción de cambios
1.01	05/04/2023	Versión Inicial
1.02	26/07/2023	Cambio de formato y estructura del documento