D:\Proyectos\pasarela-pago\images\bg-head.pngD:\Proyectos\pasarela-pago\images\bg-head.png

## Transacciones seguras por Internet en el Perú

D:\Proyectos\pasarela-pago\images\bg-head.png

**Documento de Integración técnico**

Empresa Editora El Comercio

Jr. Miro Quesada 300 – Lima 1

311 6500

pagoefectivo.pe

**Especificación técnica**

## Configuración

## Para que un Comercio o Servicio se pueda integrarse es necesario que este se encuentre registrado en Pago Efectivo. Una vez registrado se le proporcionará al servicio los códigos y clave que lo identificaran como Comercio registrado, además de 2 claves de encriptación en archivos con extensión “1pz” que se usará para asegurar la seguridad en la transmisión de las tramas.

## Claves de encriptación

## PagoEfectivo para la seguridad de los mensajes usa un proceso de encriptación basado en RSA lo que lleva a las entidades a usar las claves de encriptación, que por entidad serán 2 (una pública y otra privada).

## Clave privada de la entidad

## Esta clave es una clave única por entidad, se usara en todos las operaciones. Sin esta clave no se podrán ejecutar las operaciones de PagoEfectivo.

## Esta clave es usada para desencriptar tramas que han sido encriptados por su par clave (su clave pública), además se usa para generar una firma la cual servirá para validar la autenticidad de la entidad.

## Clave pública de PagoEfectivo

## Esta es una clave común de PagoEfectivo, se usara para la generación de Solicitud de Pago, para consulta de Solicitud de Pago.

## Esta clave es usada para encriptar tramas que solo serán interpretadas usando la clave privada de PagoEfectivo.

## Configuración de parámetros

## Se requiere configurar los siguientes valores en el config de la aplicación que usara el API (lib/configpe.php)

## Incluir el código de servicio de 3 caracteres que se les proporcionará

## define('MERCHAN\_ID', 'PRU');

## colocar las urls de los servicios web:

## define('SERVER', 'http://pre.pagoefectivo.pe/');

## WEBSERVICE DE ENCRIPTACION DE DATOS

## define('WSCRYPTA', SERVER.'PagoEfectivoWSCrypto/WSCrypto.asmx?WSDL');

## WEBSERVICE DE GENERACION DE CIP

Define('WSCIP2', SERVER.'PagoEfectivoWSGeneralv2/service.asmx?WSDL');

## VENTANA DE PAGOEFECTIVO PARA MOSTRAR EL CIP

## define('PE\_WSGENPAGO', PE\_SERVER.'GenPago.aspx');

## define('PE\_WSGENPAGOIFRAME', PE\_SERVER.'GenPagoIF.aspx');

## Colocar los nombres de los archivos que contienen las llaves proporcionadas

## //Llave publica de Pago Efectivo

## define('PUBLICKEY', 'SPE\_PublicKey.1pz');

## //LLave privada del comercio

## define('PRIVATEKEY', 'PRU\_PrivateKey.1pz');

## Nota: Los nombre de los archivos .1pz son referenciales, deberán ser iguales a los ubicados físicamente es lib/key/

## //Selección de modalidad – para recuperar el cip y el portal lo muestre

## define('MOD\_INTEGRACION',3); //3 -> Iframe 2 -> Redireccion a PagoEfectivo

## Uso de claves

## En las operaciones se requiere usar las claves de encriptación de la entidad que se está integrando con PagoEfectivo, previamente han sido generadas y entregadas por PagoEfectivo, en los 3 procesos se requerirá usar una o dos claves dependiendo de la operación:

## Generar Solicitud de Pago (clave privada, clave pública).

## Consultar Solicitud de Pago (clave privada, clave pública).

## Desencriptar texto (clave privada).

## Se recomienda asignar una ruta segura para estas claves.

## Funciones de consumo del API

## A continuación la especificación de los métodos:

## solicitarPago

## Nos permite generar un CIP, el método recibe como entrada un objeto BEWSGenCIPRequestMod1 (posteriormente se especificara la estructura de este objeto), a continuación un ejemplo del código en php:

## //generar número de orden dinámico por no contar con BD

## $num = rand(1,100).date('dmYHis'); //para el comercio será el número de su orden

## $xml = new App\_Service\_PagoEfectivo\_Solicitud();

## $xml->addContenido(array(

## 'IdMoneda' => 1,

## 'Total' =>22.00,

## 'MetodosPago' => '1,4',

## 'CodServicio' => 'PRU',

## 'Codtransaccion' => $num,

## 'EmailComercio' => 'alex.castillo@perucomsultores.pe',

## 'FechaAExpirar' => '31/12/2012 17:00:00',

## 'UsuarioId' => '001',

## 'DataAdicional' => '',

## 'UsuarioNombre' => 'aaaaa',

## 'UsuarioApellidos' => 'bbbbb',

## 'UsuarioLocalidad' => 'LIMA',

## 'UsuarioProvincia' => 'LIMA',

## 'UsuarioPais' => 'PERU',

## 'UsuarioAlias' => 'Alex',

## 'UsuarioTipoDoc' => 'DNI',

## 'UsuarioNumeroDoc' => 1,

## 'UsuarioEmail' => 'alex.castillo@perucomsultores.pe',

## 'ConceptoPago' => 'Pago'

## ))->addDetalle(array(array(

## 'Cod\_Origen' => 'CT',

## 'TipoOrigen' => 'TO',

## 'ConceptoPago' => 'Transaccion comision 1',

## 'Importe' => 22.00

## 

## ));

## $pagoefectivo = new App\_Service\_PagoEfectivo();

## //Creación de la solicitud

## $paymentRequest = $pagoefectivo->solicitarPago($xml);

## //Obtención del valor del Cip

## Echo 'CIP: '. $paymentRequest->NumeroOrdenPago;

Dentro del objeto que responde el Servicio a esta solicitud se encuentra un campo XML, donde contiene información de la solicitud, uno de las propiedades de las solicitud es el campo token con el cual podrá generar el CIP posteriormente.

## Descripción de objetos

**BEWSGenCIPRequestMod1:**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo de dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| cServ | Cadena | SI | Código del servicio asignado.  Ejemplo: Neoauto: NEO, Urbania: URB, Kotear: KOT |
| CClave | Cadena | SI | Es la trama firmada con la llave privada –proporcionada por PagoEfectivo- a través de un WebService que se describe más adelantes. |
| Xml | Cadena | SI | Es la trama en formato XML que contiene todos los datos relevantes de la operación y debe ser encriptado a través de otro WebService que se describe también más adelante.  Ejemplo de texto:  <?xml version="1.0" encoding="utf-8" ?>  <SolPago>  <IdMoneda>1</IdMoneda>  <Total>22.00</Total>  <MetodosPago>1,2</MetodosPago>  <CodServicio>PRU</CodServicio>  <Codtransaccion>0000040</Codtransaccion>  <EmailComercio>admin@pagoefectivo.pe</EmailComercio>  <FechaAExpirar>31/12/2016 17:00:00</FechaAExpirar>  <UsuarioId>001</UsuarioId>  <DataAdicional></DataAdicional>  <UsuarioNombre>Victor</UsuarioNombre>  <UsuarioApellidos>Avila</UsuarioApellidos>  <UsuarioLocalidad>LIMA</UsuarioLocalidad>  <UsuarioProvincia>LIMA</UsuarioProvincia>  <UsuarioPais>PERU</UsuarioPais>  <UsuarioAlias>Victor</UsuarioAlias>  <UsuarioTipoDoc>1</UsuarioTipoDoc>  <UsuarioNumeroDoc>1</UsuarioNumeroDoc>  <UsuarioEmail>alex.castillo@perucomsultores.pe</UsuarioEmail>  <ConceptoPago>Orden 456 en Comercio Prueba</ConceptoPago>  <Detalles>  <Detalle>  <ConceptoPago> Orden 456 en Comercio Prueba </ConceptoPago>  <Importe>22.00</Importe>  </Detalle>  </Detalles>  </SolPago>  **Nota:** Las fechas tienen el formato es DIA/MES/AÑO HORA:MINUTO:SEGUNDO |

**Xml:**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo de dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdMoneda | Cadena | SI | Código de la moneda:  1: Soles y 2: Dólares |
| Total | Cadena de 21 caracteres, 18 enteros, 2 decimales y separados por una coma | SI | Monto total del pago. |
| MetodosPago | Cadena | SI | Códigos de Pago, pueden ser:  1: Bancos/FullCarga  2: Cuenta Virtual (Monedero Virtual)  En el caso sea más de un método de pago se enviará separado por comas, ejemplo:  1,2 |
| CodServicio | Cadena | SI | Código de servicio en referencia.  Ejemplo: Neoauto: NEO, Urbania: URB, Kotear: KOT |
| Codtransaccion | Cadena | SI | Código de Referencia al Pago |
| EMailComercio | Cadena | NO | Es el correo electrónico del representante del comercio o servicio. Este dato es referencial. |
| FechaAExpirar | Cadena | SI | La fecha y hora en el que el CIP expirará.  Formato: DIA/MES/AÑO HORA:MINUTO:SEGUNDO |
| UsuarioID | Cadena | SI | Código de usuario de referencia |
| DataAdicional | Cadena | NO | Información de referencia |
| UsuarioNombre | Cadena | SI | Nombre del usuario de referencia de la generación del CIP |
| UsuarioApellidos | Cadena | SI | Apellidos del usuario de referencia de la generación del CIP |
| UsuarioLocalidad | Cadena | NO | Nombre de la Localidad del usuario. (dato de referencia) |
| UsuarioProvincia | Cadena | NO | Nombre de la Provincia del usuario. (dato de referencia) |
| UsuarioPais | Cadena | NO | Nombre del País del usuario. (dato de referencia) |
| UsuarioAlias | Cadena | NO | Alias del usuario (dato de referencia) |
| UsuarioTipoDoc | Cadena | NO | Tipo de Documento:  DNI, LE, RUC, etc. |
| UsuarioNumeroDoc | Cadena | NO | Número del Documento |
| UsuarioEmail | Cadena | NO | Email del usuario (dato de referencia) |
| ConceptoPago | Cadena | SI | Descripción del Concepto de Pago |
| (\*)Detalles | XML |  | Contiene los detalles del pago asociado, especificado más abajo. |
| (\*\*)ParamsURL | XML |  | Contiene los parámetros de la Url de notificación, especificado más abajo. |
| (\*\*\*)ParamsEmail | XML |  | Contiene los parámetros que se enviarán por email al usuario, especificado más abajo. |

**(\*)Detalles:**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo de dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| ConceptoPago | Cadena | SI | Descripción del Concepto de Pago |
| Importe | Cadena de 21 caracteres, 18 enteros, 2 decimales y separados por un punto | SI | Monto del pago del detalle. |
|  |  |  |  |

**BEWSGenCIPResponseMod1:**

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| Estado | Cadena de 1 carácter | Representa el resultado de la operación:  “1”: Se llegó a realizar la solicitud  “0”: No se llegó a realizar la solicitud  “-1”: Se dio una excepción no controlada |
| Mensaje | Cadena | Contiene el mensaje resultado de haber realizado la operación:  En el caso del Estado “1”:   * “Se ha Generado la Solicitud de Pago: 35345.”   En el caso del Estado “0”:  “El Código de Identificación de Pago (CIP) ya se ha generado anteriormente para esta operación. CIP 000000001”   * “El Código de Identificación de Transaccion ya se ha generado anteriormente para esta operación. Código Transaccion: 1232424   En el caso del Estado “-1”:   * “Ha ocurrido un Error no Manejado” |
| Xml | Cadena | Texto encriptado con la estructura de un Xml que retorna información del cip.  Ejemplo:  **<?xml version="1.0" encoding="utf-8" ?><ConfirSolPago><idResSolPago>1628</idResSolPago><CodTrans>30020143</CodTrans><Token>f6b98a03-1c63-4de5-9903-b5ca51bdafcc</Token><MetodoPago>1,2</MetodoPago><Codigo></Codigo><Fecha>2013-06-17 17:13:15</Fecha><Estado>592</Estado><ParamsURL><ParamURL><Nombre>IDCliente</Nombre><Valor>15474</Valor></ParamURL><ParamURL><Nombre>FechaHoraRegistro</Nombre><Valor>19/08/2010</Valor></ParamURL></ParamsURL><CIP><IdOrdenPago>1040719</IdOrdenPago><IdCliente>466</IdCliente><IdEstado>22</IdEstado><IdServicio>49</IdServicio><IdMoneda>1</IdMoneda><NumeroOrdenPago>00000001040719</NumeroOrdenPago><OrderIdComercio>30020143</OrderIdComercio><Total>22.00</Total><MerchantID></MerchantID><FechaEmision>2013-06-17 17:13:15</FechaEmision><FechaCancelado></FechaCancelado><FechaConciliado></FechaConciliado><FechaAnulado></FechaAnulado><FechaExpirado></FechaExpirado><FechaEliminado></FechaEliminado><TiempoExpiracion>31031.779166666667</TiempoExpiracion><EstaConciliado></EstaConciliado><MailComercio></MailComercio><UsuarioID>135</UsuarioID><DataAdicional></DataAdicional><UsuarioNombre>Victor</UsuarioNombre><UsuarioApellidos>Avila</UsuarioApellidos><UsuarioAlias>Victor</UsuarioAlias><UsuarioEmail>victoravila578@gmail.com</UsuarioEmail><UsuarioDomicilio></UsuarioDomicilio><UsuarioLocalidad>LIMA</UsuarioLocalidad><UsuarioProvincia>LIMA</UsuarioProvincia><UsuarioPais>PERU</UsuarioPais><Detalles><Detalle><IdDetalleOrdenPago>46230</IdDetalleOrdenPago><ConceptoPago>Transaccion Comision 1</ConceptoPago><Importe>10.00</Importe><Tipo\_Origen>TO</Tipo\_Origen><Cod\_Origen>CT</Cod\_Origen><Campo1></Campo1><Campo2></Campo2><Campo3></Campo3></Detalle><Detalle><IdDetalleOrdenPago>46231</IdDetalleOrdenPago><ConceptoPago>Transaccion Comision 2</ConceptoPago><Importe>12.00</Importe><Tipo\_Origen>TO</Tipo\_Origen><Cod\_Origen>CT</Cod\_Origen><Campo1></Campo1><Campo2></Campo2><Campo3></Campo3></Detalle></Detalles></CIP><Trans></Trans></ConfirSolPago>** |

**Xml:**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo Dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| idResSolPago | Cadena | SI | Identificador de la solicitud de pago creada, este identificador se usara posteriormente para consultar el estado. |
| CodTrans | Cadena | SI | Número de Código de Referencia al Pago. |
| Token | Cadena | SI | Código Único generado para la operación que se usará posteriormente para la generación de CIP cuando se haga un re direccionamiento a la página de “Pago Efectivo”. |
| Fecha | Cadena | SI | Campo la fecha en que realizo la operación, tiene el formato es DIA/MES/ANIO HORA:MINUTO:SEGUNDO |
|  |  |  |  |
| MetodoPago | Cadena | SI | Códigos de Pago, pueden ser:  1: Bancos/FullCarga  2: Cuenta Virtual (Monedero Virtual)  En el caso sea más de un método de pago se enviará separado por comas, ejemplo:  1,2 |
|  |  |  |  |

## \* Los demás campos son referenciados en la generación del cip

## ConsultarCip

## Para consultar el estado de uno o varios CIPs, está disponible un Servicio Web ConsultarCIP, a continuación un ejemplo del código en el framework Zend en php:

if($\_POST['cip']){

$numcip = $\_POST['cip'];

$response = $pagoefectivo->consultarCip($numcip);

if($response->Estado != -1)

var\_dump($response);

} else

echo 'Debe ingresar un número de cip';

## El resultado de la operación devuelve una entidad del tipo BEWSConsultarCipResponse.

## Descripción de objetos

## BEWSConsultarCipRequest

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo Dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| CAPI | Cadena | SI | Código API asociada al Servicio que solicita el uso del servicio web |
| CClave | Cadena | SI | Clave asociada al Servicio que solicita el uso del servicio web. |
| CIPs | Arreglos de Cadenas | SI | Arreglo de cadenas que contienen los números de los CIPs a consultar. |
| InfoRequest | Cadena | No |  |

## BEWSConsultarCipResponse:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de dato o Estructura) |
| Estado | String(1) | “-1”: Se dio un error no controlado.  “0”: No existe la solicitud de pago.  “1”: Se encontró la Solicitud de pago. |
| Mensaje | String | Contiene el mensaje resultado de haber realizado la operación:  En el caso del Estado “-1”:   * “Ha ocurrido un Error no Manejado”   En el caso del Estado “0”:   * “No se encontró el cip con id: “1454”   En el caso del Estado “1”:   * “Se encontró el cip” |
| CIPs | Arreglo de BEWSConsCIP | CIPS conciliados y no conciliados de acuerdo a filtros |
| InfoResponse | Cadena |  |

## BEWSConsCIP:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdOrdenPago | Cadena | Identificador único del CIP |
| IdCliente | Cadena | Identificador único del Cliente en PagoEfectivo |
| NumeroOrdenPago | Cadena | Número de CIP Generado |
| OrderIdComercio | Cadena | Número de Código de Referencia al Pago |
| IdMoneda | Cadena | Código de la moneda: 1 : Soles y 2: Dólares |
| Total | Cadena | Monto total del Pago |
| FechaEmision | Cadena | Fecha en la que se emitió el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| FechaCancelacion | Cadena | Fecha en la que se canceló el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| UsuarioID | Cadena | Código de usuario de referencia |
| UsuarioNombre | Cadena | Nombre de usuario de referencia de la generación del CIP |
| UsuarioApellidos | Cadena | Apellidos del usuario de referencia de la generación del CIP |
| UsuarioAlias | Cadena | Alias del usuario de referencia de la generación del CIP |
| UsuarioEmail | Cadena | Email del usuario de referencia de la generación del CIP |
| DataAdicional | Cadena | Información de referencia |
| Estado | Cadena | Estado del CIP (Generada, Cancelada, Expirado, Eliminado) |
| Detalle | Arreglo de BEWSConsCIPDetalle | Contiene los detalles del pago asociado |

## BEWSConsCIPDetalle:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdDetalleOrdenPago | Cadena | Identificador único del detalle del CIP |
| ConceptoPago | Cadena | Descripción del Concepto de Pago |
| Importe | Cadena | Monto del pago del detalle. |
| Tipo\_Origen | Cadena | Tipo de Código de Origen (OP: Orden Publicación, BOL: Boleta, FAC: Factura) |
| Cod\_Origen | Cadena | Código de Origen que indica el # de documento a Cancelar o # de documento de Referencia. |
| NumRecibo | Cadena | Número de Serie y Correlativo (En caso de que el tipo Origen es BOL o FAC) |
| FechaVencimiento | Cadena | Fecha de vencimiento del documento (En caso de que el tipo Origen es BOL o FAC) |
| CodigoEnteCliente | Cadena | Código de Ente del Cliente. |

## DesencriptarTexto

## Nos permite desencriptar un texto, esta operación se usara en las notificaciones, PagoEfectivo envía notificaciones a la entidad vía request a una página, por método POST envía un dato encriptado en el cual se encuentra la información de la notificación, que procederá a desencriptar usando el API de la siguiente manera:

//$string = "AS65D4A6SD54A61D3AS21D3AS4D65W4E4W564E65W4";

$info = $this->\_loadService('DecryptText', array('encryptText' => $string, 'privateKey' => $this->getPrivateKey()));

return ($info!==false) ? $info->DecryptTextResult : false;

## Donde resultado es el texto desencriptado y tendrá la estructura XML de “ConfirSolPago”

## ConfirSolPago:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdResSolPago | Int64 | Identificador único de la Solicitud de pago |
| CodTrans | String | Número de Código del servicio de Referencia al Pago |
| Token | String | Código Único generado para la operación. |
| MetodoPago | String | Códigos de Pago, puede n ser:  1: Bancos/FullCarga  2: Cuenta Virtual (Monedero Electrónico)  3: Visa  4: MasterCard  En el caso sea más de un método de pago se enviara separado por comas, ejemplo:  1,3,4 |
| Codigo | String | Código de Servicio de Referencia. Ejemplo: Neoauto: NEO, Urbania: URB, Kotear: KOT |
| Fecha | DateTime | Campo la fecha en que realizo la operación, tiene el formato es DIA/MES/ANIO HORA:MINUTO:SEGUNDO |
| Estado | Integer | Estado en que se encuentra la Operación:  1: Solicitud de Pago Pendiente  2: Solicitud de Pago Generada (generó CIP)  3: Solicitud de Pago Pagado con Dinero Virtual  4: Solicitud de Pago Pagado con Visa  5: Solicitud de Pago Pagado con MasterCard  6: Solicitud de Pago Expirada (sobrepaso el tiempo de expiración) |
| (\*)ParamsURL | XML | Parámetros URL que se enviaron al realizar la solicitud de Pago |
| (\*\*)CIP | XML | Si la solicitud genero un CIP se retornará los datos de dicha CIP en un formato XML. |

## (\*)ParamsURL:

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Tipo de dato | Mandatorio | Descripción y Estructura  (Tipo de Dato o Estructura) |
| Nombre | String | SI | Nombre del Parámetro que se enviara al momento de notificar. |
| Valor | String | SI | Valor del Parámetro que se enviará al hacer request a la url de notificación. |

## (\*\*)CIP:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdOrdenPago | Int64 | Identificador único del CIP |
| IdCliente | Integer | Identificador único del Cliente en Pago Efectivo |
| IdEstado | Integer | Indica el estado 21:Expirado,22:Generada,23:Cancelada,25:Eliminado |
| IdServicio | Integer | Indicador único del servicio |
| IdMoneda | Integer | Código de la moneda: 1 : Soles y 2: Dólares |
| NumeroOrdenPago | String | Número de CIP Generado |
| OrderIdComercio | String | Número de Código de referencia al pago |
| Total | Decimal(18,2) | Monto total del pago. |
| MerchantID | String | Código de referencia del servicio |
| FechaEmision | String | Fecha en la que se emitió el CIP (formato: aaa-mm-dd hh:mm:ss) |
| FechaCancelado | String | Fecha en la que se canceló el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| FechaConciliado | String | Fecha en la que se concilió el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| FechaAnulado | String | Fecha en la que se concilió el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| FechaExpirado | String | Fecha en la que se concilió el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| FechaEliminado | String | Fecha en la que se concilió el CIP (formato: aaaa-mm-dd hh:mm:ss) |
| TiempoExpiracion | Integer | Número en Horas de disponibilidad del CIP para cancelarlo que es válido desde el tiempo que se emite. |
| EstaConciliado | String | Campo que indica si está conciliado: 1 Si está conciliado o 2 si todavía no se ha conciliado el pago. |
| MailComercio | String | Correo electrónico de administrador de referencia. |
| UsuarioID | String | Código de usuario de referencia |
| DataAdicional | String | Información de referencia |
| UsuarioNombre | String | Nombre de usuario de referencia de la generación del CIP |
| UsuarioApellidos | String | Apellidos del usuario de referencia de la generación del CIP |
| UsuarioAlias | String | Alias del usuario de referencia de la generación del CIP |
| UsuarioEmail | String | Email del usuario de referencia de la generación del CIP |
| UsuarioDomicilio | String | Descripción del domicilio del usuario |
| UsuarioLocalidad | String | Descripción de la localidad del usuario |
| UsuarioProvincia | String | Descripción de la provincia del usuario |
| UsuarioPais | String | Descripción del país del usuario |
| (\*\*\*)Detalles | XML | Contiene los detalles del pago asociado |

## (\*\*\*)Detalles:

|  |  |  |
| --- | --- | --- |
| Nombre | Tipo de dato | Descripción y Estructura  (Tipo de Dato o Estructura) |
| IdDetalleOrdenPago | Int64 | Identificador único del detalle del CIP |
| ConceptoPago | String | Descripción del Concepto de Pago |
| Importe | Decimal(18,2) | Monto del pago del detalle. |
| Tipo\_Origen | String | Tipo de Código de Origen (OP: Orden Publicación, BOL: Boleta, FAC: Factura) |
| Cod\_Origen | String | Código de Origen que indica el # de documento a Cancelar o # de documento de Referencia. |
| NumRecibo | String | Número de Serie y Correlativo (En caso de que el tipo Origen es BOL o FAC) |
| FechaVencimiento | String | Fecha de vencimiento del documento (En caso de que el tipo Origen es BOL o FAC) |
| CodigoEnteCliente | String | Código de Ente del Cliente. |

## La notificación de PagoEfectivo hacia los Servicios es por medio de una petición web (WebRequest), hacia una ruta (UrlNotificación) que haya establecido el Servicio. Esta notificación se da en los siguientes casos:

## Generación de CIP

## Pago de CIP

## Anulación (Extorno) de CIP

## Expiración de CIP

## La notificación que haga PagoEfectivo contendrá 2 campos, teniendo como parámetro variable al Estado:

## data: Con la misma trama encriptada que en ConfirSolPago utilizado en la consulta de solicitud.

## versión: Con la versión actual de la integración, la actual es la 2.

## El campo Estado podrá tener los siguientes valores, según el evento que se realice:

## 592: Solicitud de Pago Pendiente (CIP fue extornado, el banco realizo reversa de Pago)

## 593: Solicitud de Pago Pagado

## 595: Solicitud de Pago Expirada (sobrepaso el tiempo de expiración)

$pagoefectivo = new App\_Service\_PagoEfectivo();

$data = $\_POST['data'];

$version = $\_POST['version'];

//asegurarse que los 2 parametros sean enviados

if($data && $version == '2'){

//desencriptar la data y darle formato

$solData = simplexml\_load\_string($pagoefectivo->desencriptarData($data));

//ID Transaccion para guardarla en la BD

$idPayment = $solData->CodTrans;

//Según el estado de la solicitud 🡪 Procesar

Switch($solData->Estado){

case 592: //

echo 'Estado: pendiente de pago';

//o bien registrarlo en un log

$pagoefectivo->addRowFileLog(\_\_DIR\_\_.'/log/cip.txt', $solData->CIP->NumeroOrdenPago.':GENERADO');

break;

case 593: //Cip Pagado

$pagoefectivo->addRowFileLog(\_\_DIR\_\_.'/log/cip.txt', $solData->CIP->NumeroOrdenPago.':PAGADO');

echo 'Estado: Generado';

break;

case 595://Cip Vencido

$pagoefectivo->addRowFileLog(\_\_DIR\_\_.'/log/cip.txt', $solData->CIP->NumeroOrdenPago.':VENCIDO');

echo 'Estado: Vencido';

break;

default: echo 'Estado: ERROR'; return;

}

}

exit();

La url de notificación debe existir en fisico aunque todavia no procese nada, algo como   
<http://URL/pagoefectivo/notificacion>   o el que uds elijan, sin parametros es decir sin el caracter '?'  ni '&'

PagoEfectivo enviara por POST 2 parametros a la url que nos indiquen:  
VERSION (la modalidad que uds usan es la 2), y el mas importante el DATA que es el que contendra los datos de la solicitud de pago y del CIP generado todo esto llegará a uds encriptado.

hay un ejemplo en el archivo notificacion.php del kit de integracion, al desencriptar el parametro "data" la trama resultante sería: (Trama de Ejemplo)

  <?xml version="1.0" encoding="utf-8"?>

<ConfirSolPago>

<idResSolPago>1012</idResSolPago>

<CodTrans>2000002</CodTrans>

<Token>d8dfdfc6-dc2e-4d16-91ae-8b36a23af074</Token>

<MetodoPago>1</MetodoPago>

<Codigo>GNA</Codigo>

<Fecha>2013-01-11 18:27:07</Fecha>

<Estado>592</Estado>

<ParamsURL></ParamsURL>

<CIP>

<IdOrdenPago>1044962</IdOrdenPago>

<IdCliente>1150</IdCliente>

<IdEstado>22</IdEstado>

<IdServicio>48</IdServicio>

<IdMoneda>1</IdMoneda>

<NumeroOrdenPago>00000001044962</NumeroOrdenPago>

<OrderIdComercio>2000002</OrderIdComercio>

<Total>22.00</Total>

<MerchantID>GNA</MerchantID>

<FechaEmision>2013-01-11 18:27:23</FechaEmision>

<FechaCancelado></FechaCancelado>

<FechaConciliado></FechaConciliado>

<FechaAnulado></FechaAnulado>

<FechaExpirado></FechaExpirado>

<FechaEliminado></FechaEliminado>

<TiempoExpiracion>478.543333333333</TiempoExpiracion>

<EstaConciliado></EstaConciliado>

<MailComercio></MailComercio>

<UsuarioID>1201</UsuarioID>

<DataAdicional></DataAdicional>

<UsuarioNombre>Joe</UsuarioNombre>

<UsuarioApellidos>Doe</UsuarioApellidos>

<UsuarioAlias></UsuarioAlias>

<UsuarioEmail>[joedoe@mail.com](mailto:joedoe@mail.com)</UsuarioEmail>

<UsuarioDomicilio></UsuarioDomicilio>

<UsuarioLocalidad></UsuarioLocalidad>

<UsuarioProvincia></UsuarioProvincia>

<UsuarioPais></UsuarioPais>

<Detalles><Detalle>

<IdDetalleOrdenPago>49382</IdDetalleOrdenPago>

<ConceptoPago>detail transaction 1</ConceptoPago>

<Importe>10.00</Importe>

<Tipo\_Origen>FAC</Tipo\_Origen>

<Cod\_Origen></Cod\_Origen>

<Campo1></Campo1>

<Campo2></Campo2>

<Campo3></Campo3>

</Detalle>

<Detalle>

<IdDetalleOrdenPago>49383</IdDetalleOrdenPago>

<ConceptoPago>detail transaction 2</ConceptoPago>

<Importe>12.00</Importe>

<Tipo\_Origen>FAC</Tipo\_Origen>

<Cod\_Origen></Cod\_Origen>

<Campo1></Campo1>

<Campo2></Campo2>

<Campo3></Campo3>

</Detalle>

</Detalles>

</CIP>

<Trans></Trans>

</ConfirSolPago>

 De estos datos lo mas resaltantes:  
  
**<Estado>592</Estado>**

* **592:** Solicitud de Pago Pendiente

                  Actualizar la orden local como Pendiente de Pago

           FIN SI

* **593:** Solicitud de Pago Pagado

          Para este caso es facil solo deberan buscar su orden local y actualizarla como pagada

           Su orden local viene en <OrderIdComercio>2000002</OrderIdComercio> para el ejemplo

* **595:** Solicitud de Pago Expirada (sobrepaso el tiempo de expiración)

          De la misma manera para expirada, ubicar la orden y actualizarla como Anulada (un cip expirado ya no puede ser pagada)

 Queda indicar que un cip extornado queda hábil de nuevo para ser pagado por cualquier medio disponible en PagoEfectivo, y que un CIP expirado ya no se puede pagar.