ePay.bg communication package for merchants

I. Payment request by WEB merchant
II. Payment request by WEB merchant (direct credit card payment)
III. Payment request - "Free transfer"

ePay.bg communication package for merchants

ePay.bg <office@epay.bg>, <epay2_demo@epay.bg>

The package consists of the following files:

demo.cgi
- sample payment request written in Perl

demo.php
- sample payment request written in PHP

demo_recv.cgi
- example for receiving a notification concerning a registered payment order (Paid or Rejected) written in Perl

demo_recv.php
- example for receiving a notification concerning a registered payment order (Paid or Rejected) written in PHP

demo.html
- simple payment request by user/merchant, "Free transfer"

I. Payment request by WEB merchant

Scheme:

1. A customer of the WEB merchant makes an order in merchant's web site.

2. After being ready with the order, the customer proceed to payment by choosing the payment button on merchant's web site, which creates “a payment request”.

3. The merchant makes a package - payment request and redirects the customer to the ePay.bg system, by applying the payment request in the way described below.

4. At ePay.bg web site the customer chooses how to pay – with bank card trough ePay.bg, in cash in Easypay or at ATMs.

4.1 If the customer chooses to use your ePay.bg profile he/she logs in ePay.bg system using his/her user name and password and after a successful login the system, the payment request (if it is correctly applied) is registered/recorded as a pending bill to the this merchant. The customer confirms the payment by choosing payment instrument (registered bank card or microaccount) and confirms the payment entering his/her password.

If the customer does not make login at all or makes an unsuccessful tries to login, the payment request is NOT recorded in the system.

The customer can pay the bill immediately or to postpone the payment.

4.2 If the customer chooses to pay in cash (in Easypay) or with bank card at an ATM, he/she must use automatically generated by ePay.bg **10-digit code for payment.** ePay.bg automatically generated 10-digit payment code related to each payment request sent by the merchant. This code is visible at ePay.bg website when the customer choose to pay in cash or at an ATM.

5. The ePay.bg system keeps track the status of the registered/recorded pending bills and when a payment is made, rejected or expired, a notification about the status of the payment is sent to the merchant . The notifications are sent to ports 80 for HTTP and 443 for HTTPS.

6. When receives a notification from ePay.bg, the merchant should form a corresponding answer:

```
Merchant   --[PAYMENT REQUEST]-->             ePay.bg
Customer   --[ENTRY]--[REQUEST RECORDING]-->  ePay.bg
ePay.bg    --[NOTIFICATION]-->                Merchant
Merchant   --[NOTIFICATION REPLY]-->          ePay.bg
```

Each merchant registered in the system has generated **alpha-numeric secret word with length 64 and an identification number** (CIN). The merchant can see them in his personal data profile without being able to change them.

A. **The payment request is sent as HTTP POST request to https://www.epay.bg/**

Example:

```
<form action="https://www.epay.bg/" method=post>
 <input type=hidden name=PAGE value="paylogin">
 <input type=hidden name=ENCODED value="[ENCODED]">
 <input type=hidden name=CHECKSUM value="[CHECKSUM]">
 <input type=hidden name=URL_OK value="http://...">
 <input type=hidden name=URL_CANCEL value="http://...">
 <input type=submit>
 </form>
```

(*) If the ePay.bg DEMO system is used at address:
https://devep2.datamax.bg/ep2/epay2_demo/, the TEST payment requests will be sent to it!!!

The following fields are compulsory: PAGE, ENCODED and CHECKSUM, the value of PAGE obligatorily being "paylogin".

Fields URL_OK and URL_CANCEL are optional:

URL_OK - URL to which the customer will be referred in case he confirms the payment (**it does not guarantee the payment has been done.** A payment is successfully processed, if the merchant has been received the relevant message from ePay.bg.)

URL_CANCEL - URL to which the customer will be referred in case he denies the payment for the moment (he can pay it or reject it later but not after the date, indicated by the merchant)

Signing of the payment request:

ENCODED - encoded with base64 payment request, EOL=''
CHECKSUM - checksum on ENCODED, generated as HMAC with
an algorithm SHA-1 and the secret word of the merchant.

Perl example code: {

```
 # Encode the payment request
 $ENCODED  = encode_base64('DATA', ''); # '' for EOL (def. å "\n")

 # Generate checksum
 $CHECKSUM = hmac_hex($ENCODED, $secret, \&sha1);
```

}

PHP example code: {

```
 # Encode the payment request
 $ENCODED  = base64_encode('DATA');

 # Generate checksum
 $CHECKSUM = hmac('sha1', $ENCODED, $secret);
 # the code of hmac function can be seen in demo.php
```

}

Example of payment request data:

MIN=1000000000	(obligatory - Identification number of the merchant)	
INVOICE=123456	(obligatory	Invoice number)
AMOUNT=22.80	(obligatory	Amount)
CURRENCY=BGN	(optional	Currency - BGN, USD or EUR)
EXP_TIME=01.08.2020	(obligatory	Final date/hour for payment)
DESCR=Test	(optional	Description up to 100 characters)
ENCODING=utf-8	(optional	encoding of the DESCR parameter)

This order is not obligatory.

MIN - only digits, corresponds to CIN in personal data of the merchant at his ePay.bg profile
INVOICE - only digits
AMOUNT - valid amount > 0.01 (for example: 22, 22.8, 22.80)
CURRENCY - accepted currencies are BGN, USD or EUR; if not present then BGN by default
DESCR - CP1251 characters if not passed another ENCODING
ENCODING - accepted is only utf-8; can be passed and as an HTTP parameter

The following format is valid for field EXP_TIME:

```
EXP_TIME=DD.MM.YYYY[ hh:mm[:ss]]

EXP_TIME=01.08.2020
EXP_TIME=01.08.2020 23:15          (can also be applied with hour:min)
EXP_TIME=01.08.2020 23:15:30       (can also be applied with hour:min:sec)
```

After the certain payment request is registered, the system will notify the merchant about the payment state: 'Paid', 'Rejected' or 'Expired'. The payment is marked as 'Expired', if the customer does not confirm or rejects the order before the applied final date.

Request with certain INVOICE can be registered in the system only ONE time and it waits for 'confirmation' or 'rejection' of the customer.

B. The notification of the system is sent to given by the merchant URL as HTTP POST request, to which the merchant returns a reply in the same HTTP session.

Signing of the payment notification
ENCODED - encoded with base64 (RFC 3548) notification, EOL="
CHECKSUM - checksum on ENCODED, generated as HMAC with
an algorithm SHA-1 and the secret word of the merchant.

Perl example code: {

```
 $data = decode_base64($ENCODED);

 # Calculate checksum
 # Must $CHECKSUM_CALC == $CHECKSUM
 $CHECKSUM_CALC = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP example code: {

```
 $data = base64_decode($ENCODED);

 # Calculate checksum
 # Must $CHECKSUM_CALC == $CHECKSUM
 $CHECKSUM_CALC = hmac('sha1', $ENCODED, $secret);
 # the code of hmac function can be seen in demo.php

        }
```

Sample notification from ePay.bg:

```
INVOICE=123456:STATUS=PAID:PAY_TIME=YYYYMMDDhhmmss:STAN=[6 digits]:BCODE=[6 digits/letters]
INVOICE=123457:STATUS=DENIED
```

INVOICE=123457:STATUS=EXPIRED

STATUS=[PAID | DENIED | EXPIRED] - Paid | Rejected | Expired
PAY_TIME - Date/hour/sec of the payment
STAN - Transaction number
BCODE - Authorization code

For each invoice number in the notification the merchant should send back status:
OK - if the notification is successfully processed by merchant
ERR - for error
NO - if he does not know about this invoice

At the sending back of OK or NO the system stops sending notification about the invoice.

Sample reply of the merchant:

```
INVOICE=123456:STATUS=OK
INVOICE=123457:STATUS=ERR
INVOICE=123458:STATUS=NO
```

In the most usual case the merchant sends back OK and ePay.bg ceases to send subsequent notification messages.
If there is something incorrect in the notification sent by the ePay.bg system, the merchant sends back ERR=description.

Example:
ERR=description of the global error (for example incorrect CHECKSUM)

If ePay.bg system does not mark certain invoice as received by the merchant (for example returned status ERR or failed communication), the system will make an attempt to send the failed notifications again.

Scheme for sending notifications to a given invoice:
1) 5 attempts at an interval of < 1 minute
2) 4 attempts at an interval of 15 minutes
3) 5 attempts at an interval of 1 hour
4) 6 attempts at an interval of 3 hours
5) 4 attempts at an interval of 6 hours
6) 1 attempt a day

The system stops sending notification for an invoice, if it is not received by the merchant in the course of 30 days.

If the merchant has not announced URL, at which he can receive notifications about the payments, does not want or does not have the possibility to process these notifications, the merchant can see the status of his payment orders in the ePay.bg system.

II. Payment request by WEB merchant (direct credit card payment)

The scheme is the same as "(I) Payment request by WEB merchant, with the difference that PAGE=credit_paydirect.

Example:

```
<form action="https://www.epay.bg/" method=post>
<input type=hidden name=PAGE value="credit_paydirect">
<input type=hidden name=LANG value="[LANG]">
<input type=hidden name=ENCODED value="[ENCODED]">
<input type=hidden name=CHECKSUM value="[CHECKSUM]">
<input type=hidden name=URL_OK value="http://...">
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>

LANG = bg | en
```

(*) If the ePay.bg DEMO system is used at address:
https://devep2.datamax.bg/ep2/epay2_demo/, the TEST payment requests will be sent to this address!!!

For testing purposes: for card number - enter number 1111111111111111, for card expiration date – enter 11/11 .

III. Simple payment request - "Free transfer"

By Simple payment request - "Free transfer" an ePay customer may automatically send payment requests to more than one ePay customers. For each payments may be specified different amount and ground of the payment. Both customer and receiver must be registered users in ePay.bg

The customer who pays may personally choose how to pay - with a bank card over the Internet, through an ATM, through the mobile application, through bank transfer or by paying cash in Easypay.
The amount paid by customer is received at a Microaccount of the receiver.

The request for "Free transfer" is sent as HTTP POST request to https://www.epay.bg/

Example:

```
<form action="https://www.epay.bg/" method=post>
<input type=hidden name=PAGE value="paylogin">
<input type=hidden name=MIN value="[MIN]">
<input type=hidden name=INVOICE value="[INVOICE]">
<input type=hidden name=TOTAL value="[TOTAL]">
<input type=hidden name=DESCR value="[DESCR]">
<input type=hidden name=ENCODING value="[ENCODING]">
<input type=hidden name=URL_OK value="http://...">
```

```
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>
```

The value of field PAGE is obligatorily "paylogin".

MIN   (obligatory Identification number in ePay.bg)
INVOICE   (optional   Invoice number)
TOTAL   (obligatory Amount)
DESCR   (optional   Description up to 100 characters)
ENCODING (optional   encoding of the DESCR parameter)

MIN   - only digits (corresponds to CIN in personal data of the receiver)
INVOICE - only digits
TOTAL   - valid amount > 0.01 (for example: 22, 22.8, 22.80)
DESCR   - CP1251 characters if not passed another ENCODING
ENCODING - accepted is only utf-8

AMOUNT is acceptable equivalently to TOTAL.

Scheme:

Receiver   --[REQUEST FOR FREE TRANSFER]-->   ePay.bg
Customer   --[ENTRY]--[REQUEST RECORD]-->   ePay.bg

Fields URL_OK and URL_CANCEL are optional:

URL_OK   - URL to which the customer will be referred in case he
            confirms the payment (does not guarantee the payment will be done)
URL_CANCEL - URL to which the customer will be referred in case he
            rejects the payment

In this scheme of payment there is neither signing of the requests nor notifications sending to the merchant about the payment status.

END