

Payout API Interface

Version 1.1

Table of Contents

[Document Illustration](#)

[Preparation](#)

[Add beneficiary via curl](#)

[Add fund via curl](#)

[Send fund via curl](#)

[Check payment status](#)

[Response Status](#)

[Payment Status](#)

[Beneficiary Status](#)

[Currency Support](#)

Document Illustration

The documents provider is PSP, we focus on providing our merchants a standard API integration rules. We have the right of final explanation on the document and have reserved right to update it any time.

1. After using the document, you can achieve the following functions: 'Accept credit payment online', 'Query order information', 'Refund the order' and so on.
2. The documents reader is merchants' administrator, maintainer and developer.
3. Except defined, platform mentioned in this document means PSP.

Preparation

1. GatewayURL - Provided by support team.
2. Read the entire API document carefully.
3. Obtain 3 very important parameters, they are "**secret_key**", "**payout_token**", "**payout_secret_key**".

Replace of 3 very important parameters. You may find it at Payout:

1. **secret_key**
2. **payout_token**
3. **payout_secret_key**

Preparation steps

You need to create an account.

Setup your web site for proper work with PSP. You will need to specify the following:

1. **Secret Key:** This is a string value. You can generate via **Payout Key** section under payout.
2. **Payout Token:** This is a 89 characters string value. You can generate via **Payout Key** section under payout.
3. **Payout Secret Key:** You can generate Payout Secret Key via enter Security Word
4. **Success URL (success_url):** This is the URL address where you will be redirected upon successful payment execution.
5. **Notify URL (notify_url):** Notify url or callback url will notify you that payment has been processed successfully.
6. **Fail URL (error_url):** This is the URL address of a web page that you will be redirected, when he cancels his payment or when his payment has failed.

Encryption pattern

Payload content-type will be in text/plain. Base64 encoded encrypted Cipher will be passed as a payload.

```
$payload = "payout_token={payout_token}&payout_secret_key={payout_secret_key}&checkout=CURL&client_ip={$_SERVER['SERVER_ADDR']}&source=Encode-Curl-API&source_url={source_url}&param1={param1}&param2={param2}&param3={param3}....notify_url={notify_url}&success_url={success_url}&error_url={error_url}";
```

```
$iv = substr(hash('sha256', $payout_token), 0, 16);  
$output=trim(strtr(base64_encode(openssl_encrypt($payload, "AES-256-CBC",$secret_key,0,$iv)),'+/','-_'),'=');
```

Add beneficiary

Endpoint: "/payout/addbeneficiary.do"

Table below lists all acceptable fields via POST method (curl).

Parameters	Required	Format	Description
secret_key	Y	String (64 char)	Generate and copy from Payout Section
payout_token	Y	String (86 char)	Generate and copy from Payout Section
payout_secret_key	Y	String	Security Word. Generate through Payout Section
client_ip	Y	String	Valid whitelisted IP. Method: (\$_SERVER['HTTP_X_FORWARDED_FOR']? \$_SERVER['HTTP_X_FORWARDED_FOR'] : \$_SERVER['SERVER_ADDR'])
source	Y	String	a string for default (fixed) value Encode-Curl-API
source_url	Y	String	Product or Service url of source_url as per your domain - isset(\$_SERVER["HTTPS"]) ? 'https://' : 'http://'. \$_SERVER['SERVER_NAME']. \$_SERVER['REQUEST_URI'].
beneficiary_type	Y	String	Fiat or Crypto
beneficiary_nickname	Y	String	Valid Nickname
beneficiary_name	Y	String	Account holder name as mention in account
account_number	Y	Alpha Numeric	Bank Account Number or Crypto Wallet Address
beneficiary_ac_repeat	Y	Alpha Numeric	Bank Account Number or Crypto Wallet Address - Repeat
beneficiary_bank_name	Y	String	Name of Bank or Crypto Network
bank_code1	N	String	Bank code as ISO standard or Crypto Wallet Provider
bank_code2	Y	String	Use for bank unique code or Crypto Coin Name
bank_code3	N	String	Use for Swift code for International transaction
notify_url	Y	String	Synchronous notification URL, you can find it in the response / callback data.
success_url	N	String	A valid url, you can find the redirect after SUCCESS response.
error_url	N	String	A valid url, you can find the redirect after FAIL response.

Sample code for add beneficiary

```
$payload =  
"payout_token={payout_token}&payout_secret_key={payout_secret_key}&checkout=CURL&client_ip={$_SERVER['SERVER_ADDR']}&source=Encode-Curl-API&source_url={source_url}&beneficiary_nickname={Beneficiary Nickname}&beneficiary_name={Beneficiary Name}&account_number={Beneficiary Account Number}&beneficiary_ac_repeat={Beneficiary Account Number}&beneficiary_bank_name={Bank Name}&bank_code1={bank_code1}&bank_code2={bank_code2}&bank_code3={bank_code3}&notify_url={notify_url}&success_url={success_url}&error_url={error_url}";  
  
$iv = substr(hash('sha256', $payout_token), 0, 16);  
$encrypted = rtrim(strtr(base64_encode(openssl_encrypt($payload, "AES-256-CBC", $secret_key, 0, $iv)), '+/', '-_'), '=');
```



```
$pram_encode=array();
$pram_encode['pram_encode']=$encrypted.{payout_token};

$curl = curl_init();
curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_0);
curl_setopt($curl, CURLOPT_URL, {GatewayURL}."/payout/addbeneficiary.do");
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($curl, CURLOPT_USERAGENT, $_SERVER['HTTP_USER_AGENT']);
curl_setopt($curl, CURLOPT_REFERER, {source_url});
curl_setopt($curl, CURLOPT_POST, 1);
curl_setopt($curl, CURLOPT_POSTFIELDS, $pram_encode);
curl_setopt($curl, CURLOPT_HEADER, 0);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);

$response = curl_exec($curl);
curl_close($curl);
```

Success response:

```
{
    "status": "0000",
    "bene": "beneficiary_id",
    "reason": "Success",
    "remark": "Beneficiary successfully added"
}
```

Fail response:

```
{
    *"status": "status_code",
    "reason": "Failed reason",
}
```

*check status code at the end of document

Add fund

Endpoint: "/payout/addfund.do"

Table below lists all acceptable fields via POST method (curl).

Parameters	Required	Format	Description
secret_key	Y	String (64 char)	Generate and copy from Payout Section
payout_token	Y	String (86 char)	Generate and copy from Payout Section
payout_secret_key	Y	String	Security Word. Generate through Payout Section
client_ip	Y	String	Valid whitelisted IP. Method: (\$_SERVER['HTTP_X_FORWARDED_FOR']? \$_SERVER['HTTP_X_FORWARDED_FOR'] : \$_SERVER['SERVER_ADDR'])
source	Y	String	a string for default (fixed) value Encode-Curl-API.
source_url	Y	String	Product or Service url of source_url as per your domain - isset(\$_SERVER["HTTPS"]) ? 'https://': 'http://'. \$_SERVER['SERVER_NAME'].\$_SERVER['REQUEST_URI'].
amount	Y	Numeric	Amount format in decimal (x.xx)
curr	Y	String (3 char)	Three digit currency code as ISO standard (service provider for approved currencies)
remarks	Y	Strings	Detail about payment
request_id	Y	Alpha Numeric	Must be unique
product_name	Y	Numeric	Fixed value - 2
notify_url	Y	String	Synchronous notification URL, you can find it in the response / callback data.
success_url	N	String	A valid url, you can find the redirect after SUCCESS response.
error_url	N	String	A valid url, you can find the redirect after FAIL response.

Sample code for add fund

```
$payload =  
"payout_token={payout_token}&payout_secret_key={payout_secret_key}&checkout=CURL&client_ip={$_SERVER['SERVER_ADDR']}&source=Encode-Curl-API&source_url={source_url}&amount={amount}&curr={currency code}&remarks={remarks}&request_id={RequestId}&product_name=2&notify_url={notify_url}&success_url={success_url}&error_url={error_url}";  
  
$iv = substr(hash('sha256', $payout_token), 0, 16);  
$encrypted = rtrim(strtr(base64_encode(openssl_encrypt($payload, "AES-256-CBC", $secret_key, 0, $iv)), '+/', '-_'), '=');
```

```
$pram_encode=array();
$pram_encode['pram_encode']=$encrypted.{payout_token};

$curl = curl_init();
curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_0);
curl_setopt($curl, CURLOPT_URL, {GatewayURL}."/payout/addfund.do");
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($curl, CURLOPT_USERAGENT, $_SERVER['HTTP_USER_AGENT']);
curl_setopt($curl, CURLOPT_REFERER, {source_url});
curl_setopt($curl, CURLOPT_POST, 1);
curl_setopt($curl, CURLOPT_POSTFIELDS, $pram_encode);
curl_setopt($curl, CURLOPT_HEADER, 0);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);

$response = curl_exec($curl);
curl_close($curl);
```

Success response:

```
{
  "status": "0000",
  "transaction_id": "transaction_id"
  "amount": "amount",
  "reason": "Success"
}
```

Fail response:

```
{
  *"status": "status_code",
  "reason": "Failed reason",
}
```

*check status code at the end of document

Send fund

Endpoint: "/payout/sendpayout.do"

Table below lists all acceptable fields via POST method (curl).

Parameters	Required	Format	Description
secret_key	Y	String (64 char)	Generate and copy from Payout Section
payout_token	Y	String (86 char)	Generate and copy from Payout Section
payout_secret_key	Y	String	Security Word. Generate through Payout Section
client_ip	Y	String	Valid whitelisted IP. Method: (\$ _SERVER['HTTP_X_FORWARDED_FOR']? \$ _SERVER['HTTP_X_FORWARDED_FOR'] : \$ _SERVER['SERVER_ADDR'])
source	Y	String	a string for default (fixed) value Encode-Curl-API.
source_url	Y	String	Product or Service url of source_url as per your domain - isset(\$_SERVER["HTTPS"]) ? 'https://': 'http://'. \$_SERVER['SERVER_NAME']. \$_SERVER['REQUEST_URI'].
price	Y	Numeric	Amount format in decimal (x.xx)
curr	Y	String (3 char)	Three digit currency code as ISO standard (service provider for approved currencies) Or Crypto Currency such as USDT
remarks	Y	Strings	Detail about payment
narration	N	Strings	Detail of narration
request_id	Y	Alpha Numeric	Must be unique
product_name	Y	Numeric	Fixed value - 3
beneficiary_id	Y	Numeric	Fetch from beneficiary list
notify_url	Y	String	Synchronous notification URL, you can find it in the response / callback data.
success_url	N	String	A valid url, you can find the redirect after SUCCESS response.
error_url	N	String	A valid url, you can find the redirect after FAIL response.

Sample code for send fund

```
$payload =  
"payout_token={payout_token}&payout_secret_key={payout_secret_key}&checkout=CURL&client_ip={$_SERVER['SERVER_ADDR']}&source=Encode-Curl-API&source_url={source_url}&price={price}&curr={currency code}&remarks={remarks}&narration={narration}&request_id={RequestId}&product_name=3&notify_url={notify_url}&success_url={success_url}&error_url={error_url}";  
  
$iv = substr(hash('sha256', $payout_token), 0, 16);  
$encrypted = rtrim(strtr(base64_encode(openssl_encrypt($payload, "AES-256- CBC", $secret_key, 0, $iv)), '+', '-_'), '=');
```

```
$pram_encode=array();
$pram_encode['pram_encode']=$encrypted.{payout_token};

$curl = curl_init();
curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_0);
curl_setopt($curl, CURLOPT_URL, {GatewayURL}."/payout/sendpayout.do");
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($curl, CURLOPT_USERAGENT, $_SERVER['HTTP_USER_AGENT']);
curl_setopt($curl, CURLOPT_REFERER, {source_url});
curl_setopt($curl, CURLOPT_POST, 1);
curl_setopt($curl, CURLOPT_POSTFIELDS, $pram_encode);
curl_setopt($curl, CURLOPT_HEADER, 0);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
```

```
$response = curl_exec($curl);
curl_close($curl);
```

Success response:

```
{
    "status": "0000",
    "reason": "Success",
    "transaction_id": "transaction_id", //eg. 1233331233
    "status_nm": 1
    "payout_currency": "payout currency", //eg. USD - service provider for approved currencies
    "payout_amount": "paid amount", //eg. 9.99
    "available_balance": "Current available balance" //eg. 1234.56
}
```

Fail response:

```
{
    *"status": "status_code",
    "reason": "Failed reason",
}
```

*check status code at the end of document

Check Payment Status

Endpoint: "/payout/payoutdetail.do"

Table below lists all acceptable fields via POST method (curl).

Parameters	Required	Format	Description
secret_key	Y	String (64 char)	Generate and copy from Payout Section
payout_token	Y	String (86 char)	Generate and copy from Payout Section
payout_secret_key	Y	String	Security Word. Generate through Payout Section
client_ip	Y	String	Valid whitelisted IP. Method: (\$ _SERVER['HTTP_X_FORWARDED_FOR']? \$_SERVER['HTTP_X_FORWARDED_FOR'] : \$_SERVER['SERVER_ADDR'])
source	Y	String	a string for default (fixed) value Encode-Curl-API.
source_url	Y	String	Product or Service url of source_url as per your domain - isset(\$_SERVER["HTTPS"]) ? 'https://': 'http://' . \$_SERVER['SERVER_NAME'] . \$_SERVER['REQUEST_URI'].
transaction_id	Y	Numeric	Transaction id
order_number	Y	Alpha Numeric	Request id or order id

Sample code to check status

```
$payload =  
"payout_token={payout_token}&payout_secret_key={payout_secret_key}&checkout=CURL&client_ip={$_SERVER['SERVER_ADDR']}&source=Encode-Curl-API&source_url={source_url}&transaction_id={transaction_id}&order_number={order number or request id}";  
  
$iv = substr(hash('sha256', $payout_token), 0, 16);  
$encrypted = rtrim(strtr(base64_encode(openssl_encrypt($payload, "AES-256-CBC", $secret_key, 0, $iv)), '+/', '-_'), '=');
```

```

$pram_encode=array();
$pram_encode['pram_encode']=$encrypted.{payout_token};

$curl = curl_init();
curl_setopt($curl, CURLOPT_HTTP_VERSION, CURL_HTTP_VERSION_1_0);
curl_setopt($curl, CURLOPT_URL, {GatewayURL}."/payout/payoutdetail.do");
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($curl, CURLOPT_USERAGENT, $_SERVER['HTTP_USER_AGENT']);
curl_setopt($curl, CURLOPT_REFERER, {source_url});
curl_setopt($curl, CURLOPT_POST, 1);
curl_setopt($curl, CURLOPT_POSTFIELDS, $pram_encode);
curl_setopt($curl, CURLOPT_HEADER, 0);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);

$response = curl_exec($curl);
curl_close($curl);

```

Success response:

```

{
  "status": "0000",
  "transaction_type": "transaction type",
  "transaction_for": "transaction for",
  "transaction_date": "transaction date",
  "order_currency": "order currency",
  "order_amount": " order amount",
  "transaction_currency": "transaction currency",
  "transaction_amount": " transaction amount",
  "mdr_amt": "mdr amt",
  "mdr_percentage": "mdr percentage",
  "payout_amount": "payout amount",
  "available_balance": "available balance",
  "sender_name": "sender name",
  "beneficiary_id": "beneficiary id",
  "remarks": "remarks",
  "narration": "narration",
  "transaction_status": 1
  "notify_url": "notify url",
  "success_url": "success url",
  "failed_url": "failed url",
  "host_name": "host name"
}

```

Fail response:

```

{
  *"status": "status_code",
  "reason": "Failed reason",
}

```

*check status code at the end of document

Response code and status

- 0000 - Success
- 0001 - Low Balance
- 0002 - Duplicate Entry
- 0003 - Beneficiary not exists.
- 0004 - Unauthorized
- 0005 - Invalid payout token or Invalid secret key
- 0006 - Invalid value in action
- 0007 - transaction_id required
- 0008 - order_number required
- 0009 - transaction_id not exists
- 0010 - order_number not exists
- 0011 - Payout Secret Key is in cooling period

Payment status

- 0 – Pending
- 1 - Success
- 2 - Failed
- 3 - Processing
- 10 - Scrubbed

Beneficiary status

- 0 - Pending
- 1 - Permitted
- 2 - Blocked
- 3 - Under Process

Currency support

Please check with service provider for approved currencies