Invoicing and payment of bills

Purpose and main features of bills

Services for invoicing and payment of bills provides a simple way of processing a payment by link (URL) which client can use to follow and make a payment on order in IPS Assist. The merchant calls the service of bill creation (providing info about bill number, amount, payment timeout and etc.), IPS Assist makes a bill record and payment token associated to this bill. A bill also can be created manually within the personal account (https://account.paysec.by/). IPS Assist sends the link with the payment token to customer via E-mail or the enterprise generates an URL using the received payment token, and redirects the customer to the bill payment page. The enterprise also can send the payment link to customer via E-mail, SMS or via other means of communication from its own side. Payment link looks like the following:

https://<SERVER_NAME>/bill/paybill.cfm?ID=c2RmZ2UzNXkyNXR5MjU0

When the client follows by the link and agrees to pay the bill IPS Assist creates order. Next, the client enters his payment mean data and performs a payment similarly to the standard order payment.

This manual describes services for bill creation, cancellation of bill payment, bill information requests and process of payment. It also covers the questions of setting up of interaction with those services of IPS Assist.

Procedure of connection to bill invoicing and payment services for merchant

- send a request for connection to bill services to the Assist support team (support@belassist.by);
- receive the confirmation from Assist support team that all necessary settings are done for the merchant.

Bill creation service (createbill)

The service accepts the bill parameters in HTTP POST/SOAP or JSON (swagger specification: https://docs.belassist.by/swagger/) format and returns (in case of successful bill creation) a payment token/ID corresponding to that bill. The enterprise can make a payment link using this payment token/ID to redirect the customer to the bill payment page or send it to customer from its own side (E-mail, SMS or other type of communications), or IPS Assist can send it by E-mail customer's address specified in the parameters.

URL for the request:

https://<SERVER-NAME>/bill/createbill.cfm

List of request parameters:

Parameter	Mandatory field	Adopted values	Default value	Description
merchant_Id	Yes	Number		The enterprise identifier in IPS Assist
login	Yes	8 - 20 characters		Login (Latin letters, digits and symbol _)
password	Yes	8 - 20 characters		Password (Latin letters and digits)
bill	Yes/No [1]	String		Bill number (in creating order will be used as orderNumber)
bill_Amount	Yes	15 digits, two digits after the delimiter (delimiter '.')		Bill amount (*orderAmount* in order)

bill_Currency	Yes	String		Bill currency (orderCurrency)
bill_Comment	No	String		Comment (*orderComment* in order)
customer_Name	No	String		Customer's name (*firstName* in order)
customer_LastName	No	String		Customer's last name (*lastName* in order)
customer_MiddleName	No	String		Customer's middle name (*middleName* in order)
customer_Email	Yes/No 2	String		Customer's e-mail (Email in order)
customer_Phone	No	String		Customer's phone number (*homePhone* in order)
customer_Mobile	No	String		Customer's mobile phone number (*mobilePhone* in order)
language	No	RU EN	Setting of basic merchant account language	Language of authorized pages
pay_Until	No	Date/time	1 day	Expire date of bill payment (date and time in GMT): YYYYMMDDThhmm, when parameter is not passed then its value is calculated based on merchant settings
delayPayment	No	Number	0	Attribute of a credit card authorization for the double-stage operation mode (parameter *delay* in order): 0- one-stage operation, 1 - double-stage operation.
sendNotification	No	Number	Depending on merchant settings	Bill sending by E-mail: 0 – not send, 1 – send by E-mail, 2 – send by SMS, 3 - send by E-mail and by SMS.
url_Return	No	256 characters		Page URL, to where a customer should return after performing his/her payment in the system (option "*Go to the shop page*" should be enabled in merchant's settings in Personal account).). It is recommended to use HTTPS protocol for secure communication.
url_Return_Ok	No	256 characters	url_Return parameter value, or, if none, return URL parameter after successful authorization, specified in Personal account	Page URL, to where a customer should return after successful performing his/her payment in IPS Assist (option "*Go to the shop page*" should be enabled in merchant's settings in Personal account)). It is recommended to use HTTPS protocol for secure communication.
url_Return_No	No	256 characters	url_Return parameter value, or, if none, return URL specified in Personal account	Page URL, to where a customer should return if the payment is not performed in IPS Assist or payment is still in progress (current payment status "*In Progress*"), see details. The option "*Go to the shop using return URL*" should be enabled in merchant's settings in Personal account. It is recommended to use HTTPS protocol for secure communication.
cardPayment	No	1 – pay using bankcard; 0 – without using bankcard	1	A customer can perform his/her payment using bankcard, if parameter value 1 has been sent, or the parameter is missing (and bankcard payments are enable for the enterprise)
eripPayment	No	1 – payment with using ERIP; 0 – without using ERIP	1	A customer can perform his/her payment using ERIP, if parameter value 1 has been sent, or the parameter is missing (and ERIP payments are enable for the enterprise).

eposPayment	No	1 – use E-POS > payment, 0 – without E-POS	1	A customer can perform his/her E-POS payment, if parameter value 1 has been sent, or the parameter is missing (and E-POS payments are enable for the merchant)
googlePayPayment	No	1; 0	1	Payment via Google Pay attribute (=1)
applePayPayment	No	1; 0	1	Payment via Apple Pay attribute (=1)
samsungPayPayment	No	1; 0	1	Payment via Samsung Pay attribute (=1)
checkValue	Yes	String		Request validation code. [3]
customerNumber	No	String		Merchant's internal customer identification (for payments in OneClick program).

[1] parameter BILL is not mandatory in case when automatic bill numbering is switched on.

[2] If the value of the E-mail address is not sent, sending the bill by E-mail is not possible. If the value of phone number is not sent, sending the SMS is not possible. **Note** . For *sendNotification* = 3, if there is no phone number in the request, then the link to the bill will be sent only by E-mail; and if there is no E-mail address in the request, the notification will not be performed at all.

[3] Check value is calculated by following formula:

uppercase(md5(uppercase(md5(SALT) + md5(X)))) , where SALT – secret word; X – result of string concatenation (with semicolon (;) as delimiter) of the request parameters in order as they located in table from top to bottom excluding *delayPayment, sendNotification, checkValue* and *customerNumber*, '+' – string concatenation.

> All request parameters are automatically validated. The validation rules are given in the table " The validation rules for input parameters ".

When bill with the same number is already exist then new bill is not created and an error is returned: "*Bill with the specified number already exists*".

If request doesn't contain mandatory parameters or wrong data format provided the new bill is not created and an error is returned (similar to the error of order creation with not enough or wrong parameters).

When unique payment token/ID cannot be created the new bill is not created and an error is returned: "*Unique payment token/ID cannot be created*".

Information about using the 3-D Secure 2.0 protocol when creating a payment.

Example of bill creation request (HTTP POST):

```
<FORM ACTION="https://<SERVER-NAME>/bill/createbill.cfm" method="POST">
<INPUT TYPE="hidden" NAME="merchant_Id" VALUE="Your Merchant_ID">
<INPUT TYPE="hidden" NAME="login" VALUE="Your login">
<INPUT TYPE="hidden" NAME="password" VALUE="Your password">
<INPUT TYPE="hidden" NAME="bill" VALUE="Bill number">
<INPUT TYPE="hidden" NAME="bill_Amount" VALUE="Bill amount">
<INPUT TYPE="hidden" NAME="bill_Currency" VALUE="Bill currency">
<INPUT TYPE="hidden" NAME="bill_Comment" VALUE="Bill comment">
<INPUT TYPE="hidden" NAME="customer_Name" VALUE="Customer name">
<INPUT TYPE="hidden" NAME="customer_LastName" VALUE="Customer lastname">
<INPUT TYPE="hidden" NAME="customer_MiddleName" VALUE="Customer middlename">
<INPUT TYPE="hidden" NAME="customer_Email" VALUE="Customer Email">
<INPUT TYPE="hidden" NAME="customer_Phone" VALUE="Customer phone">
<INPUT TYPE="hidden" NAME="customer_Mobile" VALUE="Customer mobile phone">
<INPUT TYPE="hidden" NAME="language" VALUE="Language of authorized pages">
<INPUT TYPE="hidden" NAME="pay_Until" VALUE="Expire date of bill payment">
<INPUT TYPE="hidden" NAME="sendNotification" VALUE="Bill sending flag">
<INPUT TYPE="hidden" NAME="checkValue" VALUE="Request validation code">
<INPUT TYPE="Submit"></FORM>
```

List of response parameters:

Parameter	Value
Hash	Payment token/bill ID

Example of return parameter in CSV format:

```
Hash: xKPWpz4ZzDe5A9anPhnN
```

In XML format:

```
<?xml version='1.0' encoding='utf-8' standalone='yes'?>
<result
   firstcode="0"
   secondcode="0"
   count="1">
      <return>
          <Hash>akPWp08t84MTQ9anTy30</Hash>
      </return>
</result>
```

SOAP web-service description:

https://<SERVER_NAME>/bill/createbill.wsdl

Example of bill creation request:

```
<Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Header/>
    <Body>
        <CreateBill>
            <Bill>
                <merchant_id>423422</merchant_id>
                <login>login</login>
                <password>password</password>
                <bill>123456</bill>
                <bill_amount>100.00</bill_amount>
                <bill_currency>BYN</bill_currency>
                <bill_comment></bill_comment>
                <customer_name>Test</customer_name>
                <customer_lastname>Testov</customer_lastname>
                <customer_middlename>Testovich</customer_middlename>
                <customer_email>test@testpost.by</customer_email>
                <customer_phone></customer_phone>
                <language>EN</language>
                <pay_until>20150611T1212</pay_until>
                <sendnotification>0</sendnotification>
                <checkvalue> AA3BC3F48B7FE23988044B53AA98F169</checkvalue>
            </Bill>
        </CreateBill>
    </Body>
</Envelope>
```

Example of request result:

```
<soapenv:Envelope
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
        <ASS-NS:BillResponse xmlns:ASS-NS="http://www.paysecure.ru/ws/">
            <return xsi:type="si:SOAPStruct" xmlns:si="http://www.paysecure.ru/type/">
                <Hash xsi:type="xsd:string">6CPV7F4a0aWQg9XsXhrS</Hash>
            </return>
        </ASS-NS:BillResponse>
    </SOAP-ENV:Body>
```

Request for bill status (billstatus)

The service allows to receive the full data about bill, payment and cancellations on it. The data can be received by the number of bill or by the *billNumber* value.

When bill can't be found by his number the error is returned: "*Bill is not found*".

When it can't be found by *billNumber* the error appears: "*Bill is not found within processed*".

URL of request for bill status:

https://<SERVER-NAME>/bill/billstatus.cfm

List of request parameters:

Parameter	Mandatory field	Adopted values	Default value	Description
merchant_Id	Yes	Number		The merchant identifier in IPS Assist
login	Yes	8 - 20 characters		Login (Latin letters, digits and symbol _)
password	Yes	8 - 20 characters		Password (Latin letters and digits)
bill	Yes/No*	String		Bill number (in creating order will be used as *orderNumber*)

billNumber	Yes/No*	15 or 16 digits		Unique number of payment corresponding to the order number in IPS Assist

* One of the parameters Bill or BillNumber must be provided in the request. When both of them are provided then search is performed by *billNumber*.

Example of bill status request (HTTP POST):

```
<FORM ACTION="https://<SERVER-NAME>/bill/billstatus.cfm" method="POST">
<INPUT TYPE="hidden" NAME="merchant_Id" VALUE="Your Merchant_ID">
<INPUT TYPE="hidden" NAME="login" VALUE="Your login">
<INPUT TYPE="hidden" NAME="password" VALUE="Your password">
<INPUT TYPE="hidden" NAME="bill" VALUE="Bill number">
<INPUT TYPE="hidden" NAME="billNumber" VALUE="Number of payment">
<INPUT TYPE="Submit"></FORM>
```

List of response parameters:

Parameter	Value
merchant_id	The merchant identifier in IPS Assist
bill	Bill number
bill_amount	Bill amount
bill_currency	Bill currency
bill_comment	Bill comment
bill_date	Date and time of bill (GMT)
bill_status	Bill status
billnumber	Number of payment (empty if the order is not created) = *billNumber* of order
payment_date	Date and time of bill payment (empty if the order is not created) = *orderDate* (GMT)
payment_amount	Payment amount (empty if the order is not created) = amount of payment operation *amount*
payment_currency	Payment currency (empty if the order is not created) = currency of payment operation *currency*
payment_rrn	Additional identifier of payment/RRN (empty if there was no payment)
refund_amount	Total amount of successful refunds for this payment (empty if the order is not created; 0 if there was payment, but there wasn’t any refund)
customer_name	Customer name
customer_lastname	Customer lastname

customer_middlename	Customer middlename
customer_email	Customer Email-address
customer_phone	Customer phone number
customer_mobile	Customer mobile phone number
language	Language
pay_until	Expire date of bill payment (GMT)
sendnotification	Bill sending by E-mail (0 – not send, 1 - send, 10 – is sent)*
bill_type	Bill type (1 – by default)
checkvalue	Request validation code is calculated by following formula: uppercase(md5(uppercase(md5(SALT) + md5(X)))), where SALT – secret word; X – *r*esult of string concatenation (with semicolon (;) as delimiter) of the request parameters in order as they located in table from top to bottom excluding *sendnotification* and *checkvalue*, delimiter is also added in case of missing parameter, '+' – string concatenation.

* when E-mail sending was required in bill creation request (by parameter *sendNotification* =1 or by merchant settings in IPS Assist) then value of *sendnotification* in the result can have the following values:

1 – sending of E-mail message has not performed yet (for example when E-mail was not provided);

10 – E-mail message has been prepared and sent (it was placed in the sending queue).

Results example in CSV format:

```
merchant_id:515089; bill:2016.07.06 11:19-85; bill_amount: 100.00; bill_currency: BYN; bill_comment:; bill_date:
06.07.2016 08:19:41; bill_status:Invoiced; billNumber:;payment_date: ; payment_amount:100.00; payment_currency:
BYN; payment_rrn:; refund_amount:; customer_name:Test; customer_lastname:Testov; customer_middlename:;
customer_email:test@testpost.by; customer_phone:; customer_mobile:; language:EN; pay_until:20150701T0819;
sendnotification:1; checkValue:26397652E3DB8E25407A9B091B9431B3
```

In XML format:

```xml
<?xml version='1.0' encoding='utf-8' standalone='yes'?>
<result firstcode="0" secondcode="0" count="1">
   <orders>
      <order>
          <merchant_id>515089</merchant_id>
          <bill>20150630111985</bill>
          <bill_amount>100.00</bill_amount>
          <bill_currency>BYN</bill_currency>
          <bill_comment/>
          <bill_date>06.07.2016 08:19:41</bill_date>
          <bill_status>Invoiced</bill_status>
          <billNumber/>
          <payment_date></payment_date>
          <payment_amount>0.00</payment_amount>
          <payment_currency/>
          <payment_rrn/>
          <refund_amount/>
          <customer_name>Test</customer_name>
          <customer_lastname>Testov</customer_lastname>
          <customer_middlename/>
          <customer_email>test@testpost.by</customer_email>
          <customer_phone/>
          <customer_mobile/>
          <language>EN</language>
          <pay_until>20150701T0819</pay_until>
          <sendnotification>1</sendnotification>
          <checkValue>26397652E3DB8E25407A9B091B9431B3</checkValue>
      </order>
   </orders>
</result>
```

SOAP web-service description

https://<SERVER_NAME>/bill/billstatus.wsdl

Example of bill status request:

```xml
<Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Header/>
    <Body>
        <BillStatus>
            <Bill>
                <merchant_id>423422</merchant_id>
                <login>login</login>
                <password>password</password>
                <bill>123456</bill>
                <billnumber></billnumber>
            </Bill>
        </BillStatus>
    </Body>
</Envelope>
```

Example of results:

```
<soapenv:Envelope
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
        <ASS-NS:BillResponse xmlns:ASS-NS="http://www.paysecure.ru/ws/">
            <return xsi:type="si:SOAPStruct" xmlns:si="http://www.paysecure.ru/type/">
                <merchant_id xsi:type="xsd:string">515089</merchant_id>
                <bill xsi:type="xsd:string">20150610115527</bill>
                <bill_amount xsi:type="xsd:string">100.00</bill_amount>
                <bill_currency xsi:type="xsd:string">BYN</bill_currency>
                <bill_comment xsi:type="xsd:string">testbill</bill_comment>
                <bill_date xsi:type="xsd:string">06.07.2016 09:03:07</bill_date>
                <bill_status xsi:type="xsd:string">Invoiced</bill_status>
                <billNumber xsi:type="xsd:string"/>
                <payment_date xsi:type="xsd:string"></payment_date>
                <payment_amount xsi:type="xsd:string">0.00</payment_amount>
                <payment_currency xsi:type="xsd:string"/>
                <payment_rrn xsi:type="xsd:string"/>
                <refund_amount xsi:type="xsd:string"/>
                <customer_name xsi:type="xsd:string">test</customer_name>
                <customer_lastname xsi:type="xsd:string">testov</customer_lastname>
                <customer_middlename xsi:type="xsd:string"/>
                <customer_email xsi:type="xsd:string">test@testpost.by</customer_email>
                <customer_phone xsi:type="xsd:string"/>
                <customer_mobile xsi:type="xsd:string"/>
                <language xsi:type="xsd:string">EN</language>
                <pay_until xsi:type="xsd:string">0150611T0903</pay_until>
                <sendnotification xsi:type="xsd:string">0</sendnotification>
                <checkValue xsi:type="xsd:string">D9F0E4933007B96BCC2594983F7A24E3</checkValue>
            </return>
        </ASS-NS:BillResponse>
    </SOAP-ENV:Body>
```

Bill revocation request (billrevoke)

The service allows to revoke a bill that was not yet payed. Call of this service change the bill status to *Deleted* if it was in status *Inviced* or *Timed out* on moment of the request.

URL for bill revocation request:

https://<SERVER-NAME>/bill/billrevoke.cfm

List of request parameters:

Parameter	Mandatory field	Adopted values	Default value	Description
merchant_Id	Yes	Number		The merchant identifier in IPS Assist
login	Yes	8 - 20 characters		Login (Latin letters, digits and symbol _)
password	Yes	8 - 20 characters		Password (Latin letters and digits)
bill	Yes	String		Bill number
sendNotification	No	Number	Depending on merchant settings	Flag of sending notice of bill revocation (when this parameter is not passed in the request then the default setting for the merchant in IPS Assist is used).
checkValue	Yes	String		Request validation code (is calculated the same way as by bill creation, excluding *sendNotification* and *checkValue*)

Request example in HTTP POST format:

```
<FORM ACTION="https://test.paysecure.ru/bill/billrevoke.cfm" method="POST">
<INPUT TYPE="hidden" NAME="merchant_Id" VALUE="Your Merchant_ID">
<INPUT TYPE="hidden" NAME="login" VALUE="Your login">
<INPUT TYPE="hidden" NAME="password" VALUE="Your password">
<INPUT TYPE="hidden" NAME="bill" VALUE="Bill number">
<INPUT TYPE="hidden" NAME="sendNotification" VALUE="Notice sending flag">
<INPUT TYPE="hidden" NAME="checkValue" VALUE="Request validation code">
<INPUT TYPE="Submit"></FORM>
```

List of response parameters:

Parameter	Value
bill	Bill number
bill_status	Bill status

Results example in CSV format:

```
bill:2015.06.30 11:17-81; bill_status:Revoked
```

In XML format:

```
<?xml version='1.0' encoding='utf-8' standalone='yes'?>
<result firstcode="0" secondcode="0" count="1">
      <return>
          <bill>20150630111985</bill>
          <bill_status>Revoked</bill_status>
      </return>
</result>
```

SOAP web-service description:

https://<SERVER_NAME>/bill/billrevoke.wsdl

Example of bill revocation request:

```
<Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Header/>
    <Body>
        <BillRevoke>
            <Bill>
                <merchant_id>423422</merchant_id>
                <login>login</login>
                <password>password</password>
                <bill>123456</bill>
                <sendnotification>0</sendnotification>
                <checkvalue> AA3BC3F48B7FE23988044B53AA98F169</checkvalue>
            </Bill>
        </BillRevoke>
    </Body>
</Envelope>
```

Results example:

```
<soapenv:Envelope
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
        <ASS-NS:BillResponse xmlns:ASS-NS="http://www.paysecure.ru/ws/">
            <return xsi:type="si:SOAPStruct" xmlns:si="http://www.paysecure.ru/type/">
                <bill xsi:type="xsd:string">20150610115527</bill>
                <bill_status xsi:type="xsd:string">Revoked</bill_status>
            </return>
        </ASS-NS:BillResponse>
    </SOAP-ENV:Body>
```

Messages templates

After bill creation, the client will receive the E-mail message that looks like the following:

Subj	**The bill has been invoiced for payment**
Text	Hello Test Testov, We inform you that bill # *1c362fd* on amount of *88.80 BYN* has been invoiced for payment. You can pay it till *25.07.2016 18:00 (MSK)*. To view and pay the bill go by link below. https://payments.belassist.by/bill/paybill.cfm?ID=c2RmZ2UzNXkyNXR5MjU0 Please address *internet-shop #1* at *shop1@testpost.by* in case of any questions concerning about this bill. _______________ With best regards, ASSIST Customer Service Team www.belassist.by

In case when bill was revoked the client will be also notified via E-mail by message looks like:

Subj	**The bill was annulled**
Text	Hello Test Testov, We inform you that bill # *1c362fd* on amount of *88.80 BYN* was annulled. Please address *internet-shop #1* at *shop1@testpost.by* in case of any questions concerning about this bill. _______________ With best regards, ASSIST Customer Service Team www.belassist.by

If necessary, the customer can create his request by a simple reply to the received E-mail. If the merchant specified several addresses as the address for customer requests when connecting to the IPS Assist, then the E-mail will be sent to all these addresses.

Messages' templates can be tuned individually for the enterprise. Please, contact the Assist support team for it: support@belassist.by .

Back on top