Secure Payments via Internet

(EMV 3DS protocol)

Instructions for Virtual Merchants

Integration via CGI / WWW Forms

Identifier: **P-OM-41-EN**
Version: **5.0 / 29.05.2024**
Security Level: **C1 / General use**

Table of contents

Figures

Tables

1. Introduction

1.1 Purpose of the Document

This document aims to provide guidelines for integration of e-merchants with Borica APGW Acquiring and Payment server, in compliance with EMV 3-DSecure requirements. It describes the format and the methods of message exchange between BORICA and merchant using EMV Co compliant protocol.

The document is intended for developers of merchants' sites and includes the requirements and clarifications necessary to connect to BORICA's APGW in order to perform payments via 3-D Secure scheme.

There are additional appendices to the instruciton that describe specific operations. These additional appendices are provided upon request by the institutions that offer the relevant services (use of tokens, etc.)

1.2 Definitions and acronyms

1.2.1 Definitions

Pre-authorization
A process in which an issuer or processor on behalf of the issuer approves a payment transaction.

Pre-Pre-authorization
Transaction performed at two steps. First – APGW registers the request for pre-Pre-authorization. This request approves account balance and blocks the requested amount on the account or cars. Second – Completion, initiated by the merchant. This way the payment is effected. The amount of completion should be lower or equal to the one of pre-quthorisation.

Reversal

A process in which an issuer or processor on behalf of the issuer reverses (cancels) a payment transaction.

Acquirer

A financial institution which is member of an international and/or national card organization and is in contractual relations with a merchant for acceptance (acquiring) of payments by card products of the respective association. Within the 3-D Secure, the acquiring institution or its authorized agent (processor) determins whether the respective merchant is to be included in the scheme of payments through the open network of Internet.

Issuer

A financial institution which is member of an international and/or national card organization and is in contractual relations with a merchant for acceptance (acquiring) of payments by card products of the respective association. Within the 3-D Secure, the acquiring institution or its authorized agent (processor) determins whether the respective merchant is to be included in the scheme of payments through the open network of Internet.

e-Merchant
A legal entity in contractual relations with the acquiring institution for acceptance of cards payments.

MID
Identifier of the Merchant, provided by FI.

TID
Identifier of the Terminal, provided by FI.
N.B.! TID could be different for development and production environment.

EMV® Three-Domain Secure (3DS)
A message protocol, developed by EMVCo., that allows cardholders to be authenticated to card issuers when performing transactions via Internet.

Issuer Domain
Contains the systems and performs the functions related to the issuer and the clients using its services (cardholders).

GUARDTIME
The maximum time limit for completing a transaction request. In the current version of the interface the maximum time limit for completing a transaction request is 15 minutes (900 seconds).

Return URL (BackRef)
Merchant's address where responses from BORICA's payment server are sent.

Merchant Portal
Application where merchants could monitor the transactions performed at their terminals. The addresses are:

Develeopment environment:

https://3dsgate-dev.borica.bg/mwp_cert

Production:

https://3dsgate.borica.bg/mwp/static/

1.2.2 Acronyms

ACS	Access Control Server
APGW	Acquiring and Payment Gateway
API	Application Programming Interface
BIN	Bank Identification number. The first 6 or 8 digits of the PAN unambiguously defining the Issuer of the card
CGI	Common Gateway Interface
DS	Directory Server
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol - application protocol for hypermedia documents exchange, for example HTML
PAN	Primary Account Number
URL	Application protocol for hypermedia documents exchange, for example HTML
JRE	Java Runtime Environment
SSL	Secure Socket Layer

OpenSSL	A free software library, providing a set of cryptographic functions and definitions. https://www.openssl.org
UTF-8	8-bit Unicode Transformation Format – variable-width character encoding used for electronic communication
Keystore	Certificates and private keys storage

2. Specification of the e-merchant interface

2.1 E-Merchant interface to e-Commerce Gateway

The communication and parameters' transfer in the interface are performed via HTML Forms and HTTP POST to APGW.

The communication between the merchant and CGI e-Commerce Gateway includes:

- **Sending of data for initiation of "Sale" or "Pre-Pre-authorization" transactions to the APGW**

  The data in the request is related to the e-merchant (order number, amount, etc.). It is sent to the acquiring and payments server (APGW) of BORICA as a first step of the transaction before the client enters its financial information (card number, validity, etc.) on BORICA's payment page. Data is transferred via HTTP POST from the cardholder's browser to the site of BORICA.

  Each vPOS can work in one currency only. The currency of the request should be the same as the one of the terminal.

- **Receiving result of "Sale" or "Pre-authorization" from the APGW**

  The e-merchant receives a result from "Sale" or "Pre-authorization" (either successful or unsuccessful), after passing all the steps to authenticate the cardholder and authorize the transaction by the Pre-authorization system of the card issuer.

  The data is transmitted by forwarding from the cardholder's browser to the e-merchant 's website via HTTP POST.

  The e-merchant is responsible for verifying the digital signature of the response data in order to validate that the result is signed by BORICA. BORICA's test and production publick keys are published at https://3dsgate-dev.borica.bg/ They are in .pem format or with certificate (.cer).

  Each merchant should assure a procedure for replacement of BORICA's public key.

  The address where the merchant receives APGW's responses (BACKREF) should be preliminarly set in the system for each terminal.

  The e-merchant is responsible for visualizing the result to the cardholder after receiving the response.

- **Receiving information about the transaction's status**

  It is possible (due to the nature of the Internet) that the e-merchant will never receive a "Sale" result or an "Pre-authorization" result of a successfully or unsuccessfully authorized transaction. This can happen if the cardholder inadvertently closes his browser after APGW has sent the result, or due to an Internet connection issue at that time.

By TRTYPE=90 the merchant can check the transaction status for operations performed within previous 24 hours.

- **Completion**

  With “Pre-authorization”, APGW returns the result to the e-merchant, initiating a deferred payment. Upon successful "Pre-authorization" it is necessary for the e-merchant to initiate "Pre-authorization Completion” for the Pre-authorization.

  The amount of completion should be lower or equal to the one of pre-Pre-authorization.

  For each pre-authorisation is possible.

  In case the completion is not successful by some reason a new pre-Pre-authorization should be performed.

- **Reversal of Pre-authorisation**

  e-Merchant can reverse the pre-authorisation, in case it is successfully completed. The reversal could be performed within 30 days after the pre-authorisation. The mechanism of these operations is described in Chapter 4 “Trasaction types”.

  The amount of the reversal should be equal to the one of pre-authorisationd.

  For each successful operation only one reversal can be performed (successful or bob-successful). In case the reversal is not successful, the merchant should contact the acquiring institution.

  Acquiring institution could have additional requirments for these kind of transaction.

- **Reversal of Sale**

  The e-merchant is given the opportunity to cancel the "Sale" or "Pre-authorization" of a successfully completed transaction (Reversal). Cancellation can be made no later than 30 days after the successful transaction. The way this is done is described in Section 4 "Transaction types".

  For each successful operatio only one reversal can be made (successful or non successful).

  Acquirng institution could have additional requirments for these kind of transaction.

  In case of non-successful reversal the amount can be returned to the cardholder through the acquiring institution.

- **Second transaction in case of Soft Decline (RC 65/1A)**

  When a response from the Pre-authorization system is received with RC 65 for Mastercard or 1A for VISA, the terminal must perform second request for the same transaction. This is done automatically and the merchant can see bothh operations – the first-unsuccessfull and the second- with result provided by Issuer’s host.

  In case the second operation is not successful, the transaction ends with RC 1A.

2.2 Message Exchange

The exchange of messages between the e-merchant’s site and the acquiring and payments server of BORICA is run through the cardholder browser via HTTP POST method. Figure 2-1 illustrates the scheme of sending the message from the e-merchant’s server to BORICA’s server.

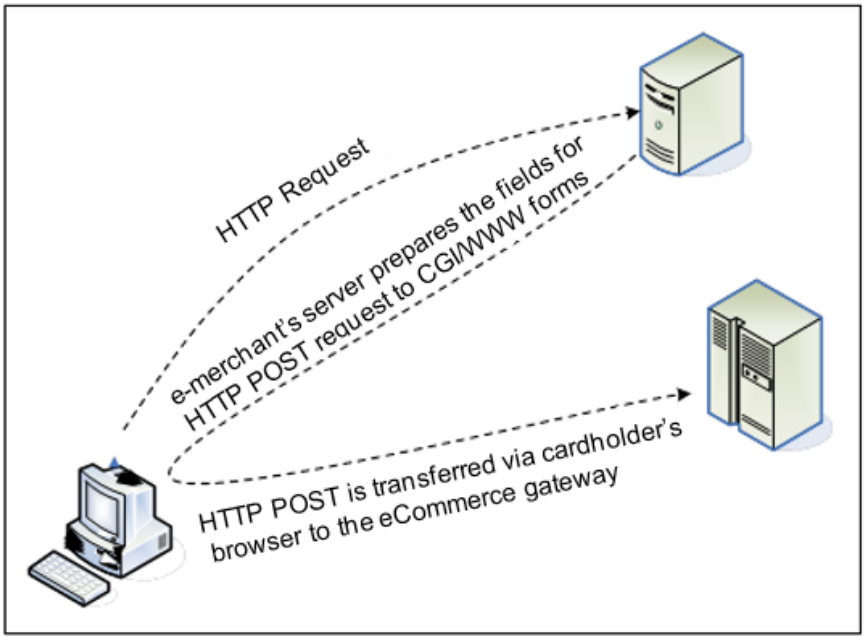


Figure 2-1 HTTP POST message flow

In CGI interface the data transfer from the e-merchant to BORICA is performed via HTML Forms fields using HTTP POST.
The parameters format and names are described in detail in Section 3."Fields in the message for e-merchant communication - APGW ".

When the cardholder would like to approve the payment at merchant's site (by pressing "Pay" button for example), merchant's site creates HTML form with parameters for „Sale" or „Pre-authorisation" and sends it via cardholder's browser to BORICA's site via HTTP POST.

The cardholder's browser establishes SSL/TLS connection with the BORICA site using the server certificate of BORICA, submits the parameters prepared by the e-merchant and initiates the authentication dialogue and Pre-authorization of payment with the cardholder.

Addresses to send the requests:

Test (Development): https://3dsgate-dev.borica.bg/cgi-bin/cgi_link

Production: https://3dsgate.borica.bg/cgi-bin/cgi_link

Requests for all transaction types are processed via the above URLs. No card information is entered on e-merchant's site.

3. Fields in the messages for e-merchant communication - APGW

3.1 Fields in the request from e-merchant to APGW

Parameters sent via HTML Form fields are:

Field	Length	M/O/C	Value
TERMINAL	8	M	Terminal ID Provided by FI
TRTYPE	1-2	M	Allowed values: 1, 12, 21, 22, 24, 90
AMOUNT	4-12	C	Total amount of the order as per ISO_4217 with decimal point separator (e.g. 12.00) *
CURRENCY	3	C	Order currency: 3-character currency code as per ISO 4217* Mandatory if the message contains amount value.
ORDER	6	M	Order number for the e-merchant request. Contains 6 digits, right aligned and supplemented with leading zeros. * **ATTENTION! Must be unique for the terminal within the last 24 hours (e.g. "000123").**
DESC	1-50	C	Order description *
MERCHANT	10	C	Merchant ID Provided by FI
MERCH_NAME	1-80	C	Merchant Name *
MERCH_URL	1-250	O	Merchant's primary web site URL (e.g. webshop.merchant.com)
COUNTRY	2	C	Merchant's terminal 2-character country code as per ISO 3166-1.
MERCH_GMT	3	C	Merchant's UTC/GMT time offset (e.g. +03).
LANG	2	O	Transaction's Language: BG or EN. Default language is BG.
ADDENDUM	5	C	System field with value "AD,TD". Mandatory if field „AD.CUST_BOR_ORDER_ID" is present
AD.CUST_BOR_ORDER_ID	6-22	C	ORDER + 16 characters for merchant order number ** **ATTENTION! Must not include semicolon ";".**
TIMESTAMP	14	C	Transaction date/time in **UTC**: YYYYMMDDHHMMSS. Timestamp difference between TIMESTAMP and current e-Gateway server time must not exceed 15 minutes, otherwise e-Gateway will reject this transaction
TRAN_TRTYPE	1-2	C	Original transaction type in "Transaction Status Check"
RRN	12	C	Retrieval Reference Number (ISO-8583 -1987, field 37). Mandatory for reversal operations.
INT_REF	16	C	E-Commerce gateway internal reference number. Mandatory for reversal operations.
M_INFO	0-35000	M	Optional data set as per EMV 3DS v.2 protocol, Base64 Basic encoded string of JSON-formatted "parameter":"value" data. Mandatory data: - Cardholder name (up to 45 characters, Latin only) - E-mail and/or telephone

		C	Possible conditional values: - For requesting mandatory authentication form the customet Example: { "threeDSRequestorChallengeInd":"04" }
NONCE	32	M	Contains 16 unpredictable random bytes in hexadecimal format. Allowed characters are capital Latin letters A..F and numbers 0..9. **ATTENTION! Must be unique for the terminal within the last 24 hours ВНИМАНИЕ!**
P_SIGN	512	M	Message Authentication Code (MAC). Contains 256 bytes in hexadecimal format. Allowed characters are capital Latin letters A..F and numbers 0..9.

Table 1 Fields used for the request to APGW

∗ Used to display information from the e-merchant to the cardholder on payment page. Cyrillic is supported.

** Used for information to help the cardholder and the e-merchant recognize the payment transferred via financial files. Supports numbers and Latin letters.

N.B.! When Cyrilic is used should be used **encoding UTF-8 only**.

M/O/C:
M – Mandatory
C – Conditional – depending on transaction's type
O – Optional

Table 1 describes all valid fields. Depending on transaction type, some fields are not used.

The parameters from Table 1 are transmitted via HTML Form via HTTP / POST.

IMPORTANT! Section 4 lists the participating fields for each type of transaction. The fields used to form signing string for the request must be present in the request.

Sample POST request for TRTYPE=1 (Sale):

M_INFO value upon submitted:

- **Cardholder name and e-mail**:
  Input data:
  {"email":"user@sample.com","cardholderName":"CARDHOLDER NAME"}
  Encoded data:
  eyJlbWFpbCI6InVzZXJAc2FtcGxlLmNvbSIsImNhcmRob2xkZXJOYW1lIjoiQ0FSREhPTERFUiBOQU1FIn0=

  Cardholder name and telephone
  Input data:
  {"cardholderName":"CARDHOLDER NAME","mobilePhone":{"cc":"359","subscriber":"893999888"}}
  Encoded data:
  eyJjYXJkaG9sZGVyTmFtZSI6IkNBUkRIT0xERVIgTkFNRSIsIm1vYmlsZVBob25lIjp7ImNjIjoiMzU5Iiwic3Vic2NyaWJlciI6Ijg5Mzk5OTg4OCJ9fQ==

- **Cardholder name and telephone + mandatory authentication indicator**
  Input data:
  {"threeDSRequestorChallengeInd":"04","cardholderName":"CARDHOLDER NAME","mobilePhone":{"cc":"359","subscriber":"893999888"}}
  Encoded data:
  eyJ0aHJlZURTUmVxdWVzdG9yQ2hhbGxlbmdlSW5kIjoiMDQiLCJjYXJkaG9sZGVyTmFtZSI6IkNBUkRIT0xERVIgTkFNRSIsIm1vYmlsZVBob25lIjp7ImNjIjoiMzU5Iiwic3Vic2NyaWJlciI6Ijg5Mzk5OTg4OCJ9fQ==

- **Cardholder name, e-mail and telephone + mandatory authentication indicator**
  Input data:
  {"threeDSRequestorChallengeInd":"04","cardholderName":"CARDHOLDER NAME","email":"user@sample.com","mobilePhone":{"cc":"359","subscriber":"893999888"}}
  Encoded data:

  eyJ0aHJlZURTUmVxdWVzdG9yQ2hhbGxlbmdlSW5kIjoiMDQiLCJjYXJkaG9sZGVyTmFtZSI6IkNBUkRIT0xERVIgTkFNRSIsImVtYWlsIjoidXNlckBzYW1wbGUuY29tIiwibW9iaWxlUGhvbmUiOnsiY2MiOiIzNTkiLCJzdWJzY3JpYmVyIjoiODkzOTk5ODg4In19

Sample POST request for “Sale” (TRTYPE=1):

```
<form name="pay" action=https://3dsgate-dev.borica.bg/cgi-bin/cgi_link method="POST">
AMOUNT: <input type="text" name="AMOUNT" size="5" value="80.05" readonly="readonly"/><br>
CURRENCY: <input type="text" name="CURRENCY" size="3" value="BGN" readonly="readonly"/><br>
DESC: <input type="text" name="DESC" size="16" value="Детайли плащане." readonly="readonly"/><br>
TERMINAL: <input type="text" name="TERMINAL" size="8" value="V1800001" readonly="readonly"/><br>
MERCH_NAME: <input type="hidden" name="MERCH_NAME" size="13" value="Магазин цветя"
readonly="readonly"/><br>
MERCHANT: <input type="hidden" name="MERCHANT" size="10" value="1600000001"
readonly="readonly"/><br>
TRTYPE: <input type="hidden" name="TRTYPE" size="1" value="1" readonly="readonly"/><br>
ORDER: <input type="hidden" name="ORDER" size="6" value="155827" readonly="readonly"/><br>
M_INFO: <input type="hidden" name="M_INFO" size="152" value=
"eyJlbWFpbCI6InVzZXJAc2FtcGxlLmNvbSIsImNhcmRob2xkZXJOYW1lIjoiQ0FSREhPTERFUiBOQU1FIiwibW9iaWxlUGhvbmUiOnsiY2MiOiIzNTkiLCJzdWJzY3JpYmVyIjoiODkzOTk5ODg4In19"
readonly="readonly"/><br>
AD.CUST_BOR_ORDER_ID: <input type="hidden" name="AD.CUST_BOR_ORDER_ID" size="12"
value="155827ORDnnn" readonly="readonly"/><br>
COUNTRY: <input type="hidden" name="COUNTRY" size="2" value="BG" readonly="readonly"/><br>
TIMESTAMP: <input type="hidden" name="TIMESTAMP" size="14" value="20240516125932"
readonly="readonly"/><br>
NONCE: <input type="hidden" name="NONCE" size="32" value="DC221A460F21374894BD289AE3B34AB4"
readonly="readonly"/><br>
ADDENDUM: <input type="hidden" name="ADDENDUM" size="5" value="AD,TD" readonly="readonly"/><br>
```

```
P_SIGN:          <input          type="hidden"          name="P_SIGN"          size="512"
value="61F1EDF199D68D0139D1A512D713D27FD55A6C891B598F929A3CE3466D8577F709EAFDCCF8
BA06C8966EF8440B3D189950DE54F8E6786EBB00D9C96BF22B1F1A6D4A5FFA0BEE3F3301331BAB32C6CD58E955B09F4B0
FCE88E053553FED50D0D07AA23F04DF30EA1A0
20E7FAD1ECF5E7D5FC2D01A196FC8CF7C90A06635DD09B8536E7E922F462024CA94228F28F81629FD44760CB17193D25B
F4998E6D183D8A3EDC5F1435238ADC7DA7888C
96A8C6EBB01D292E9B886C0CFF1DB9A3B64D86297D77CE87B68FEFB185B2EBC154DF638F44FA84F1C9CA35A72B0B265E7
EF74CC65287FC3F4743521AFB93F1F1E1D4E67
A1A55C259F5907D7E3FDA74733AF541FE" readonly="readonly"/><br>
<br><INPUT type="submit" name="Submit" value="Approve"></br>
```

The e-merchant can implement additional validations and logic using JavaScript on its page for the input data.

N.B.: All the Merchants (MPIs/ 3DSS) using cookie are advised to adhere/reference to RFC 6265 standards for cookie handling grammar and behavior.

3.2 Fields in the response from APGW to e-merchant

The e-merchant's site receives the result of a transaction request from APGW through the cardholder's browser using the HTML Form and HTTP POST method.

The fields used are the following::

Field	Length	M/O/C	Value
ACTION	1-2	M	E-Gateway action code: 0 – Transaction successfully completed; 1 – Duplicate transaction found; 2 – Transaction declined, original issuer's response is returned 3 – Transaction processing error 6 – Duplicate, declined transaction 7 – Duplicate, authentication error 8 – Duplicate, no response 21 – Soft Decilne
RC	2	M	ISO-8583 Field 39: response code or APGW addition response code
STATUSMSG	1-255	C	Text message corresponding to the response code
TERMINAL	8	M	Echo from the request
TRTYPE	1-2	M	Echo from the request
AMOUNT	4-12	C	Final transaction amount
CURRENCY	3	C	Echo from the request
ORDER	6	M	Echo from the request
LANG	2	O	Echo from the request
TIMESTAMP	14	M	Date/time of the response in UTC: YYYYMMDDHHMMSS
TRAN_DATE	14	C	Date/time of the transaction: YYYYMMDDHHMMSS
TRAN_TRTYPE	1-2	O	Transaction type of the original transaction in "Transaction Status Check"
APPROVAL	6	O	Cardholder's bank approval code (ISO-8583 Field 38). May be empty if not sent by card issuer..

RRN	12	O	Retrieval reference number (ISO-8583 Field 37)
INT_REF	16	M	APGW internal reference
PARES_STATUS	1	C	3-D Secure authentication status
AUTH_STEP_RES	1-32	C	3-D Secure authentication status level
CARDHOLDERINFO	1-128	C	Issuer ACS cardholder information. **ATTENTION! E-merchant must display the value to the cardholder.**
ECI	2	C	Electronic Commerce Indicator
CARD	16-19	C	Masked card number (e.g. „5100XXXXXXXX0022“)
CARD_BRAND	1-4	C	Card’s Brand
NONCE	32	M	Echo from the request.
P_SIGN	512	M	Message Authentication Code (MAC). Contains bytes in hexadecimal format. Allowed characters are capital Latin letters A..F and numbers 0..9.

Table 2 Fields in APGW response

The value of the ACTION field contains the code from the execution of the transaction. All codes different from "0" mark non-successful transactions.

Field RC "Response code" contains a code from the execution of the transaction. All codes different from "00" mark non-successful transactions. Transaction requests declined by APGW have negative RC (e.g. -17 for bad signature). It is possible the same transaction request, but with valid signature, to be posted to APGW later. APGW will process the request and if transaction is authenticated and authorized by the issuer, returned code will be “00”. In case the operation is rejected by the Issuer’s hostq the RC will be positive, different from“00”.

N.B.: Successful transaction is the one for wich RC „**00**” and Action “**0**” are received.

The response for transactions that are not related to the cardholder's browser but are transmitted directly from the e-merchant's website to APGW using GET or POST methods, are in json.

Transaction matching can be done using fields TERMINAL, ORDER and NONCE values.

3.3 Considerations for APGW interface fields

1) AMOUNT contains the amount of the order including the decimal point, for example "10.20" or „0.29“.
2) ORDER field contains digits only
3) TIMESTAMP is in UTC
4) AD.CUST_BOR_ORDER_ID is used to transfer the order number to the e-merchant's Bank in the financial files. The field must contain the value of the field ORDER (6 digits), concatenated with a character string up to 16 characters long. The same string can be used as alpha-numeric e-merchant order number up to 16 characters in length.

N.B.! The field must not contain a semicolon (";") symbol

4. Transaction types

APGW processes transaction requests within the GUARDTIME time slot. If the TIMESTAMP in the request is older than GUARDTIME, the request is rejected. The default value of GUARDTIME is 15 minutes (900 seconds).

N.B.!: For all transaction types - field RFU does not participate in the request or response to/from APGW, but is included in the string for signing with length 1 byte 0x2D (minus sign "-").

4.1 Sale

TRTYPE=1

"Sale" transaction is used for payment of goods and services.

Participating fields: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, DESC, MERCHANT, MERCH_NAME, MERCH_URL, EMAIL, COUNTRY, MERCH_GMT, LANG, ADDENDUM, AD.CUST_BOR_ORDER_ID, TIMESTAMP, M_INFO, NONCE, P_SIGN

Field	M/O	Condition
TERMINAL	M	
TRTYPE	M	
AMOUNT	M	
CURRENCY	M	
ORDER	M	
DESC	M	
MERCHANT	M	
MERCH_NAME	M	
MERCH_URL	O	
EMAIL	O	
COUNTRY	O	
MERCH_GMT	O	
LANG	O	
ADDENDUM	M	
AD.CUST_BOR_ORDER_ID	M	
TIMESTAMP	M	
M_INFO	M C	Mandatory data: - Cardholder name (up to 45 symbols, Latin), - Telephone and/or e-mail To request mandatory cardholder authentication (SCA) the following value is to be provided “eyAidGhyZWVEU1JlcXVlc3RvckNoYWxsZW5nZUluZCI6IjA0IiB9”
NONCE	M	
P_SIGN	M	

Table 3 Fields in „Sale“ request

Fields in request signing string: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU

Fields in response signing string: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.2 Transaction status check

TRTYPE=90

The request can be HTTP GET or POST. Usually, the request is generated by the e-merchant's server and sent directly to APGW, when response is not received for the transaction.

APGW stores transaction data for the last 24 hours. In case APGW finds a record for a transaction, the response contains data about the transaction in json format. If the transaction is not found, the default currency in the response is USD.

In case the response is RC=-40 (Client side transaction form in progress), when the original transaction is older than thhe GUARDTIME, the original transaction is not successful (timeout).

N.B.! ORDER field contains the number of the original operation, and TRAN_TRTYPE indicates the original transaction's type. The rest of the fields should be filled-in according to requirments, descried in Table 1.

N.B.! In case the transaction is reversed – transaction status check should be performed for the original transaction type 22 or 24. The response contains information for the respective transaction (TRAN_TRTYPE parameter).

Some of meanings of ACTION and RC fields in the response:

ACTION	RC	Description
0	00	Transaction is successfully processed. The response contains the original transaction‘s information.
2	Code from the Issuer	Transaction is declined by the issuer. The response contains the original transaction‘s information.
3	-19	Unsuccessful authentication. StatusMsg field contains additional details:
3	-31	Transaction is processed by the Issuer
3	-33	Authentication in progress
3	-39	Request for approval of cardholder
3	-40	Request for approval of operation

Table 4 Transaction status

Meanings of ACTION and RC fields are described in p.3.2

N.B.! Negative values of RC firld could change during transaction processing within the GUARDTIME.

Participatinf fields: TERMINAL, TRTYPE, ORDER, TRAN_TRTYPE, NONCE, P_SIGN

Field	M/O	Condition
TERMINAL	M	
TRTYPE	M	
ORDER	M	
TRAN_TRTYPE	M	
NONCE	M	
P_SIGN	M	

Table 5 Fields in the request for Transaction status check operation

Fields in request sign: TERMINAL, TRTYPE, ORDER, NONCE

Fields in response sign: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.3 Reversal

TRTYPE=24

Reversal request could be HTTP GET или POST. Usually the request is generated by e-merchant's server and is sent directly to APGW.

The reversal is a cancellation of previous Sale operation. The reversed amount could be less or equal to the one of the initial operation. For each successful Sale only one reversal could be processed (successful or not) within 30 days from the initial transactin date.

In case the reversal is not successful, the merchant should contact its Acquiring institution.

The acquirnf institution might have some additional requirments for this type of transactions.

Participating fields: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, DESC, MERCHANT, MERCH_NAME, MERCH_URL, EMAIL, COUNTRY, MERCH_GMT, LANG, ADDENDUM, AD.CUST_BOR_ORDER_ID, TIMESTAMP, RRN, INT_REF, NONCE, P_SIGN

Field	M/O	Condition
TERMINAL	M	
TRTYPE	M	
AMOUNT	M	
CURRENCY	M	
ORDER	**M**	
DESC	M	
MERCHANT	M	
MERCH_NAME	M	
MERCH_URL	O	
EMAIL	O	
COUNTRY	O	
MERCH_GMT	O	
LANG	O	
ADDENDUM	M	
AD.CUST_BOR_ORDER_ID	M	
RRN	**M**	
INT_REF	**M**	

TIMESTAMP	M	
NONCE	M	
P_SIGN	M	

Table 6 Fields for reversal transaction

N.B.! Fields ORDER, RRN and INT_REF contain the values from the original Sale transaction.

Fields in request sign: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU

Fields in response sign: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.4 Pre-authorisation

TRTYPE=12

Pre-authorisation transaction should be followed by Completion operation.

Participating fields: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, DESC, MERCHANT, MERCH_NAME, MERCH_URL, EMAIL, COUNTRY, MERCH_GMT, LANG, ADDENDUM, AD.CUST_BOR_ORDER_ID, TIMESTAMP, M_INFO, NONCE, P_SIGN

Field	M/O	Condition
TERMINAL	M	
TRTYPE	M	
AMOUNT	M	
CURRENCY	M	
ORDER	M	
DESC	M	
MERCHANT	M	
MERCH_NAME	M	
MERCH_URL	O	
EMAIL	O	
COUNTRY	O	
MERCH_GMT	O	
LANG	O	
ADDENDUM	M	
AD.CUST_BOR_ORDER_ID	M	
TIMESTAMP	M	
M_INFO	M C	Mandatory data: - Cardholder name (up to 45 symbols, Latyn), - Telephone and/or e-mail To request mandatory cardholder authentication (SCA) the following value is to be provided “eyAidGhyZWVEU1JlcXVlc3RvckNoYWxsZW5nZUluZCI6IjA0IiB9”
NONCE	M	
P_SIGN	M	

Table 7 Fields for Pre-authorisation operation

Fields in request sign: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU
Fields in response sign: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.5 Compelion

TRTYPE=21

The request could be HTTP GET or POST. Usually the request is generated by e-merchant's server and is sent directly to APGW.

Completion is used to close the Pre-authorisation operation. The amount of Completion should be less or equal to the one of Pre-authorisation. For each successful pre-authorisation only one Completeion could be processed (successful or not) within 30 days from the pre-authorisation.

The accepting institution of the e-merchant may have additional requirements for the execution of this type of transaction.
In case of unsuccessful completion, the scrap dealer should make a new Pre-authorization.

Participating fields: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, DESC, MERCHANT, MERCH_NAME, MERCH_URL, EMAIL, COUNTRY, MERCH_GMT, LANG, ADDENDUM, AD.CUST_BOR_ORDER_ID, TIMESTAMP, RRN, INT_REF, NONCE, P_SIGN

Поле	M/O	Условие
TERMINAL	M	
TRTYPE	M	
AMOUNT	M	
CURRENCY	M	
ORDER	**M**	
DESC	M	
MERCHANT	M	
MERCH_NAME	M	
MERCH_URL	O	
EMAIL	O	
COUNTRY	O	
MERCH_GMT	O	
LANG	O	
ADDENDUM	M	
AD.CUST_BOR_ORDER_ID	M	
RRN	**M**	
INT_REF	**M**	
TIMESTAMP	M	
NONCE	M	
P_SIGN	M	

Table 8 Fields in Completion request

N.B.! Fields ORDER, RRN and INT_REF contain values from the original transaction Pre-authorisation.

Fields in request sign: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU

Fields in response sign: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.6 Reversal of Pre-authorisation

TRTYPE=22

The request could be HTTP GET or POST. Usually the request is generated by the e-merchant's server and is sent directly to APGW.

Transaction type Reversal is used to cancel the previous Pre-authorization.

The reversal's amount should be equal to the one of the initial Pre-authorization. For each operation only one reversal can be performed (successful or not) within 30 days from the original transaction.

In case the reversal is not successful the merchant should contact its acquiring institution.

Merchant's acquiring institution could have additional requirments for this type of operations.

Participating fields: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, DESC, MERCHANT, MERCH_NAME, MERCH_URL, EMAIL, COUNTRY, MERCH_GMT, LANG, ADDENDUM, AD.CUST_BOR_ORDER_ID, TIMESTAMP, RRN, INT_REF, NONCE, P_SIGN

Поле	**M/O**	**Условие**
TERMINAL	M	
TRTYPE	M	
AMOUNT	M	
CURRENCY	M	
ORDER	**M**	
DESC	M	
MERCHANT	M	
MERCH_NAME	M	
MERCH_URL	O	
EMAIL	O	
COUNTRY	O	
MERCH_GMT	O	
LANG	O	
ADDENDUM	M	
AD.CUST_BOR_ORDER_ID	M	
RRN	**M**	
INT_REF	**M**	
TIMESTAMP	M	
NONCE	M	
P_SIGN	M	

Table 9 Fields for reversal of Pre-Pre-authorization

N.B! Fields ORDER, RRN and INT_REF contain values from original Pre-authorization transaction.

Fields in request sign: TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU

Fields in response sign: ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

4.7 Second transaction in case of Soft Decline

TRTYPE=1,12

Upon response received from Pre-authorization host with RC 65 for Mastecard or 1A for VISA, APGW automatically generates a second request to Issuer's ACS for srong customer authentication and a second request for Pre-authorization to the Issuer.

In cae the second operation is not successfulq the transaction ends with RC 1A.

5. Cryptographic operations

Application protocol for connection to BORICA's payment server (APGW) requiers to digitally sign the exchanged messages.

This requires the necessity of basic cryptographic operations.

Each e-merchant signs the requests with its private key and checks BORICA's with APGW public key.

Some exapmles to use OpenSSL for specific cryptographic operations are shown hereafter.

5.1. Digital signature to sign a message

Digital sign assures 3 main goals related to information security – message integrity, authentication of participating parties and and impossibility to deny during data exchange between e-merchant and APGW.

Depending on the message type – request or response the sign applies for part of the fields in Table 1 or Table 2. Depending on transaction type (TRTYPE), additional fields could be included. For each type of transaction exact set of fields, participating in the sign of request and response to/from APGW are described in Chapter 4.

For digitaly sign the requests to APGW each e-merchant uses its own pair of RSA keys, one for development and one for production environment. The algorytm used to generate the key pair should be RSA PKCS#1.

Safekeeping and replacement of private key is merchant's responsability. Private keys should be kept securely and are not to be disclosed to third parties. Private keys should not be sent under no circumstances via e-mail, or other channel since there is a risk of compromise. In such case keys are deactivated and new ones should be generated.

It is strongly recommended to issue a new private key to be generated each time when certificate or pubic key expires.

Code table used by default is UTF-8.

The signing algorithm applies on the symbol string. The e-merchant's private key is used (respectively for test and production environment).

The veriificacion of signature in APGW responses is performed by each e-merchant usind APGW's public key (respectively for test and production environment).

In case of missing field, participating in signature verification, in APGW request, it is replaced by one byte 0x2D (minus sign "-").

In case of missing field, participating in signature verification, in APGW response, it is replaced by one byte 0x2D (minus sign "-").

Fields for the symbol string for MAC_GENERAL:

N	TRTYPE	P_SIGN_FIELDS_REQUEST	P_SIGN_FIELDS_RESPONSE
1	1	TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU
2	12	TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU
3	21	TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU
4	22	TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU
5	24	TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU
6	90	TERMINAL, TRTYPE, ORDER, NONCE	ACTION, RC, APPROVAL, TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, RRN, INT_REF, PARES_STATUS, ECI, TIMESTAMP, NONCE, RFU

Table 10 Fialds for signature signing using MAC_GENERAL scheme, depending on message type

N.B.!: In the present interface version the meaning of field RFU (Reserved for Future Use) in the symbol string is one byte 0x2D (minus sign "-"). Fiald RFU is reserved for future use and is not participating in the request and response to/from APGW.

Appendix 1 contains examples for signing request and validation of responses from APGW on PHP.

Examples of interface developments and additional information could be found at:
https://3dsgate-dev.borica.bg/

N.B.!: TIMESTAMP field is in UTC. The maximum allowed difference between request's TIMESTAMP and current APGW time is 15 minutes. For Bulgaria the offset is "+03" summer time and "+02" winter.

Example PHP:

```
$fldTimeStamp = gmdate('YmdHis');
```

5.2 Generation of private key for message signing using OpenSSL

Test terminal:

openssl genrsa -out privatekeyname_T.key [-aes256] 2048

Production terminal:

openssl genrsa -out privatekeyname_P.key [-aes256] 2048

Remarks:

- privatekeyname_T – name of private key generated for test terminal;
- privatekeyname_P – name of private key generated for production terminal;
- optional parameter -aes256 – used in case the private key is passworf protected;
- the key should be 2048 bits.

Using the above command 2 different keys should be generated: for test and for production terminal. Private keys are used to sign the messages dent to BORICA's payment server. Private keys are generated by the e-merchant and should be kept securely.

N.B.!: It is strongly recommended to protect private keys by password.

5.3 Generating certificate request using OpenSSL

Names of private keys generated on previous step are used in commands for certificate requests - [privatekeyname_T.key и privatekeyname_P.key].

Test terminal:

openssl req -new -key privatekeyname_T.key -out VNNNNNNN_YYYYMMDD_T.csr

Production terminal:

openssl req -new -key privatekeyname_P.key -out VNNNNNNN_YYYYMMDD_P.csr

E-merchant should generate 2 certificate requests and send them to BORICA for signing:

- certificate request for test terminal;
- certificate request for production terminal;

File names are subject to the following convention:

VNNNNNNN_YYYYMMDD_Z, where:
VNNNNNNN – TID – terminal ID (provided by Acquiring institution)
YYYYMMDD – request's date in Year Month Day format
Z – certificate type- T – for test environment, P – for production environment

Each .csr file is archived in .zip named VNNNNNNN_YYYYMMDD_Z.zip where:
VNNNNNNN – TID - terminal ID (provided by Acquiring institution)
YYYYMMDD – request's date in Year Month Day format
Z – certificate type- T – for test environment, P – for production environment

5.4 Mandatory fields for the certificate

Mandatory certificate fields
(using Latin, no special symbols):

- Common name (CN) – domain name (ex.: merchantdomain.bg ; sould not contain http:// or https://)
- Organization Unit Name (OU) – TID
- Organization Name (O) – company's name,
- Locality Name (L) – Lcation (city)
- State or Province Name (ST) – region
- Country Name (C) = BG
- Email Address

Private key and certificate request could be generated at https://3dsgate-dev.borica.bg/generateCSR/

Private keys should be entered in merchant's system. Tets Certificate request (.csr file) is to be uploaded in Merchant Portal – development https://3dsgate-dev.borica.bg/mwp_cert, and Production Certificate request (.csr file) - in MP Prod. https://3dsgate.borica.bg/mwp/static/.

Upload of private key in MP is not allowed

5.5 Conversion of private key and certificate in PKCS12 format with OpenSSL

openssl pkcs12 -export -inkey privatekeyname_Z.key -in VNNNNNNN_YYYYMMDD_Z.cer -out keystore_name.p12

openssl pkcs12 -export -inkey privatekeyname_Z.key -in VNNNNNNN_YYYYMMDD_Z.cer -out keystore_name.pfx

privatekeyname_Z.key is the name of respective private key (generated as per p. 5.2), meanings: T – for test environment, P – for production environment;
VNNNNNNN_YYYYMMDD_Z is the respective certificate's name (received fron BORICA upon requests sent as per p. 5.3), meanings: T – for test environment, P – for production environment.

5.6 Check for correspondence key/ certificate using OpenSSL

The Merchant can check whether the private key and the public one (certificate received from BORICA) are valid pair RSA keys by calculating md5 (check) sum. In case check sums are equal – private and public keys are RSA pair keys.

openssl rsa –noout –modulus –in privatekeyname_Z.key | openssl md5

openssl x509 –noout –modulus –in VNNNNNNN_YYYYMMDD_Z.cer | openssl md5

Where:
- **privatekeyname_Z.key –** private key's name,
- **VNNNNNNN_YYYYMMDD_Z.cer** certificate's name,
- **Z** – certificate type, meanings T – for test environment, P – production environment

5.7 Forming signature for the request to APGW

The request to APGW should be signed by merchant's private keyq respectively for test and production environment.

The folloing table lists request fields participating in the request for transaction type SALE:

Field	Bytes in UTF-8	Meaning
TERMINAL	8	V1800001
TRTYPE	1	1
AMOUNT	4	9.00
CURRENCY	3	BGN
ORDER	6	154744
TIMESTAMP	14	20201012124757
NONCE	32	9EADBD70C0A5AFBAD3DF405902602F79
RFU (Reserved for Future use)	0	-

Table 11 Example for symbol string for signing SALE transaction

The second column shows the number of bytes for the meaning of respective field.

Symbol string is:

8V1800001**1**1**4**9.00**3**BGN**6**154744**14**2020101212475**32**9EADBD70C0A5AFBAD3DF405902602F79-

The length is in green.

Total length of the field in the example is 78 symbols.

For illustration only, the above symbol string, if converted in hexadecimal type is:

38563138303030303131313**4**392E3030**33**42474E**36**313534373434**3134**32303230313031323132343735373**332**39454144424437304330413541464241443344463430353930323630324637392D

N.B.! The order of fields when forming the string for signature should be as shown in Table 11.

5.8 Check of signature in APGW response

The merchant should check signature of APGW response using BORICA's public key, respectively for test and production environment. In case some field is missing in the responceq it is substituted by one byte 0x2D (minus sign "-").

The next Table lists the fields in APGW response, for SALE transaction, along with their lenghts.

Field	Number of bytes in UTF-8	Meaning
ACTION	1	1
RC	2	00
APPROVAL	6	S97539
TERMINAL	8	V1800001
TRTYPE	1	1
AMOUNT	4	9.00
CURRENCY	3	BGN
ORDER	6	154744
RRN	12	028601253152
INT_REF	16	97E2F39EFCA1CAF1
PARES_STATUS	0	
ECI	0	
TIMESTAMP	14	20201012160009
NONCE	32	9EADBD70C0A5AFBAD3DF405902602F79
RFU (Reserved for Future use)	0	-

Table 12 Example for symbol string for checking the response for SALE

The second column shows the number of bytes for the respective field.

11**2**00**6**S975398V180000**1**1**4**9.00**3**BGN**6**154744**12**028601253152**16**97E2F39EFCA1CAF1--**14**20201012160009**32**9EADBD70C0A5AFBAD3DF405902602F79-

Th elenghts are marked in green.

Total length of the field in this example is 124 symbols.

For illustration only, the above symbol string, if converted in hexadecimal type is:

3131**32**3030**36**533937353339**38**5631383030303031**31**31**34**392E3030**33**42474E**36**31353437343**4313232**3032383630313235333135**3136**393745324633394546434131434146312D2D**3134**32303230313031323136303030393**3332**394541444244373043304135414642414433444634303539303236303246373 92D

N.B.! The order of fields when checking the signature should be as shown in Table 12.

6. Examples for transactions

Data below are extracted from different test operations.
Examples of interface developments could be found at:
https://3dsgate-dev.borica.bg/

The screens shown aim to illustrate the representation of information during the transaction. It is possible each real site to look differently.

6.1 SALE

Information from the merchant to APGW:

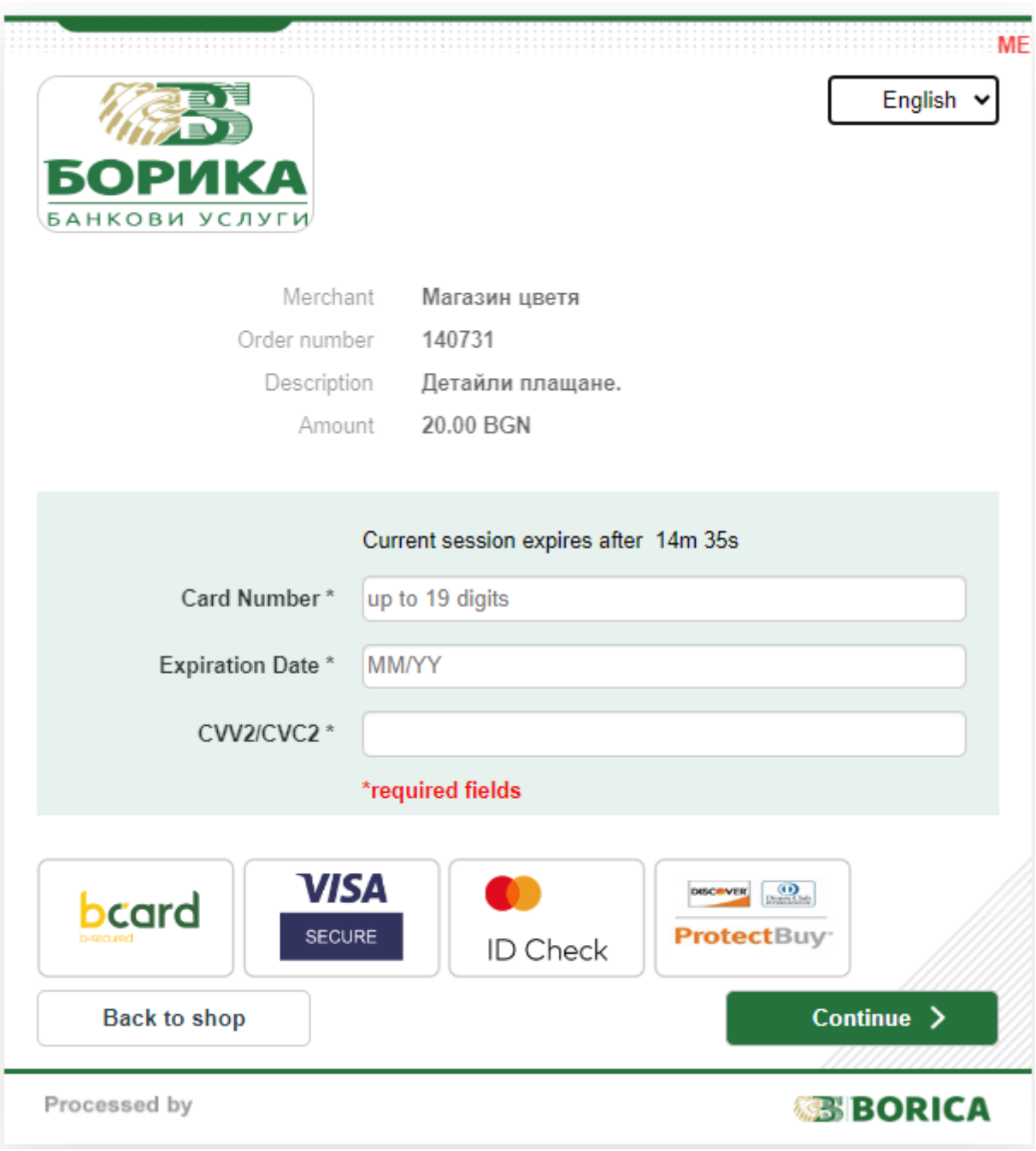
TERMINAL	V1800001
TRTYPE	1
AMOUNT	1.00
CURRENCY	BGN
ORDER	145659
DESC	Детайли плащане.
MERCHANT	1600000001
MERCH_NAME	Мол България
MERCH_URL	http://www.borica.bg
EMAIL	merchant@borica.bg
COUNTRY	BG
MERCH_GMT	+02
ADDENDUM	AD,TD
AD.CUST_BOR_ORDER_ID	145659ORD@<n>
TIMESTAMP	20201013115715
M_INFO	{"email":"user@sample.com","cardholderName":"CARDHOLDER NAME","mobilePhone":{"cc":"359","subscriber":"893999888"}}
NONCE	FC8AC36A9FDADCB6127D273CD15DAEC3
P_SIGN	95B299B9706ED8D9FDA2F3EC3ADCBF0346A1299C512CFB498321DB8AFAE853F6A96BE472B54A75231F894D19F488E2BD3803D893E0924B678BD9777DDF922BCB0BD8F38E887E2FDEF675C428E7C023C420679D93E72A90A51B9B21E2209C5751813754F3ACC30F35BA3E61298D43BFBB2902B59B3B226F71BFA2DB8A17488B42FB60466983B421442DD4C9799C612579DECC32192153B62EF2AF02C24BD3433BE02AE7AB5976C7B769666DE5984293AE1CA814C9FB2E0D2B45FA098F0B08591832AEC8A334C6783A274F4C2D25E1B0296139439D41B313E1CDB4C730DBC2E32812135FE7E7F0CB97E535D1742EBA848B5F6D20259D364B46D9449955CE46B335

MAC (P_SIGN) - Calculation info

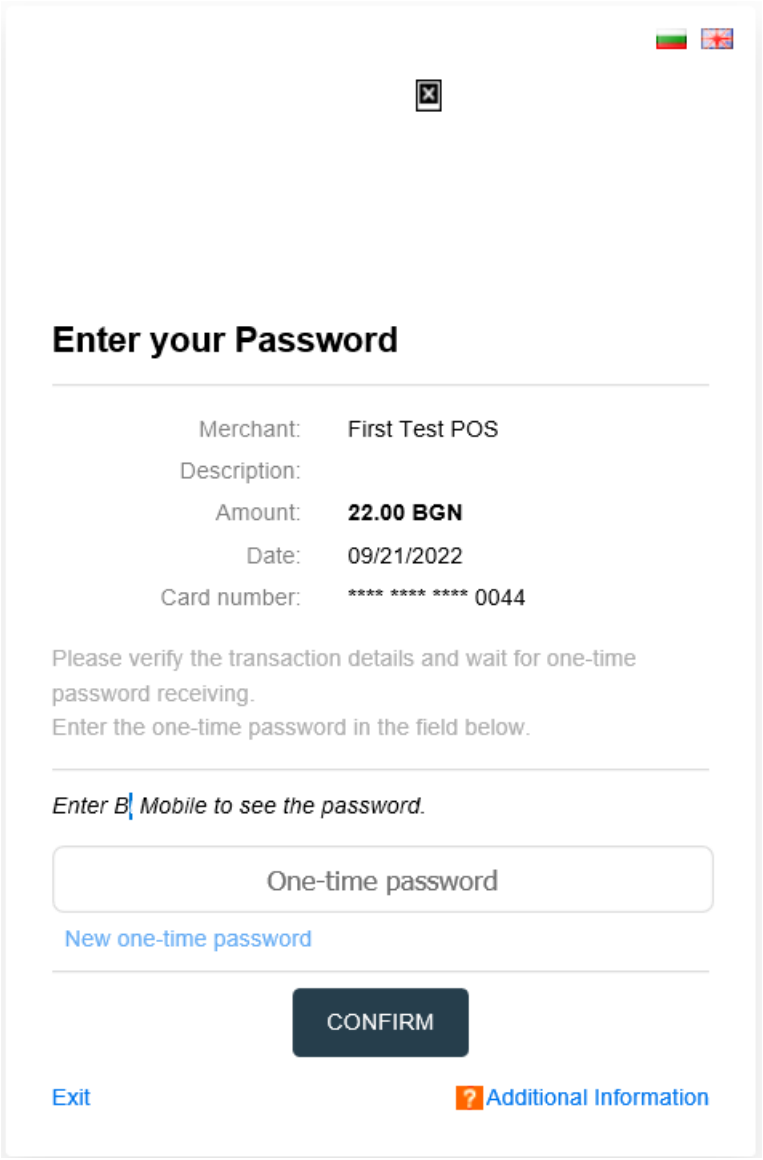
macFields	[TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU]
macSourceValue	8V18000011141.003BGN6145659142020101311571532FC8AC36A9FDADCB6127D273CD15DAEC3-

Table 13 Example for SALE

After pressing Approve button the payment page is visualized:



If the card is registered in ACS for 3-D Secure, a screen for cardholder's authentication via cleint's ACS appears:



Upon completion of the transaction, control is passed to a URL for the e-merchant in APGW database (BACKREF).

The field RC = 00 indicates a transaction approved by the card issuer.

The following table lists all variables and their values for the transaction result.

APGW Response:

Parameter	Length	Value
ACTION	1	0
RC	2	00
STATUSMSG	8	Approved. No errors
TERMINAL	8	V1800001
TRTYPE	1	1
AMOUNT	4	1.00
CURRENCY	3	BGN
ORDER	6	170403
TIMESTAMP	14	20201013140707
TRAN_DATE	14	20201013170707
APPROVAL	6	S19527
RRN	12	028701253242
INT_REF	16	B7A68A9F37E8586E
LANG	0	
PARES_STATUS	0	
AUTH_STEP_RES	7	VERES_N
CARDHOLDERINFO	0	
ECI	2	
CARD	16	5100XXXXXXXX0022
CARD_BRAND	3	MCC
NONCE	32	22EA51788AFE61A9D814B771A8FA6379
P_SIGN	512	31C6507191249D361086E1CA70A2A0374ACF9191D765055E10ACB93D720E934FEBE44E59D41D19C7B976CF358FA572B12EB08556EA602141E983F6FC93F106B0249780C192FAD7BC6411C33E966317804681D692CCDAF42F7494B1B7A7ED8AB23CB8DE5F0621E0C3582671BD222A3E5409538D9BD93F11B150B75D0C59AAC5E77D439FE14A6B494C8FECB1C23867A77D291E34425B5F1A6E9CBA9B92E3BC344E2C9AFAD45E2AE2D1313200A80DE26C2DD870E63AFEADA9EDAEF4DF5B32AD533D68665CB8F7F6E42D8ED7FFE31415FFAED25B3BA159063A9FC542FA958719016697CE9760954A58A2AF077BA049D1DD2216242D80572AA0EA98A39CD7C8DDB5BE

Table 14 “Sale” response example

==== Response signature from ====

macFields = [ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]

macSourceValue = [102006S195278V18000011141.003BGN617040312028701253242 16B7A68A9F37E8586E--1420201013140707322EA51788AFE61A9D814B771A8FA6379-]

Signature = [true]

6.2. “Transaction Status Check” example

The fields with data from the original transaction are in **bold**.

Request from e-merchant to APGW (for SALE transaction)**:**

TERMINAL	V1800001
TRTYPE	90
ORDER	**114233**
TRAN_TRTYPE	**1**
NONCE	622CAAA8BF20C5A21A917DCB8401C336
P_SIGN	5FD6E5A6A0121A599594DB1F0FC96F2CEB4CCC7B3B829E9DBA74E1DC4AF115B774A5460AAA268DB65E04B71C6E9EB6A3F7A820C27D4EA1BC648A19BC97D2577F510F4CDF4BFD6EDA4B8D2B85564791ED6287A08282027099F07166FA8416F123FEEBBC920A33A0ED5964CA02C49A7ED7D5E61F4B5D53CC14DF542BDF4221DCDA22C5864F9F722BF989CB7A2BF2ABE0B76F823561A33F2152772312429204AAB94B58C7AFC82F64D5C20069D4A5B1DF406041CAB77BCCE88C6F84704B2B33AFC82216C2F41B92129D68933CE1C59F87CEAE6B1E8CFBE6DD4CE5898F8FE6453CC7DB7519801FB05BBDE7973E18A86AFF020121B74A65EAD2741BC1D6E39DD42564

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, ORDER, NONCE]
macSourceValue	8V18000012906114233326 22CAAA8BF20C5A21A917DCB8401C336

Table 15 “Transaction Status Check” request example

The result of “Transaction Status Check” is in json format. If the field used to check P_SIGN is missing or has an empty value, it is replaced by one byte 0x2D (minus sign "-") in the response signing string.
Example:
“102006S449738V180000129041.003BGN61203531202870125319516 6AF46A8970774DBB--14202010131524293 25E7EFC5D43E684642F0FB8B7F22167B9-”.

APGW response (“Sale” transaction is successful):

{
"ACTION":"0",
"RC":"00",
"STATUSMSG":"Approved",
"TERMINAL":"V1800001",
"TRTYPE":"90",
"AMOUNT":"1.00",
"CURRENCY":"BGN",
"ORDER":"114233",
"TIMESTAMP":"20201016084515",
"TRAN_DATE":"20201016114310",
"TRAN_TRTYPE":"1",
"APPROVAL":" S78952",
"RRN":"029001254078",
"INT_REF":"4C9B34468610CF9F ",
"PARES_STATUS":"Y",
"AUTH_STEP_RES":"ARES_Y",
"CARDHOLDERINFO":"",
"ECI":"05",
"CARD":"4341XXXXXXXX0044",
"CARD_BRAND":"",
"NONCE":"7A9A2E5CD173AF3F69A87F06E1F602ED",
"P_SIGN":"A20DE81C5723E3A92D8D1B73C7C2B8848A42D3380E9DF9951127E5878AF989E6951F595A52C16CC9B9F690BDC0165DE8E4CF2FA5892A17C5F8026011D604AF5723DF4C35486AA0094C1C23AE9617F8BE2C11F448EA40CDB332EBAB73DE2D33A01AC1BEE83108B788D22D8653F86DFAE8BAEB17048869156D2876FD7F8E232BDB1311D5D4EB63C630EC4941EDBFC70802508F86147714CD7E671014EC8D56882070B6B203FFECE07A67FED6D20C9F4E4637E8EA5B0FE274AD4D8965CB7025BD205F259E41EAF2E48E5566099842B02FB89E7534081CFD4289F6F5F7727DAAB7EBB472FDFD9D091F57616120190732BF635D49EF9519B4CEE26D8DFBB34C2D033B"
}

==== Response signature ====
macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]
macSourceValue =
[102006S789528V180000129041.003BGN611423312029001254078164C9B34468610CF9F1Y2051420201016084515327A9A2E5CD173AF3F69A87F06E1F602ED-]
Signature = [true]

Request from e-merchant to APGW (for Reversal operation)

TERMINAL	V1800001
TRTYPE	90
ORDER	**114233**
TRAN_TRTYPE	**24**
NONCE	B1A1B57F8D66EF6B604690BF7141B53C
P_SIGN	AEC96B3551F4B951E91A5BE6DFD91AD6AF859D4358B7A5D7CD5E8E5B7B4C32E995A6B5FFDBC4265F535D16ED8D591E06DA57E7A05357C93153A13807E2FBA6BB7C9A94AE6B2F2253F9DB8A7D0273AB68B8B9A427814B2646C6585E51396A531BABB3A8EF034496EA0ECEB29379A3E97195FB65DF85B571537620C27FF33483FDD09E8E106EE02FC59B15E70C4D692BD8A3A269DAF24DCBF300B3AB9DA623F789855828AE876CB6304D43027F212EFDB3CD1271A800920725BB3A8A247C84824B468EBF55DDD0540B5E7B6E844BBE28FBA49B62A91BB623A05158DC0D8CD4E6B1FF6BEE0D0EA1012EB04E44E930A3D728F0178BC4458734A3E1D462EB5BEA259E

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, ORDER, NONCE]
macSourceValue	8V1800001290611423332B1A1B57F8D66EF6B604690BF7141B53C

Table 16 “Transaction Status Check” request example

APGW response (“Sale” transaction is not reversed):

{
"ACTION":"3",
"RC":"-24",
"STATUSMSG":"Transaction context mismatch",
"TERMINAL":"V1800001",
"TRTYPE":"90",
"AMOUNT":"",
"CURRENCY":"USD",
"ORDER":"114233",
"TIMESTAMP":"20201016084907",
"TRAN_DATE":"",
"TRAN_TRTYPE":"24",
"APPROVAL":"",
"RRN":"",
"INT_REF":"",
"PARES_STATUS":"",
"AUTH_STEP_RES":"",
"CARDHOLDERINFO":"",
"ECI":"",
"CARD":"",
"CARD_BRAND":"",
"NONCE":"B1A1B57F8D66EF6B604690BF7141B53C",

"P_SIGN":"774F0E62105F5AEED1AED347D81AC12E122423F3E5F0DFBA2DEA3E93D9FC30EFBA9067E6F8A26DA4F44A9CB1B1824A942DA759B051C14CD5D303AA2A11285382C2CFD6B1188ED0DA2E4D1B5E33143DF8A27F0D785749597F7269A40A44113FE5EEF7ACD6D4B0A924053538462BF9F7C58FBD0CB3AC47E61EA039F6A0693B992E1AD0CA278D6B9BC2BA0F3BB1FFDCBCA68D631D7B00B8877004E8C758E335EF3C46E468D9A06C2F94FBF0753FF95A33404FBD8F9BFCB4D60AAA593C5C37AF9BEC3FFCA234B419528A635FCBAA8ED498D1A68834FF71C62286EF5DCC6992EAED703B6AAC262225A655874E8B7277138E68DD8886C44930E7814661B5F9006C0013"
}

==== Response signature ====

macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]
macSourceValue = [133-24-8V1800001290-3USD6114233----142020101608490732B1A1B57F8D66EF6B604690BF7141B53C-]
Signature = [true]

APGW response (“Sale” transaction is reversed successfully):

{
"ACTION":"0",
"RC":"00",
"STATUSMSG":"Approved",
"TERMINAL":"V1800001",
"TRTYPE":"90",
"AMOUNT":"1.00",
"CURRENCY":"BGN",
"ORDER":"114233",
"TIMESTAMP":"20201016085138"
"TRAN_DATE":"20201016115039",
"TRAN_TRTYPE":"24",
"APPROVAL":"S78952",
"RRN":"029001254078",
"INT_REF":"4C9B34468610CF9F",
"PARES_STATUS":"",
"AUTH_STEP_RES":"",
"CARDHOLDERINFO":"",
"ECI":"",
"CARD":"",
"CARD_BRAND":"",
"NONCE":"E8CAC1D2FBE11A899204AED74C02BDEC",
"P_SIGN":"9C22C8E340976C8360B7CB53C5EC90B99BA9A67EE86FE703715766ED3BF8490366C43B579DD1454C0C38B4D31CCD94515EA63AF97FFEB9884234B907B92E4FDF5CF7E806C114C2211BD800E0A659EC35CFD45F0027F05FA66C6F5468982743581416DA42EDC33EDC83537CB57598D527DE193C7BAA360E383CA7172AC0720A50BE2A3530008E8C867427B69CEC9A281907ECE7584BAA49D287BA33F80B49E7857E57509E69CF1F54D83555BF2258F45D36CC4764F9F5803F3D6710FF2F1A82AE4CD345BBB40102563FCA605479759D9E6C1CACBF3A9B1D48BFEC17388261782745CECEE27E3B75A106E0560A2D2403A5EE9DB38932E995D920F38875ABA2D3AF",
}

==== Response signature ====

macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]
macSourceValue = [102006S789528V180000129041.003BGN611423312029001254078164C9B34468610CF9F--14202010160851383 2E8CAC1D2FBE11A899204AED74C02BDEC-]
Signature = [true]

6.3. “Sale Reversal” example

The fields with data from the original transaction are in **bold**.

Request from e-merchant to APGW:

TERMINAL	V1800001
TRTYPE	24
AMOUNT	1.00
CURRENCY	BGN
ORDER	**145659**
DESC	Детайли плащане.
MERCHANT	1600000001
MERCH_NAME	Мол България
MERCH_URL	http://www.borica.bg
EMAIL	merchant@borica.bg
COUNTRY	BG
MERCH_GMT	+02
ADDENDUM	AD,TD
AD.CUST_BOR_ORDER_ID	125353ORD@<n>
TIMESTAMP	20201014095541
RRN	**028701253242**
INT_REF	**B7A68A9F37E8586E**
NONCE	7D51498A3C22B86DD57EFB699A175714
P_SIGN	277DC35B76CD5CAA9BB025A7A5B39EEBF1B3005EA5214F6EB781995FE65418378C5AFA60925977E9A3376D937292C7D57928E3F6B635C78C67411683FB38ABDB876A8EB122196D8534B355A9940934BAD88D2B7FBC25B43CD294059FA6BBB7FDFDC5DBDA0D9306D30F4E387EA879FBC59ED50E64569E3D36A068D6BC6CA57F1FC22F8B0373AF7B1612880648C68E428AF74374AE96A8043C99C99ED21C72B7FFB64EDFCD67BDFCC71B1220FF8CD7A2DFA106EDDD8F5D9B92E4AA8B46FA65F1C3849CE31635FEE43B950240FDE0EB3D638644B9066AE83051F96A34D64C8BF94E92A868C33684DD6A56BD2D26D104EDF8462E2585491BA8B65B8C2B9176C80FC8

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU]
macSourceValue	8V180000122441.003BGN6145659142020101409554132 7D51498A3C22B86DD57EFB699A175714-

Table 17 “Sale Reversal” request example

APGW response:

{
"ACTION": "0",
"RC": "00",
"STATUSMSG":"Approved",
"TERMINAL": "V1800001",
"TRTYPE": "24",
"AMOUNT": "1.00",
"CURRENCY": "BGN",
"ORDER": "145659",
"TIMESTAMP": "20201014100040",
"TRAN_DATE": "20201014125901",
"APPROVAL": "S19527",
"RRN": "028701253242",
"INT_REF": "B7A68A9F37E8586E",
"PARES_STATUS": "",
"AUTH_STEP_RES":"",
"CARDHOLDERINFO":"",
"ECI": "",
"CARD":"",
"CARD_BRAND":"",
"NONCE": "7D51498A3C22B86DD57EFB699A175714",
"P_SIGN":
"4B2C8E02632CA1A753CF9904DF782A2015C8C70546D154842451F5C97ED348D242FBC367CFB91FAAFA53ED2537BF7747CF
C2680E3689AD08AC0D0D97C5FE29B2ED2CF8AA8A12E709021FC9C2A179C993A4D673A80F4C27A76D4141DC85D394BBCCA
1977196042D81AEA907B77B507F95FA4210B13E65D68965294110E483B42D3E1E27FFC06F566A2741BA48FD97092B20896CF8
C66523E92AA1AD2D43CDEDFB21DA875E06581D94B51375FCBC772B93EA91C191DF9BE4C531D5D5FD9E9FE5F8E840B464B
DA150D1AC00D28F58750E0C45F4C62BB8D13A5311E59F8201CCDA601AD47526ED542535E428ED77DBD194E4E87876A270A
7E743873F191639D2DDD7"
}

==== Response signature ====

macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,
NONCE, RFU]

macSourceValue = [102006S195278V180000122441.003BGN6145659120287012532421 6B7A68A9F37E8586E--
1420201014100040327D51498A3C22B86DD57EFB699A175714-]

Signature = [true]

Result RC=00 indicates successful transaction reversal.

6.4. “Pre-authorization” example

Request from e-merchant to APGW:

TERMINAL	V1800001
TRTYPE	12
AMOUNT	3.00
CURRENCY	BGN
ORDER	170000
DESC	Детайли плащане.
MERCHANT	1600000001
MERCH_NAME	Мол България
MERCH_URL	http://www.borica.bg
EMAIL	merchant@borica.bg
COUNTRY	BG
MERCH_GMT	+02
ADDENDUM	AD,TD
M_INFO	{"email":"user@sample.com","cardholderName":"CARDHOLDER NAME","mobilePhone":{"cc":"359","subscriber":"893999888"}}
AD.CUST_BOR_ORDER_ID	170000ORD@<n>
TIMESTAMP	20201012140015
NONCE	C3ACF912658C0A2310EA5AAAF739E627
P_SIGN	A76449365B63104B514683D2C02F47C6EBA202394C60732821A4A7EA43E7B73204C60023B8739B7B91E27F1E0E5CE18B8C1C116408A41D90AC70A392CEF58990DD505AF87E71370D345C295C92F9E03F9F379984BFDE292131B0BAADF19F86398DE18673989F65A2D035B67A05F3114B0D0E8FCA527F513FE27AEBE63F66A8C4C1A5C36F16F4CA0B8B82C0F0E75FFEE2DD6C7139E430F08AD847145AF282B8970CDBC7D3CB8AC22CF7D730C6486C9E10E3925FB4CF9353750907FEE948940194FB702D075DA222F1C7C52C4CDCD86D8893B937B3CAA68372CAD0706A1F20F2E8AD7A4A0C3E8E54815CF6E45AE155A21AECCE77382743E9241E36B76145B7AABC

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU]
macSourceValue	8V180000121243.003BGN6170000142020101214001532C3ACF912658C0A2310EA5AAAF739E627-

Table 18 “Pre-authorization” request example

APGW response:

Parameter	Length	Value
ACTION	1	2
RC	2	05
STATUSMSG	8	Transaction declined
TERMINAL	8	V1800001
TRTYPE	2	12
AMOUNT	4	3.00
CURRENCY	3	BGN
ORDER	6	170000
TIMESTAMP	14	20201012140349
TRAN_DATE	14	20201012170349
APPROVAL	0	
RRN	12	028601253175
INT_REF	16	04F45801DAF13E22
LANG	0	
PARES_STATUS	1	Y
AUTH_STEP_RES	7	PARES_Y
CARDHOLDERINFO	0	
ECI	2	05
CARD	16	4341XXXXXXXX0044
CARD_BRAND	4	VISA
NONCE	32	C3ACF912658C0A2310EA5AAAF739E627
P_SIGN	512	95F5FFF8779932EC04CFE19CC1F75AF01CA5050E8AED8222DA9B5E16ADDBABB6FC51B0FB5501C82FAE2919345F92961E8631CD5A8807DD907E4A32B34B47B4F783EF99C3A4F37B7AB6726DE79FEF0E6E55A5F467ABA82DB3E3C0A8AC09A1E1D7F0D67A83418DC1DF5D362C94774467FA5656F7827C469C307743E93C73DB434940B002E02B0EE2FBC8A8ADB33CC69F3DF6C6D0E69F5042D5C171C840CA296928BEBD79DB9F3D3D2428730C1BEA2261C80DB1A0511687A5D77F242CBE42B204B57B6BDC7F31DDF6027D55E9CE584B101DF5520DD26A399C6D05759C1651B320C176CA206AA775DAA1D7288C60DEB12508DE2DF49A2F308BB28059EEA8FEC3BE0C

Table 19 “Pre-authorization” response example

==== Response signature from ====
macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,
PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]
macSourceValue = [12205-
3BGN43.008V18000012126170000120286012531751420201012140349 1604F45801DAF13E22
1Y20532C3ACF912658C0A2310EA5AAAF739E627-]
Signature = [true]

6.5. “Pre-authorization Completion” example

The fields with data from the original transaction are in **bold**.

Request from e-merchant to APGW:

TERMINAL	V1800001
TRTYPE	21
AMOUNT	**3.00**
CURRENCY	BGN
ORDER	**162021**
DESC	Детайли плащане.
MERCHANT	1600000001
MERCH_NAME	Мол България
MERCH_URL	http://www.borica.bg
EMAIL	merchant@borica.bg
COUNTRY	BG
MERCH_GMT	+02
ADDENDUM	AD,TD
AD.CUST_BOR_ORDER_ID	170075ORD@<n>
TIMESTAMP	20201012141516
RRN	**028601253167**
INT_REF	**92339532D5866339**
NONCE	CCF64A57E0B9E35D2E01DF4A3805DC58
P_SIGN	724DEA99E6AD3D1E1692FBE24A44F805581F176F14A8853BA9F7A8389DFFF7C10CF01C0E11FFF755503C1716552BE47B9DB383CCDFBD2B087A0D4C23DE70C4D9A2B7FF8FFA16BFE26ACF335472B2208EE6BD82DEC94DE854D141F9B30B697801629F676F0433D656E93A6450FA2435C57C1FA572BF6C84F079D3D1DC842D6E8CF7F55F9A3DDB03E07218CF986D16B08DEFCA8687142B625A714D1223B7613CA48C615DF70E56D6CC26B2EF50E07223FE246E9A21D1ED88CD746B760DEA17EF0ED10F18E7BA5F31F886917E08909AD347828F6D8C7FB957DE211E888C3A43F013391A82FC66025633E68C4BD59AF1BEF2C18651BD25FD14DC769DDB2DD3A854E7

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU]
macSourceValue	8V180000122141.003BGN6162021142020101214151632CCF64A57E0B9E35D2E01DF4A3805DC58-

Table 20 “Pre-authorization Completion” request example

APGW response:

{
"ACTION": "3",
"RC": "-20",
"STATUSMSG":"Invalid amount",
"TERMINAL": "V1800001",
"TRTYPE": "21",
"AMOUNT": "1.00",
"CURRENCY": "BGN",
"ORDER": "162021",
"TIMESTAMP": "20201013174253",
"TRAN_DATE": "20201013204253",
"APPROVAL": "",
"RRN": "028601253167",
"INT_REF": "92339532D5866339",
"PARES_STATUS": "",
"AUTH_STEP_RES":"",
"CARDHOLDERINFO":"",
"ECI": "",
"CARD":"",
"CARD_BRAND":"",
"NONCE": "CCF64A57E0B9E35D2E01DF4A3805DC58",
"P_SIGN":
"B2F33F1BE13EDAD498E67A01720AFABD93454C1506038F374EA7B771039C15B6A7C24B2FB9EBA7FEFDE49052118561A09D3D9CFEC98D3A17A8058725EF2E9909C8EF5DDD499B8CBCF5606770588B110B18A1014636F8B6A7CE9F17A3023B6499602A8BE53D3E83FC0FAD97D61B0DCD0DC2C3FBE6600B4B91A8576C34F058FEF80254F4E089567C154EDA67DD6CB997425251C6E4EA4A8531EC1724CA7AC8C9BE11438EBF86CE2B486326EAC03AF8005C443F1B32690B8031774903F847499C1F6080F626EDD5568A41341F70546F90DF67F8980BD3F391D33928554B62A4744A2B331C3350AAE64D0DE3801FE40B73DD89A772D5093D502035AE90D081A85CE8"
}

==== Response signature ====
macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE, RFU]
macSourceValue = [133-20-8V180000122141.003BGN6162021120286012531671692339532D5866339--142020101317425332CCF64A57E0B9E35D2E01DF4A3805DC58-]
Signature = [true]

6.6 “Pre-authorization Reversal” example

The fields with data from the original transaction are in **bold**.

Request from e-merchant to APGW:

TERMINAL	V1800001
TRTYPE	22
AMOUNT	1.00
CURRENCY	BGN
ORDER	**170000**
DESC	Детайли плащане.
MERCHANT	1600000001
MERCH_NAME	Мол България
MERCH_URL	http://www.borica.bg
EMAIL	merchant@borica.bg
COUNTRY	BG
MERCH_GMT	+02
ADDENDUM	AD,TD
AD.CUST_BOR_ORDER_ID	095949ORDnnn
TIMESTAMP	20201014070415
RRN	**028601253175**
INT_REF	**04F45801DAF13E22**
NONCE	D1AA7234EF80331750C61FCCDCE7C5C7
P_SIGN	4C25AC3904F371D1767AD8D75A66A0A997C8EA70C5B0524611484ECB766583F55EBB1C65306348B6FCA16E75A99815DFC32A87FB5383264C780D30E3507C26E4ECF07B50636141E6AB205338BBE340301233F116C6A4947BA565C8C1C754FC81AFEFF68ADF4B30BD7FA3CA2D0114762AE796C6F6C55EB9862AC159079D3ADFDFD262201BE74C416633A19272146A0B13D78CA6E55D6AEAA62F22AACAA617C85192AE417E445D01DFE1F06C713B35D58DF09B5ABA08CBAFA8F3D106E3699D1356A82FEFA400981A055196F906F27AD400BB34C3CA5C648A0F1A8DC47D642295736A39418B37C8FBB4596939376D170D89016D4DC97D8FD6607B2B6E68158C4438

MAC (P_SIGN) - Calculation info

macFields	[TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU]
macSourceValue	8V180000122241.003BGN6170000142020101407041532D1AA7234EF80331750C61FCCDCE7C5C7-

Table 21 “Pre-authorization Reversal” request example

APGW response:

{
"ACTION": "2",
"RC": "95",
"STATUSMSG":"Invalid amount",
"TERMINAL": "V1800001",
"TRTYPE": "22",
"AMOUNT": "1.00",
"CURRENCY": "BGN",
"ORDER": "170000",
"TIMESTAMP": "20201014070617",
"TRAN_DATE": "20201014100617",
"APPROVAL": "",
"RRN": "028601253175",
"INT_REF": "04F45801DAF13E22",
"PARES_STATUS": "",
"AUTH_STEP_RES":"",
"CARDHOLDERINFO":"",
"ECI": "",
"CARD":"",
"CARD_BRAND":"",
"NONCE": "D1AA7234EF80331750C61FCCDCE7C5C7",
"P_SIGN":
"885457783119E64E93D346C38D1050D5A848B97FB8319874CAE1BAB898D6E53B818E2FC83C96C754983B9B0C727FC25BB30
A67455DAA8CF67A5DE9086DE0A96F10FAEE8F7A8D27A9B9FEC69F956DC95E250D970FE380D65F8A99B1115B9B289E2C633
D6CB993246B383A6CC133233F9A14C9EEA554832AD58368893212CCFECDD8268498BF0B307BD414805DA7D23D1B297250B
3AE3CF9164256387E4BF4C386424886BC18B33B43808CECC436F2EE2C4A4114B8609D2D60E836DDA6B82D0BB5CFED1FC85
81418EE4FFAA34828B94B384CF2F22B043894666E13B3BA429FEFD9FAC1D67614927AB11B86141F69DBD2365E868F1B3BA2
50199C1CE4D016EF59F0"
}

==== Response signature ====
macFields =
[ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,
NONCE, RFU]
macSourceValue = [12295-8V180000122241.003BGN617000012028601253175160 4F45801DAF13E22--
142020101407061732D1AA7234EF80331750C61FCCDCE7C5C7-]
Signature = [true]

7. Test cards

The following test cards should be used for testing:

7.6 Cards with fixed response code

Тип на карта	Номер на карта (PAN)
VISA	4341792000000044
MC	5100789999999895

Table 22 Test cards

For testing purposes card details should be entered:

- Expiry – random future date (MMYY format)
- CVV/CVC – random 3 digits
- OTP (if requested) - 111111

In case test amount is ends by .65 – returned RC is 65/1A
In case test amount is 1234.56 for transaction with VISA the APGW response contains value in CARDHOLDERINFO field

8. Error codes used by APGW

The following table lists the commonly used APGW processing error codes (RC field):

RC	Description
-1	A mandatory request field is not filled in
-2	CGI request validation failed
-3	Acquirer host (TS) does not respond or wrong format of e-gateway response template file
-4	No connection to the acquirer host (TS)
-5	The acquirer host (TS) connection failed during transaction processing
-6	e-Gateway configuration error
-7	The acquirer host (TS) response is invalid, e.g. mandatory fields missing
-10	Error in the "Amount" request field
-11	Error in the "Currency" request field
-12	Error in the "Merchant ID" request field
-13	The referrer IP address (usually the merchant's IP) is not the one expected
-15	Error in the "RRN" request field
-16	Another transaction is being performed on the terminal
-17	The terminal is denied access to e-Gateway
-19	Error in the authentication information request or authentication failed.
-20	The permitted time interval (15 min by default) between the transaction timestamp request field and the e-Gateway time was exceeded
-21	The transaction has already been executed
-22	Transaction contains invalid authentication information
-23	Invalid transaction context
-24	Transaction context data mismatch
-25	Transaction confirmation state was canceled by user
-26	Invalid action BIN
-27	Invalid merchant name
-28	Invalid incoming addendum(s)
-29	Invalid/duplicate authentication reference
-30	Transaction was declined as fraud
-31	Transaction already in progress
-32	Duplicate declined transaction
-33	Customer authentication by random amount or verify one-time code in progress
-40	Client side transaction form in progress

Table 23 Additional Response codes, used by APGW

The following table lists the most commonly used error codes when processing a transaction by the issuer over ISO-8583 protocol (RC field)

RC	Description
00	Successfully completed
01	Refer to card issuer
04	PICK UP
05	Do not Honour
06	Error
12	Invalid transaction
13	Invalid amount
14	No such card
15	No such issuer
17	Customer cancellation
30	Format error
35	Pick-up, card acceptor contact acquirer
36	Pick up, card restricted
37	Pick up, call acquirer security
38	Pick up, Allowable PIN tries exceeded
39	No credit account
40	Requested function not supported
41	Pick up, lost card
42	No universal account
43	Pick up, stolen card
54	Expired card / target
55	Incorrect PIN
56	No card record
57	Transaction not permitted to cardholder
58	Transaction not permitted to terminal
59	Suspected fraud
85	No reason to decline
88	Cryptographic failure
89	Authentication failure
91	Issuer or switch is inoperative
95	Reconcile error / Auth Not found
96	System Malfunction

Table 24 Card Issuer processing response codes

9. Appendix 1:

9.6 Example for request signing in PHP:

```
<?php
//Borica Sign Data, private key without password
//execute in https://wtools.io/php-sandbox

//Private key (privatekeyname.key)
$priv_key = '-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQC5z1/LHY1GcX9f
vMOBZPx3edgmqFkPd7eV136Nog9+VeM4UMfg22d64LAwpRHdfFigTPkc9leR68xT
JXGeiiGJSaG+Vb9oUK3yb9W7YMhk1vJy4p2oyo77Sirki4bhh8RPlVWAqeVUgEL/
f5ZuZSNzB2cFkUOknbRwM/j98fft4lgZN/nYkYjW22UaPA7ULEBMxmQUKrJKi04S
PVIG1iKzLh3jVyRsxi+giFrlQ+/jVWA0wJm8B25jsRcwObjL6+MczutVKmaNjaVy
FNkBtLOWSCf4A6i4xOfafWoEx4tEa4Dl5PTqQI4PBvH6SW3lKulfNpa5m1wvnlA3
hFy9lfUPAgMBAAECggEBAJz/stI9yxQ9bEGpjovzlzsgcdngHzhpkG6EocLsryix
S4dXAjxlRp9V4KmJoHnDymLQByFIqJ98XK3YkpNB5apJ0+euLkfm+8NAaZik404J
LNyTzGFFneCIP4vStQo1HFM8ODG53DM1GOcnmCT5QiW8mHjk0AH02vR/haCU5kdR
qQeMBnGAuqOcO3T7QcuK2AMO7BoGrkq0+V58DyCdf1UpeLoi71HCdBpj8FPHcU0H
ScsPUrWXSkJSVj7R68AUtI4Sss3CEk7DbbSLcW1DfmX6esujM/fx1SLc9Bue4Ipa
0ec7wKvvMbLap0gWOOxZGRtxS9ALJ3T75AOjDx38q6ECgYEA86cPMYUePy/l9CEu
F2fsr0LnpB3clwEhhMeIIjKMCTVnPlHMy8Sm9WrRdErKMpsIbOWgeIqUaPPCt3NZ
FTgJnIJnFR4KI5qPOb3ZRA9OI6eliVvdgxe7e/bHe74b/v/uE6378ddtniHCIvgk
6Oi9/luImv3kYzX1pmJr/8VZQdECgYEAwznqU1QiPaUFtZwkb3hIoTcIYlKwIbmP
3HYdsS2p20WHh3XJC9nojABIBgJJYKdACzQyIy1FJJ3ga0fgkSZ5KL1LXmclXXL6
lzdQNIyF/boRP+XC7fB9MNwIClqJCnmciKWE4xCt9GgEiLnJDYnOhGiQ50BuxJFn
RU6RxjAPoN8CgYEA3LmYr/mx/vf7T3/Ha3jAF716b1iF/14M6WaA0soLxkPUtcYQ
yv/paCZOfRVLuzBrH4ueJdUUwUciPGKIbwqG2nvfumeuM7bOzTZJXrimxvIWqirI
rvuO4qwa5uTAI+/h034n4VyRd1GJt3gop75Ab+6oABDFF4NIeGRtBhXX2CECgYB9
/QRCHouyaKsUlt3UqjWFBPT+LwrH2O8EgZ2L2EJD5c0fGF5UrZ4rq4rPhe5A1+62
zEqG9RIoHVLVKR+9zKxoJDFdjQdKFYeuyt2R7BYUtl2ndOmlkIvaWkU+GUtTXUQt
01O9DeiVUAONEQi1GfgS70CEXMqfRI725UuiMSYE1QJ/EkPO8VPbWf+BgKovYFf1
AsStOfMMvrgVn8e/vZmWluaGb40L34Dxuvv3YRk1EsQFirGZ5XDDCm5r8H1IY8Ne
PXnxLP2opIuch1JHQdebFZqN1C68pX6hopEixOmwShhaNXNJ5RN8c9q+4NXIu73n
lKFJBDsxoMVB/VEoVeQEMg==
-----END PRIVATE KEY-----';
//Private key password. Leave empty if there is no password.
$priv_key_password = '';

//Data you want to sign
//MAC_GENERAL = TERMINAL, TRTYPE, AMOUNT, CURRENCY, ORDER, TIMESTAMP, NONCE, RFU
$data = '8V18000011141.003BGN611392014202010130839232D41AAAFC7F8119A3BB7C4868E0B256F9-';

$pkeyid = openssl_get_privatekey($priv_key,$priv_key_password);
echo 'Private Key Result: '.$pkeyid.PHP_EOL;
echo 'Data: '.$data.PHP_EOL;

openssl_sign($data,$signature,$pkeyid,OPENSSL_ALGO_SHA256);
openssl_free_key($pkeyid);

echo PHP_EOL;
echo 'P_SIGN = '.strtoupper(bin2hex($signature));
?>
```

9.7 Example for response validation PHP:

```
<?php
//Borica Verify Signature in Response
//execute in https://wtools.io/php-sandbox

//Certificate containing the public key (MPI_OW_APGW_B-Trust.cer)
$pub_key = '-----BEGIN CERTIFICATE-----
MIIGWjCCBEKgAwIBAgIIQSHpHDZ7ASAwDQYJKoZIhvcNAQELBQAwgYoxCzAJBgNVBAYTAkJHMRgw
FgYDVQRhDA9OVFJCRy0yMDEyMzA0MjYxIDAeBgNVBAoMF0JPUklDQSAtIEJBTktTRVJWSUNFIEFE
MRAwDgYDVQQLDAdCLVRydXN0MS0wKwYDVQQDDCRCLVRydXN0IFRFU1QgT3BlcmF0aW9uYWwgQWR2
YW5jZWQgQ0EwHhcNMjAwOTEwMDg0NzU5WhcNMjMwOTEwMDg0NzU5WjBkMRQwEgYDVQQDDAtNUEkg
T1cgQVBHVzELMAkGA1UECwwCSVMxEjAQBgNVBAoMCUJvcmljYSBBRDEOMAwGA1UECAwFU29maWEx
DjAMBgNVBAcMBVNvZmlhMQswCQYDVQQGEwJCRzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoC
ggEBAMmtJ1gcFkdfY/wfEk3IbqAA1dveXj9J3dCNyliHoooj1ePsX86jlYLijrdPOgayESwH01OO
nVEbcF9z2qoicH12vJaa9ZEFgqkB+qv55erfQOTjgVhd+KRb8YES+uEGkIFE8D/peLMeKeiRSleN
corRa4J1ms/V/2Oklxg0xSnEXw8tRa0U2OoPlEwCbT01DgPMoud5EitpTvD9/gc69aWgVS477Erf
ro+CW89bLGNiHh6mmZt71uIXugNtGf2RhP59fmEKBKj+DSF1QI65SVvv2eYb6JBlhHX+hZss/oAN
xvqYFSG4k6L1tkoDwctB+q7p1EbWEuqDNxYT0RidkLkCAwEAAaOCAecwggHjMB0GA1UdDgQWBBTT
nQwEEjMqWryNqt8onGmGk6nm4DAfBgNVHSMEGDAWgBT1J8z325solCubZvApcg6KPWLcmDAgBgNV
HRIEGTAXhhVodHRwOi8vd3d3LmItdHJ1c3QuYmcwCQYDVR0TBAIwADBNBgNVHSAERjBEMEIGDCsG
AQQB+3YBBwEEAjAyMDAGCCsGAQUFBwIBFiRodHRwOi8vd3d3LmItdHJ1c3Qub3JnL2RvY3VtZW50
cy9jcHMwDgYDVR0PAQH/BAQDAgOoMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjBUBgNV
HR8ETTBLMEmgR6BFhkNodHRwOi8vY3JsdGVzdC5iLXRydXN0Lm9yZy9yZXBvc2l0b3J5L0ItVHJ1
c3RUZXN0T3BlcmF0aW9uYWxBQ0EuY3JsMIGHBggrBgEFBQcBAQR7MHkwJwYIKwYBBQUHMAGGG2h0
dHA6Ly9vY3NwdGVzdC5iLXRydXN0Lm9yZzBOBggrBgEFBQcwAoZCaHR0cDovL2NhdGVzdC5iLXRy
dXN0Lm9yZy9yZXBvc2l0b3J5L0ItVHJ1c3RUZXN0T3BlcmF0aW9uYWxBQ0EuY2VyMBYGA1UdEQQP
MA2CC01QSSBPVyBBUEdXMA0GCSqGSIb3DQEBCwUAA4ICAQAUJfDjTROuVORLojCzVQdppoiPs3hX
Ra/9MaNIUP5xII0AamWmN7bTDQpnNfw5tlo8DPSBIMfP+5xJyfMTHAi43i+7vf1t1ZucEbVJ73FF
zdzZQaxw9NY0n0IBBz8WEnkaGewh45aQ6XMgNe5xcKbtP2vqq+qZiy0eyIHJwaQORKyZ9+jBlnVo
ZdzUbDrrSEMka98IQ52XO8EPbCmB/GhJlZ991yNo5/PVsFxT9sjG3VGm+sStD3G7+pjX+HsHLn65
gwWq2oRiQqe62W/HSNb5dnIWqIJldT4Zd0Ar97hQwU1ZQVnmL5ZjswsjafI7B/0N4U5QzbOvWX1W
oDXCCqmXAoTP1DDEWJ0vmvVDHGrrC0rIbluBdzQEK/D1f3A1jlCzQPyKOwUuafLlpCX17b09Zwxi
45prDk/LBqE6CI6CM+8nF0QyN3Th+r2IqUuhGpfLApGlp6sJvJdAhnqX1VCGJCdozIhzrEJ4oha3
/+HijQl+vUaYevk1d/EipZNHU1gkcocrj2qmTMOKzEw9zDs5jVSgtBZTUF5ORwUNiTXj7EZUnQUC
wANFl8k0EcWPhkU5L7v9/9rcGkMcm0S3bM5rbKksabvq01cvxkepS5qqvbxgugci/8sPCXMAThCK
eiJHilEt1uns+tFA+7RSVFKOpf07g3DBGYf5P8qKLQCFMg==
-----END CERTIFICATE-----';

//Public key (MPI_OW_APGW_B-Trust_pubkey.pem)
/*$pub_key = '-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAya0nWBwWR19j/B8STchu
oADV295eP0nd0I3KWIeiiiPV4+xfzqOVguKOt086BrIRLAfTU46dURtwX3PaqiJw
fXa8lpr1kQWCqQH6q/nl6t9A5OOBWF34pFvxgRL64QaQgUTwP+l4sx4p6JFKV41y
itFrgnWaz9X/Y6SXGDTFKcRfDy1FrRTY6g+UTAJtPTUOA8yi53kSK2lO8P3+Bzr1
paBVLjvsSt+uj4Jbz1ssY2IeHqaZm3vW4he6A20Z/ZGE/n1+YQoEqP4NIXVAjrlJ
W+/Z5hvokGWEdf6Fmyz+gA3G+pgVIbiTovW2SgPBy0H6runURtYS6oM3FhPRGJ2Q
uQIDAQAB
-----END PUBLIC KEY-----';*/

//Data you want to verify (signed message)
//MAC_GENERAL =
ACTION,RC,APPROVAL,TERMINAL,TRTYPE,AMOUNT,CURRENCY,ORDER,RRN,INT_REF,PARES_STATUS,ECI,TIMESTAMP,NONCE
$data = '112006S975398V18000011149.003BGN615474412028601253152169 7E2F39EFCA1CAF1--
14202010121600093 29EADBD70C0A5AFBAD3DF405902602F79';

//P_SIGN in hex from mpi_sign.php
$p_sign =
hex2bin('6FF21243639A23946393023839C0B549C6794C516E4C077F65DD476B700C0A53A9A23F1517B9F8F955C4E8E519CFF1C9428B32F0259E8EA2284B244B39AA8E4E4A251D840479CB3DDB988F25674D1BEB97A814DB04E846FC9795058E2BDC3A511CA503F15C71BD3F1687FF15FE9F8CA393555286CEB4A3B722683E1FFD7C30A6ED19C6EDB7D40A6356B12BD4C010DD43D596753CC6BA52523EC5DB4E0BC48B8A99DDE2D1B946D504EA3A692C3E56DA3941E83F226EEEC109DAB36C3FEE70C89E2E54000E62AC53DB43B72E75597DA735CF513BFFD8D4A61F5468C8A77C9704E9B9BD8AB5167BA1DAD0898CAF7BED831C7786F8E75100FB179657B05CC4EDA87E');


if (strpos($pub_key, 'CERTIFICATE') !== false) {
    $pkeyid = openssl_get_publickey($pub_key);
} else {
    $pkeyid = $pub_key;
}
echo 'Public Key Result: '.$pkeyid.PHP_EOL;
echo 'Data: '.$data.PHP_EOL;

//verify signature
```

```php
$result = openssl_verify($data,$p_sign,$pkeyid,OPENSSL_ALGO_SHA256);
if (strpos($pub_key, 'CERTIFICATE') !== false) {
openssl_free_key($pkeyid);
}

echo PHP_EOL;
echo 'Result = '.$result.' ';
// 1- OK, 0 - Error
if ($result == 1) {
    echo 'Valid';
} elseif ($result == 0) {
    echo 'Invalid';
} else {
    echo 'Error: '.openssl_error_string();
}
?>
```

10. Appendix 2:

10.6 AUTH_STEP_RES Values:

AUTH_STEP_RES field contains the value received from Directory Server (Visa, MasterCard), showing the cardholder's authentication level:

AUTH_STEP_RES	CARD 3DS VERSION	AUTHENTICATION RESULT	DESCRIPTION
ARES_Y	2	Authentication Successful (Frictionless)	Successful frictionless transaction without SCA authentication.
ARES_N	2	Authentication Declined (Frictionless)	Cardholder is not enrolled / Frictionless authentication declined.
ARES_R	2	Authentication Rejected (Frictionless)	The issuer rejects authentication (e.g. Cardholder is blocked/closed).
RREQ_Y	2	Authentication Successful (SCA)	The issuer has authenticated the cardholder using SCA (e.g. verification of password or other identity information).
RREQ_N	2	Authentication Failed (SCA)	The cardholder's password (or other authentication information) failed validation, thus, the issuer is not able to authenticate the cardholder. The following are reasons to fail an authentication: Cardholder fails to correctly enter the authentication information within the issuer-defined number of entries (possible indication of fraudulent user). Cardholder "cancels" authentication page (possible indication of a fraudulent user).
RREQ_U	2	Authentication Could Not Be Performed (SCA)	The issuer ACS is not able to complete the authentication request – possible reasons include: ACS not able to handle authentication request message ACS is not able to establish an SSL session with cardholder browser System failure that prevents proper processing of the authentication request

Table 25 Cardholder's authentication level

10.7 ECI (Electronic Commerce Indicator) values

ECI (Electronic Commerce Indicator) field contains the value received from Directory Server (Visa, MasterCard), showing the result from cardholder's authentication

VISA	
ECI Value	**Definition**
05	Both cardholder and card issuing bank are 3D enabled. 3D card authentication is successful
06	Either cardholder or card issuing bank is not 3D enrolled. 3D card authentication is unsuccessful, in sample situations as: 1. 3D cardholder not enrolled 2. Card issuing bank is not 3D Secure ready
07	Authentication is unsuccessful or not attempted. The credit card is either a non-3D card or card issuing bank does not handle it as a 3D transaction

Table 26 Cardholder's authentication result (Visa, Diners, Bcard)

MasterCard	
ECI Value	**Definition**
00	Authentication is unsuccessful or not attempted. The credit card is either a non-3D card or card issuing bank does not handle it as a 3D transaction
01	Either cardholder or card issuing bank is not 3D enrolled. 3D card authentication is unsuccessful, in sample situations as: 1. 3D Cardholder not enrolled 2. Card issuing bank is not 3D Secure ready
02	Both cardholder and card issuing bank are 3D enabled. 3D card authentication is successful

Table 27 Cardholder's authentication result (Mastercard)

For domestic transactions the values for VISA are used

10.8 FAQ:

10.8.1 What is the meaning of Response codes? Final and not final codes:

Response codesq used by APGW are described in Chapter 9.
To take into consideration:

- **RC=00** – Successful operation, provided that both criteria are met - RC=00 and Action=0. This result is final. Possible values for Action are described in Table 2.
- **Negative codes** – based on authentication message processing. It is possible cahnes in negative codes to occur during the GUARDTIME (15 min.) of transaction. An example for change the codes is when the cardholred press “Back” button of its browser and aa error message is generated; in case after that the cardholder continues with transaction confirming – it might be successfully completed. It is recommended to use Transaction status check operation (TRTYPE=90) when a negative RC is received. **The result received for Transaction status check operation after more than 16 minutes (the GUARDTIME + 1 min. buffer) is final.**
- **Positive codes, different from 00** – final, based on Pre-authorization message processing, respectively the response fron the Issuer’s host.
- No need to check transaction status after final result has been received. Even, for some reason different code is received – it should be ignored.

10.8.2 Common error codes in APGW responses:

- -17 – error in POST request. Possible reasons:
  - Incorrect signature due to incorrect symbol string used. The merchant should check the signing string.
  - Incorrect signature due to incorrect private key used. The merchant should check the correspondence between the private and public keyq according to p. 5.6.
  - Incorrect signing scheme used – the Acquiring institution should indicate the signing scheme used for each terminal.
  - TIMESTAMP of the request is older than 15 мин. (UTC). In this case field STATUSMSG contains value "Expired transaction".
  - APGW process at the same moment a request with the same data for the respective terminal.
- -19 – cardholder’s authentication error:
  - The card is not registered for 3D Secure Payments.
  - Expired / Not updated static password.
  - Wrong static or one time password.
  - Cardholder has cancelled the payment when authentication page has been visualised.
- 58 – the terminal is not activated. Acquiring institution should process Initialize & Enable steps.

10.8.3 Message “Missing BACKREF parameter, using default”

APGW sends data for response to URL set in APGW data base for each terminal. To be checked with Acquiring institution

10.8.4 No response is received – possible reasons:

- The cardholder has closed the browser of payment page and the transaction status remains „Transaction form in progress'" for the next 24 hours. No answer will be send.
- Incorrect BACKREF URL set in DB.

10.8.5 How the transactions are shown in Merchant Portal?

The following sections are available:

- Authentications – all operations for which APGW has processed the POST request.
- Pending – operations for wich the status is still not final, but Pre-authorization step has been completed.
- Rejected – declined by Pre-authorization host (positive RC).
- Transactions – all successful operations.

10.8.6 Now to identify in Merchant Portal the Order for which payment has been made:

- In Authentications – for each transaction ORDER is shown. The value in Order ID column could be used to link the request sent and the data in merchant's system.

- B tab Tagged Data for each transaction values for AD.CUST_BOR_ORDER ID and DESC could be seen.

N.B.! The value in AD.CUST_BOR_ORDER_ID (containing ORDER value) is transferred via financial files to the Acquiring institution and could be seen in financial statement. This value could be used for for accounting reconciliation.