BluePay Post bp10emu

API Transaction Processing

Reference Guide

April 2025

© 2024-2025 Fiserv, Inc. or its affiliates. Fiserv is a trademark of Fiserv, Inc., registered or used in the United States and foreign countries, and may or may not be registered in your country. All trademarks, service marks and trade names referenced in this material are the property of their respective owners.

http://www.fiserv.com

This document is classified as Fiserv Public.

Content

About this Document **5**
Intended Audience 5
Assistance & Feedback 5
Overview **6**
URL 6
INput Fields **7**
MICRO DEPOSITS 19
STORED CREDENTIALS 19
LEVEL 2 (Optional) 21
LEVEL 3 (Optional) 21
LODGING (Optional) 31
VEHICLE RENTAL (Optional) 34
REDIRECT BY POST 37
ADDITIONAL MERCHANT-SPECIFIC DATA 38
MERCHANT-DEFINED FIELD VALIDATORS 39
TRANSACTION TIMESTAMP 40
SWIPE OPTIONS 41
Method 1 (Primary) 41
Method 2 (Unencrypted) 41
Method 3 (Encrypted only, Magtek Dynamag HID) 41
EMV (IDTech Augusta S) 42
TRANSARMOR TOKENS 42
TAMPER PROOF SEAL 43
Captcha 45
DEPRECATED OPTIONS 46
OUTPUT FIELDS **47**
RESPONSERVERSION NULL 47

RESPONSERVERSION 1 & 2 50
RESPONSERVERSION 3 58
RESPONSERVERSION 4 64
RESPONSERVERSION 5 64
RESPONSERVERSION 6 64
RESPONSERVERSION 7 65
RESPONSERVERSION 8 65
RESPONSERVERSION 9 66
RESPONSERVERSION 10 67
RESPONSERVERSION 11 67
FORMATS 68
INPUT FORMAT 68
OUTPUT FORMAT. 68
TAMPER_PROOF_SEAL & BP_STAMP 69
TPS_HASH_TYPE 70
Calculating the TAMPER_PROOF_SEAL 71
STEP 1: Concatenate the TPS_DEF Value 71
STEP 2: Perform the Hash Type 72
Calculating the BP_STAMP 72
STEP 1: Concatenate the BP_STAMP_DEF Value 73
STEP 2: Perform the BP_STAMP 73
How Rebilling Works 74
Rebilling Examples: 75
Static HTML Example and TAMPER_PROOF_SEAL 78
Appendix I - Level 3 Measurement Units 82
Appendix II - Vehicle Rental Vehicle Types 84
Appendix III - Credit Card Address Verification Service (AVS) Response Codes 86
Appendix IV - BINDATA Values 88

Appendix V - Fraud Filter Codes **101**

Revision History **102**

About this Document

This documentation provides technical guidance to process transactions on a BluePay Payment Gateway. The system described in the document may be subject to some minor changes.

Intended Audience

This document is written for merchants, partners, and developers who will be responsible for integrating payment processing functionality with the BluePay Payment Gateway. This document provides an understanding of transaction processing using the BluePay Payment Gateway API.

Assistance & Feedback

Use the following contact information for help with the BluePay Payment Gateway integration or to provide feedback on this document.

Support Level	Contact Details
BluePay Integration Support Team	bluepay-integration@fiserv.com

Support hours are Monday through Friday 8:00am to 5:00pm (CST UTC-6).

Overview

BluePay Post (bp10emu) is our primary transaction processing API. It allows you to process transactions using the HTTP POST format with name=value pairs to the BluePay gateway and receive responses in the same format.

URL

The API endpoint is as follows:

```
Sample: https://secure.bluepay.com/interfaces/bp10emu
```

INPUT FIELDS

The following input fields are available.

MISSING_URL / APPROVED_URL / DECLINED_URL	
Required:	Optional
Description:	When using the API's redirect functionality, the payer will be redirected to the URL that matches the transaction result. If no URLs are supplied the payer will be redirected to a generic result page.

RESPONSEVERSION	
Required:	Optional
Description:	Closely following the guidelines mentioned in this document will make your system future-compatible with any updates to the BluePay System. If you are confident that you have followed these guidelines, then you may safely set this to an arbitrarily high value (like '9999') and automatically reap the benefits of any future updates. If you are not so certain, set it to the highest actual version available at the time of integration. SeeOutput Fields section for the list of output fields available for each RESPONSEVERSION
Example:	Null, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 or 11

MERCHANT	
Required:	Unless PLATFORM_MERCHANT_ID is included
Description:	Your 12-digit BluePay Gateway Account ID.

PLATFORM_MERCHANT_ID	
Required:	Unless MERCHANT is included
Description:	A unique ID that a First Data processing platform assigns to the merchant.
Length/Type:	20/N

To view this field value in the output, set the RESPONSEVERSION field to 11 or above.

PAYMENT_TYPE	
Required:	Optional

PAYMENT_TYPE	
Description:	Provides the payment type of the transaction. • CREDIT = Credit Card (Default) • ACH = Automated Clearing House/E-check • SEPA = Single Euro Payments Area, electronic transfers in the European Union

CC_NUM	
Required:	Yes, when PAYMENT_TYPE=CREDIT and a master ID (RRNO) is not used. No, for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	Provides the Credit Card Number. If F_TRANSARMOR = 1, send the TransArmor Token in place of the credit card number.

TRANSACTION_TYPE	
Required:	Yes
Description:	Provides the transaction type of the transaction. • **AUTH** = Reserve funds on a customer's card. No funds are transferred. • **SALE** = Make a sale. Funds are transferred. • **CAPTURE** = Capture a previous AUTH. Funds are transferred. • **REFUND** = Preferred method to reverse a previous SALE. Converts to a VOID if previous Sale is unsettled. Funds are transferred. ACH requires DOC_TYPE PPD or CCD. DOC_TYPE=WEB cannot be used on REFUND transactions. If refunding a DOC_TYPE=**WEB** transaction change DOC_TYPE to PPD. • **CREDIT** = Issue credit to account. ACH requires DOC_TYPE PPD or CCD. • **REBCANCEL** = Cancel a rebilling sequence. • **UPDATE** = Change a transaction amount or add a tip. Only available when using the First Data Omaha platform. • **VOID** = Prevents a previous SALE or REFUND from settling. Can only be performed prior to the previous transaction being settled.

CVCCVV2	
Required:	Optional
Description:	Three or four digit validation code from a credit card. On the back for Visa, Mastercard & Discover. On the front for Amex.

CC_EXPIRES	
Required:	Yes when PAYMENT_TYPE=CREDIT and a master ID (RRNO) is not used. No for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	The expiration date in MMYY format. Alternatively, the 2-digit Month and 2-digit Year can be sent separately as CC_EXPIRES_MONTH and CC_EXPIRES_YEAR.

CC_EXPIRES_MONTH	
Required:	Yes when PAYMENT_TYPE=CREDIT and a master ID (RRNO) is not used. No for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	The expiration month in MM format.

CC_EXPIRES_YEAR	
Required:	Yes when PAYMENT_TYPE=CREDIT and a master ID (RRNO) is not used. No for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	The expiration month in YY format.

ACH_ROUTING	
Required:	Yes, when PAYMENT_TYPE=ACH and a master ID (RRNO) is not used. No for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	Contains the nine-digit bank routing ("ABA") number for the customer's bank account.

ACH_ACCOUNT	
Required:	Yes, when PAYMENT_TYPE=ACH and a master ID (RRNO) is not used. No for other PAYMENT_TYPE values or a master ID (RRNO) is used.
Description:	Contains the customer's bank account number.

ACH_ACCOUNT_TYPE	
Required:	The ACH_ACCOUNT_TYPE defaults to ‘C’ if the value is not provided it is Optional.
Description:	Set to 'C' for a checking account, 'S' for a savings account or 'G' for general ledger.
Default Value:	'C' if not set.

ACH_SAME_DAY_FUNDING	
Required:	Optional

ACH_SAME_DAY_FUNDING	
Description:	Flag to determine if the current transaction should be processed as a Same-Day ACH transaction. Same-Day ACH funding requires approval before it can be enabled and is not available for all banks. For Same-Day funding, the transaction must be processed prior to the final Same-Day cutoff.
Valid Value:	• 1 (True) same-day funding • 0 (False) standard funding, '0', is a default value if not set
To view this field value in the output, set the RESPONSEVERSION field to 9 or above.	

IS_CORPORATE	
Required:	Optional, when PAYMENT_TYPE=CREDIT. No for other PAYMENT_TYPE values.
Description:	Set IS_CORPORATE to 1 to indicate this is a transaction for a company instead of an individual.

COMPANY_NAME	
Required:	Optional
Description:	COMPANY_NAME contains the name of the company. Optional unless IS_CORPORATE is set to 1.

BIC	
Required:	Yes, if PAYMENT_TYPE is SEPA.
Description:	Contains the bank ID of the customer's bank.

DOC_TYPE	
Required:	Optional, when PAYMENT_TYPE=ACH. No for other PAYMENT_TYPE values.

DOC_TYPE	
Description:	Documentation Type also known as Standard Entry Class (SEC) Codes. Can be one of the following: • **PPD (Prearranged Payment and Deposit):** Indicates there is a signed agreement on file for the customer. • **CCD (Corporate Credit or Debit Entry):** Indicates there is a company-signed agreement on file for the customer. • **CPA Code**: The "DOC_TYPE" parameter holds the "CPA Code" when the Payment Type is "EFT" for Canadian EFT transactions. DOC_TYPE is blank or a 3 digit CPA Code when the PAYMENT_TYPE = EFT • **WEB (Internet Initiated/Mobile Entry):** Indicates the customer has agreed to the electronic funds transfer via an internet-based or electronic form. This is the default DOC_TYPE, if the DOC_TYPE is not set. • **TEL (Telephone-Initiated Entry):** Indicates there is a recorded telephone call on file with the customer verbally agreeing to be charged. • **ARC (Accounts Receivable Entry):** Indicates that a consumer check was received in the mail or at a drop box location and converted to an ACH transfer. • **BOC (Back Office Conversion Entry):** Indicates that a consumer check was received in person from the consumer and later converted to an ACH transfer. • **POP (Point of Purchase Entry):** Indicates that a consumer check was received in person from the consumer and was converted to an ACH transfer while the consumer was present.

IBAN	
Required:	Yes, if PAYMENT_TYPE is SEPA.
Description:	Contains the bank ID associated with the customer's bank account.

MANDATE_ID	
Required:	Yes, if PAYMENT_TYPE is SEPA.
Description:	Merchant supplied Mandate ID number.

MANDATE_DATE	
Required:	Yes, if PAYMENT_TYPE is SEPA.
Description:	Merchant supplied date that Mandate was issued in YYYY-MM-DD format.

AMOUNT	
Required:	Yes, if TRANSACTION_TYPE is AUTH/SALE; Optional: for TRANSACTION_TYPE of REFUND
Description:	The amount of funds to be transferred by the transaction.

ORDER_ID	
Required:	Optional
Description:	Unique 128 Characters identification number that is associated with the transaction.

ACH_DESCRIPTION	
Required:	Optional, when PAYMENT_TYPE=ACH. No for other PAYMENT_TYPE values.
Description:	Defaults to value configured on the gateway account if not provided. • Ten-character alphanumeric field that is used to identify the type of transaction being performed. • The ach_description must be ACCTVERIFY for a Micro-Deposit. See MICRO-DEPOSITS below for further information. • ACH_DESCRIPTION can contain up to 15 alphanumeric characters when PAYMENT_TYPE is EFT • When MODE=TEST and PAYMENT_TYPE=ACH, ACH_DESCRIPTION can be used to trigger specific ACH transaction results. ACH Account Validation must be enabled on the gateway account to use the ACH Account Validation test actions. • Use PAYROLL for DOC_TYPE=PPD when TRANSACTION_TYPE=CREDIT for the payment of wages, salaries, and similar types of compensation. • Use PURCHASE for e-commerce purchases. An e-commerce purchase is the TRANSACTION_TYPE=SALE that a consumer authorizes for the online purchase of goods, including recurring online purchases previously authorized. the e-commerce purchase uses the DOC_TYPE=WEB, except the rule on Standing Authorization permits to use the DOC_TYPE=PPD or TEL.

<table>
<tr><th colspan="2">ACH_DESCRIPTION</th></tr>
<tr><td>Example:</td><td>When ACH Description is BPTESTVRB, then the BluePay test validation result indicates that account validation for this transaction was bypassed.
<table>
<tr><th>ACH_Description</th><th>Result</th></tr>
<tr><td>blank/null</td><td>Result=APPROVED VALIDATION_RESULT=(Not Returned) No Validation Performed (no validation fee billed)</td></tr>
<tr><td>BPTESTREJ</td><td>Result=DECLINED VALIDATION_RESULT=(Not Returned) MESSAGE=Account Previously Rejected: R00 (no validation fee charged)</td></tr>
<tr><td>BPTESTINSF</td><td>Result=DECLINED VALIDATION_RESULTT=(Not Returned) MESSAGE=Too Many Attempts to Resubmit Account with Insufficient Funds (no validation fee charged)</td></tr>
<tr><td>BPTESTVR15</td><td>Result=DECLINED VALIDATION_RESULT=15 MESSAGE=Bank Account Failed Validation (validation fee charged for LIVE transactions)</td></tr>
<tr><td>BPTESTVRR</td><td>Result=DECLINED VALIDATION_RESULT=R MESSAGE=Bank Account Previously Failed Validation (no validation fee charged)</td></tr>
<tr><td>BPTESTVR20</td><td>Result=APPROVED VALIDATION_RESULT=20 (Unknown bank account but with a valid format) MESSAGE=App ACH Sale (validation fee charged for LIVE transactions)</td></tr>
<tr><td>BPTESTVR25</td><td>Result=APPROVED VALIDATION_RESULT=25 (Unknown bank account) MESSAGE=App ACH Sale (validation fee charged for LIVE transactions)</td></tr>
<tr><td>BPTESTVR35</td><td>Result=APPROVED VALIDATION_RESULT=35 (Bank account found but pending transaction settlement) MESSAGE=App ACH Sale (validation fee charged for LIVE transactions)</td></tr>
<tr><td>BPTESTVR45</td><td>Result=APPROVED VALIDATION_RESULT=45 (Known good bank account) MESSAGE=App ACH Sale (validation fee charged for LIVE transactions)</td></tr>
<tr><td>BPTESTVRE</td><td>Result=APPROVED VALIDATION_RESULT=E (Error, Bank account validation failed) MESSAGE=App ACH Sale (no validation fee charged)</td></tr>
<tr><td>BPTESTVRB</td><td>Result=APPROVED VALIDATION_RESULT=B (Account Validation Bypassed) MESSAGE=App ACH Sale (no validation fee charged)</td></tr>
</table></td></tr>
</table>

AMOUNT_TIP	
Required:	Optional

AMOUNT_TIP	
Description:	• The amount of funds added to a transaction as a gratuity. • The AMOUNT_TIP value should be included within the AMOUNTvalue. AMOUNT = amount to charge + AMOUNT_TIP

AMOUNT_FOOD	
Required:	Optional
Description:	If AMOUNT_FOOD supplied AMOUNT value must be modified to include this value.

AMOUNT_MISC	
Required:	Optional
Description:	If AMOUNT_MISC supplied AMOUNT value must be modified to include this value.

COMMENT	
Required:	Optional
Description:	Comment value to be stored on the transaction record.

MODE	
Required:	Optional
Description:	• "LIVE" - To run a production transaction that moves funds. • "TEST" - To run a simulated test transaction. This is the default mode.

- If the MODE is not set and a master transaction is used, then the master transaction's mode will be used on the new transaction.
- If the MODE is anything other than "LIVE" and there is no master transaction, the new transaction will be processed using MODE of "TEST".

REBILLING	
Required:	Yes, for rebillings
Description:	• 1 for rebilling. • 0 or absent for regular non-rebilling.

REB_FIRST_DATE	
Required:	Yes, for rebillings

REB_FIRST_DATE	
Description:	The date of the first rebilling. See "How Rebilling Works" section for additional information. The valid formats are: • "YYYY-MM-DD HH:MM:SS" (Hours, minutes, and seconds are optional). • "XX UNITS" Number of time units from the transaction date and time that the first rebilling should be executed(MINUTE, HOUR, DAY, MONTH or YEAR).
Example:	1 MONTH, 10 DAYS, 1 YEAR

REB_EXPR	
Required:	Yes, for rebillings
Description:	The period of time in-between rebillings. "XX UNITS" Relative date as explained below. Format for REB_EXPR expressions is: "XXX DAY" or "XXX MONTH" or "XXX YEAR" or "XXX HOUR" or "XXX MINUTE"

Where XXX is any valid integer. For example, "10 MINUTE" represents, of course, 10 minutes.

REB_CYCLES	
Required:	Optional
Description:	Number of times to rebill. Don't send or set to "" for infinite rebillings (or until canceled).

REB_AMOUNT	
Required:	Optional
Description:	Amount to rebill. Defaults to amount of transaction if not provided.

REB_IS_CREDIT	
Required:	Optional for rebillings
Description:	Field to indicate whether the rebill transaction is a SALE or CREDIT.
Valid Value:	• 1: For CREDIT • 0: For SALE (default)

F_REBILLING	
Required:	Optional
Description:	Rebilling flag. Only used for non-BluePay generated rebilling to identify the transaction as a rebilling. Set value to 1 for rebill transaction.

RRNO	
Required:	Yes, for TRANSACTION_TYPE of CAPTURE, REFUND,VOID and REBCANCEL
Description:	Reference Routing Number or Transaction ID associated with the Transaction.When RRNO is used in transaction request then it is a Transaction ID of the previous transaction, which serves as parent id to the new transaction. Following RRNO types are available as the different transaction type: • When capturing a previous authorization transaction using TRANSACTION_TYPE = CAPTURE, the RRNO is the transaction ID of the authorization being captured. • When refunding a previous transaction using TRANSACTION_TYPE = REFUND, the RRNO is the transaction ID of the transaction being refunded. • When canceling a rebilling schedule using TRANSACTION_TYPE = REBCANCEL, the RRNO is the transaction ID of the template transaction from which payment information is copied.
Any information that is not supplied in the transaction request will be copied from the details of parent transaction. All the subsequent transactions are linked to their parent transactions.	

CUST_TOKEN	
Required:	Optional
Description:	The Customer Token whose stored payment information should be used for this transaction. Any parameters not sent will be filled out referring the last transaction that used this Customer Token.

NEW_CUST_TOKEN	
Required:	Optional
Description:	The new Customer Token that will store this customer's payment information. Must be 6-16 alphanumeric characters. Cannot include the values of NAME1, NAME2, or COMPANY_NAME. Cannot include the last 4 digits of the customer's payment account number. Must be unique to this account.

CUSTOM_ID	
Required:	Optional
Description:	Up to 16 characters of your own information.

CUSTOM_ID2	
Required:	Optional
Description:	Up to 64 characters of your own information.

CUSTOM_CODE	
Required:	Optional
Description:	Up to 16 characters. Customer or Purchase Order number.

STATEMENT_DBA	
Required:	Optional
Description:	The merchant's DBA Descriptor that will appear on a cardholders statement (25 characters) not available on all networks.

STATEMENT_PHONE	
Required:	Optional
Description:	The merchant's phone number that will appear on a cardholders statement (10 characters) not available on all networks.

DUPLICATE_OVERRIDE	
Required:	Optional
Description:	Set to 1 to turn off duplicate scrubbing for a transaction. Set to 0 or leave blank to process with duplicate scrubbing.

CUSTOMER_IP	
Required:	Optional **Highly Recommended**
Description:	The IP address of the customer's computer. When the customer does not post their payment directly to the BluePay Gateway setting this value is required to take full advantage of BluePay's fraud prevention systems. Once this value is being set on all transactions contact BluePay to have the velocity filter enabled. This will prevent costly large volume fraudulent transactions.

USER_AGENT	
Required:	Optional
Description:	Name and version of the software submitting the transaction.

USER_ID	
Required:	Optional
Description:	User ID number of the of the gateway user account that will be entered as the owner of the transaction. Owner will be the default user if not provided.

BP_STAMP_DEF	
Required:	Optional
Description:	List of field names in the order they are to be used in the calculation of BP_STAMP. This is similar to the TPS_DEF, but the BP_STAMP_DEF uses output fields rather than input fields. Refer BP_STAMP in output section for more information.

F_BYPASS_CAU	
Required:	Optional for TRANSACTION_TYPE of AUTH & SALE
Description:	The Card Account Updater service will not be used. On some occasions the Card Account Updater service receives invalid information. This flag can be used to process a transaction using the provided information instead of the updated information.
Valid Value:	• 0 or not present to use Card Account Updater if enabled on the gateway account. • 1 to process transaction as submitted.

These fields may be required for the best transaction rate.

Field	Required	Description
NAME	Optional	Name can be set to the first and last name separated by a space. We prefer NAME1 and NAME2 as separate fields.
NAME1	Optional	Name1 can be set to the first name.
NAME2	Optional	Name2 can be set to the last name.
ADDR1	Yes	Billing address or a primary address of the customer who has paid the bill
ADDR2	Optional	Additional address information of the customer who has paid the bill.
CITY	Optional	City name associated with the customer billing address.
STATE	Optional	State name associated with the customer billing address.
ZIPCODE	Yes	Zip or postal code of the customer who has paid the bill.
COUNTRY	Optional	Country name associated with the customer billing address.
PHONE	Optional	Phone number associated with the customer who has paid the bill.
EMAIL	Optional	Email address associated with the customer who has paid the bill.

MICRO DEPOSITS

A Micro-Deposit is an ACH credit or debit entry used by a merchant for the purpose of verifying a customer's account or an individual's access to an account. ACCTVERIFY in the ACH_DESCRIPTION identifies the transaction is a Micro-Deposit.

Examples of Micro-Deposits credits and debits that are permitted:

- Example 1 -One credit Micro-Deposit of $0.34 and one debit Micro-Deposit of $0.19 (aggregate net credits are permitted)
- Example 2 -Two credit Micro-Deposits of $0.18 and $0.49; no debit offsets (multiple credit Micro-Deposits are permitted)
- Example 3 -Two credit Micro-Deposits of $0.18 and $0.49; and 1 offsetting debit Micro-Deposit of $0.67 (aggregate Micro-Deposits net to zero)
- Example 4 -Two credit Micro-Deposits of $0.37 and $0.84; and 1 offsetting debit Micro-Deposit of $1.21 (debit Micro-Deposits can be greater than $1 to offset multiple credit Micro-Deposits)
- Example 5 -Two credit Micro-Deposits of $0.52 and $0.63; and 2 offsetting debit Micro-Deposit of $0.71 and $0.44 (multiple debit and credit Micro-Deposits that net to zero)
- Examples that are improper:
- Example 6 -One credit of $1.08 (a credit that is $1.00 or more)
- Example 7 -One credit of $0.19 and 1 debit for $0.34 (in aggregate, a net debit)
- Example 8 -One debit of $0.34 (a debit that is not an offset of a credit)

STORED CREDENTIALS

Only merchants who are set up to use the following processor(s) should include these fields:

North/8583

The use of a Stored Credential is only meant to provide additional information to the credit card companies to better manage risk and fraud. It is not used in place of payment information. Credit card companies want to know when a merchant stores a cardholder's payment information *with the cardholder's permission*, as well as when the merchant uses that stored payment information. The most common example of this would be when a cardholder sets up a recurring payment.

When sending initial transaction:

- Include STORED_INDICATOR=F
- Include STORED_TYPE set to either 'M' or 'C'

- Include RESPONSEVERSION with a value of 8 or higher
- A transaction identifier will be returned in the response as STORED_ID (only returned if the transaction was approved)

When sending subsequent transactions:

- Include STORED_INDICATOR=S.
- Include STORED_TYPE set to either 'M' or 'C'.
- Include STORED_ID set to the value received in the initial transaction.

The following fields are available in the Stored Credentials:

STORED_INDICATOR	
Required:	Optional
Description:	This field indicates whether a stored credential is being used with the transaction. Refer STORED CREDENTIALS section for more information. • 'F' - "First" transaction of a stored credential transaction. • 'S' - "Subsequent" transaction in a stored credential setup. • (null) - Ignore this entire new system and do nothing different.
Where sending the first transaction ('F'), the RESPONSEVERSION field must be set to 8 or higher so that the response will include the STORED_ID (which will be required for the subsequent transactions).	

STORED_ID	
Required:	Yes, if STORED_INDICATOR = 'S'
Description:	Value of the STORED_ID response field from the initial 'F' stored indicator transaction. Refer STORED CREDENTIALS section for more information.

STORED_TYPE	
Required:	Yes, if STORED_INDICATOR sent
Description:	This field indicates who initiated this transaction. Refer STORED CREDENTIALS section for more information. • 'M' - "Merchant" initiated stored credentials transaction. • 'C' - "Cardholder" initiated stored credentials transaction.

LEVEL 2 (Optional)

The following optional fields are available in the Level 2:

INVOICE_ID	
Required:	Optional; US Required: Yes (L2); Canada Required: Yes (L2)
Description:	64 Characters Invoice ID'.
Length/Type:	10/AN

AMOUNT_TAX	
Required:	Optional; US Required: Yes (L2); Canada Required: Yes (L2)
Description:	If AMOUNT_TAX is supplied AMOUNT value must be modified to include this value.
Length/Type:	6/N

CUSTOMER_CODE	
Required:	Optional
Description:	Customer or Purchase Order number
Length/Type:	Up to 16 characters

LEVEL 3 (Optional)

The following optional fields are available for the Level 3 processing. Fields with names starting with LV2 are order-level Level 3 fields and fields with names starting with LV3 are line-item detail Level 3 fields.

LV2_ITEM_TAX_RATE	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Tax rate for the order.
Length/Type:	4/N

LV2_ITEM_GOODS_TAX_RATE	
Required:	Optional; **Canada Required:** Yes (L3)

LV2_ITEM_GOODS_TAX_RATE	
Description:	Tax rate for the item and goods on the order.
Length/Type:	4/N

LV2_ITEM_GOODS_TAX_AMOUNT	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Tax amount for the item and goods on the order.
Length/Type:	6/N

LV2_ITEM_SHIPPING_AMOUNT	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Shipping amount for the order.
Length/Type:	10/N

LV2_ITEM_DISCOUNT_AMOUNT	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Discount amount for the order.
Length/Type:	10/N

LV2_ITEM_CUST_PO	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Customer PO of the order.
Length/Type:	22/AN

LV2_ITEM_GOODS_TAX_ID	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Goods Tax ID of the order.
Length/Type:	30/AN

LV2_ITEM_TAX_ID	
Required:	Optional; **Canada Required:** Yes (L3)

LV2_ITEM_TAX_ID	
Description:	Tax ID for the order.
Length/Type:	30/AN

LV2_ITEM_CUSTOMER_TAX_ID	
Required:	Optional; **Canada Required:** Yes (L3)
Description:	Customer Tax ID for the order.
Length/Type:	15/AN

LV2_ITEM_DUTY_AMOUNT	
Required:	Optional
Description:	Duty amount for the order.
Length/Type:	10/N

LV2_ITEM_SUPPLEMENTAL_DATA	
Required:	Optional
Description:	Supplemental data for the order.
Length/Type:	24/AN

LV2_ITEM_CITY_TAX_RATE	
Required:	Optional
Description:	City tax rate for the order.
Length/Type:	4/N
Amex Only:	Yes

LV2_ITEM_CITY_TAX_AMOUNT	
Required:	Optional
Description:	City tax amount for the order.
Length/Type:	6/N
Amex Only:	Yes

LV2_ITEM_COUNTY_TAX_RATE	
Required:	Optional
Description:	Country tax rate for the order.
Length/Type:	4/N
Amex Only:	Yes

LV2_ITEM_COUNTY_TAX_AMOUNT	
Required:	Optional
Description:	Country tax amount for the order.
Length/Type:	6/N
Amex Only:	Yes

LV2_ITEM_STATE_TAX_RATE	
Required:	Optional
Description:	State tax rate for the order.
Length/Type:	4/N
Amex Only:	Yes

LV2_ITEM_STATE_TAX_AMOUNT	
Required:	Optional
Description:	State tax amount for the order.
Length/Type:	6/N
Amex Only:	Yes

LV2_ITEM_BUYER_NAME	
Required:	Optional; Canada Required: Yes (L3)
Description:	Buyer name for the order.
Length/Type:	40/AN
Amex Only:	Yes

LV2_ITEM_CUSTOMER_REFERENCE	
Required:	Optional; US Required: Yes (L2); Canada Required: Yes (L2)
Description:	Customer reference for the order.
Length/Type:	17/AN
Amex Only:	Yes

LV2_ITEM_CUSTOMER_NUMBER	
Required:	Optional
Description:	Customer number for the order.
Length/Type:	30/AN
Amex Only:	Yes

LV2_ITEM_SHIP_NAME	
Required:	Optional
Description:	Ship name for the order.
Length/Type:	40/AN
Amex Only:	Yes

LV2_ITEM_SHIP_ADDR1	
Required:	Optional
Description:	Ship address1 for the order.
Length/Type:	40/AN
Amex Only:	Yes

LV2_ITEM_SHIP_ADDR2	
Required:	Optional; US Required: Yes (L2); Canada Required: Yes (L2)
Description:	Ship address2 for the order.
Length/Type:	40/AN
Amex Only:	Yes

LV2_ITEM_SHIP_CITY	
Required:	Optional
Description:	Ship city for the order.
Length/Type:	30/AN
Amex Only:	Yes

LV2_ITEM_SHIP_STATE	
Required:	Optional
Description:	Ship state for the order.
Length/Type:	2/A
Amex Only:	Yes

LV2_ITEM_SHIP_ZIP	
Required:	Optional
Description:	Ship zipcode for the order.
Length/Type:	15/AN
Amex Only:	Yes

LV2_ITEM_SHIP_COUNTRY	
Required:	Optional; Canada Required: Yes (L3)
Description:	Ship country for the item.
Length/Type:	3/AN
Amex Only:	Yes

The following fields are available for Level 3 processing with corporate purchasing cards. "x" is replaced with a unique sequence number for each item. Maximum number of items is 99.

LV3_ITEMx_ITEM_SKU	
Required:	Optional; Canada Required: Yes (L3)
Description:	Item SKU for the item.
Length/Type:	12/AN

LV3_ITEMx_ITEM_DESCRIPTOR	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	Description of the item purchased.
Length/Type:	26/AN

LV3_ITEMx_COMMODITY_CODE	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	The Commodity Code of the item. A list of Commodity Codes can be found at: https://urldefense.com/v3/__http://www.census.gov/svsd/www/cfsdat/2002data/cfs021200.pdf__;!!P9vvK-4S!0_fdYVH1S8Dz52NxzFz22EC-XgSDrWicB8sMVQLdPqL_OCoQxmxKufYFf5m3jboOtg$
Length/Type:	12/AN

LV3_ITEMx_PRODUCT_CODE	
Required:	Optional; US Required: Yes (L3)
Description:	Merchant-defined code for the product or service being purchased. This can be an inventory, catalog, UPC code or other.
Length/Type:	12/AN (Characters)

LV3_ITEMx_MEASURE_UNITS	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	The unit of measure of the item purchase. Normally EA for Each.
Length/Type:	12/AN (3 characters)
Refer:	Appendix I (Measure Units and Codes)

LV3_ITEMx_UNIT_COST	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	The cost per unit of item.
Length/Type:	10/N (9-digits decimal)

LV3_ITEMx_QUANTITY	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	The number of units of item.
Length/Type:	10/N (5-digitsLV3_ITEMx_ITEM_DESCRIPTOR)

LV3_ITEMx_ITEM_DISCOUNT	
Required:	Optional; Canada Required: Yes (L3)
Description:	The amount of any discounts on the item.
Length/Type:	10/N (12 digits decimal)

LV3_ITEMx_TAX_RATE	
Required:	Optional; US Required: Yes (L3)
Description:	Tax rate for the item.
Length/Type:	4/N (no % sign)

LV3_ITEMx_GOODS_TAX_RATE	
Required:	Optional; US Required: Yes (L3)
Description:	Goods tax rate for the item.
Length/Type:	4/N

LV3_ITEMx_TAX_AMOUNT	
Required:	Optional; US Required: Yes (L3)
Description:	Tax amount for the item.
Length/Type:	10/N (9-digits)

LV3_ITEMx_GOODS_TAX_AMOUNT	
Required:	Optional; Canada Required: Yes (L3)
Description:	Goods tax amount for the item.
Length/Type:	10/N

LV3_ITEMx_CITY_TAX_RATE	
Required:	Optional
Description:	City tax rate for the item.
Length/Type:	4/N
Amex Only:	Yes

LV3_ITEMx_CITY_TAX_AMOUNT	
Required:	Optional
Description:	City tax amount for the item.
Length/Type:	10/N
Amex Only:	Yes

LV3_ITEMx_COUNTY_TAX_RATE	
Required:	Optional
Description:	Country tax rate for the item.
Length/Type:	10/N
Amex Only:	Yes

LV3_ITEMx_COUNTY_TAX_AMOUNT	
Required:	Optional
Description:	Country tax amount for the item.
Length/Type:	10/N
Amex Only:	Yes

LV3_ITEMx_STATE_TAX_RATE	
Required:	Optional
Description:	State tax rate for the item.
Length/Type:	4/N
Amex Only:	Yes

LV3_ITEMx_STATE_TAX_AMOUNT	
Required:	Optional
Description:	State tax amount for the item.
Length/Type:	10/N
Amex Only:	Yes

LV3_ITEMx_ITEM_SKU	
Required:	Optional
Description:	Item SKU for the item.
Length/Type:	30/AN
Amex Only:	Yes

LV3_ITEMx_CUST_SKU	
Required:	Optional
Description:	Customer SKU for the item.
Length/Type:	30/AN
Amex Only:	Yes

LV3_ITEMx_CUST_PO	
Required:	Optional
Description:	Customer PO for the item.
Length/Type:	22/AN
Amex Only:	Yes

LV3_ITEMx_SUPPLEMENTAL_DATA	
Required:	Optional
Description:	Supplemental data for the item.
Length/Type:	40/AN
Amex Only:	Yes

LV3_ITEMx_GL_ACCOUNT_NUMBER	
Required:	Optional; US Required: Yes (L3); Canada Required: Yes (L3)
Description:	GL account number for the item.
Length/Type:	40/AN
Amex Only:	Yes

LV3_ITEMx_DIVISION_NUMBER	
Required:	Optional
Description:	Division number for the item.
Length/Type:	40/AN
Amex Only:	Yes

LV3_ITEMx_PO_LINE_NUMBER	
Required:	Optional
Description:	PO line number for the item.
Length/Type:	5/N
Amex Only:	Yes

LV3_ITEMx_LINE_ITEM_TOTAL	
Required:	Optional; US Required: Yes (L3)
Description:	The total amount for the item including taxes and discounts.
Length/Type:	10/N

LODGING (Optional)

The following fields are available for Lodging processing:

LODGING_FOLIO_NUM	
Required:	Yes
Description:	The lodging folio number.

LODGING_ARRIVAL_DATE	
Required:	Yes
Description:	The date the customer arrived at lodging. Format is "YYYY-MM-DD".

LODGING_DEPART_DATE	
Required:	Yes
Description:	The date the customer checked out of lodging. Format is "YYYY-MM-DD".

LODGING_LOCAL_PHONE	
Required:	Yes
Description:	The local phone number of the specific lodging location, up to 10 digits.

LODGING_COUNTRY	
Required:	Yes, if outside of USA
Description:	The code for the country in which the lodging is located. Defaults to 'USA' if not set. Use the ISO 3166-1 Alpha-3 Country Code (i.e.: USA, CAN).

LODGING_PREF_CUST_F	
Required:	Optional
Description:	• 1 if the customer is a preferred customer. • 0 or not present if not applicable.

LODGING_EXTRA_GIFTSHOP_F	
Required:	Optional
Description:	• 1 if there was an additional charge for the gift shop. • 0 or not present if not applicable.

LODGING_EXTRA_LAUNDRY_F	
Required:	Optional
Description:	• 1 if there was an additional charge for a parking violation. • 0 or not present if not applicable.

LODGING_EXTRA_MINIBAR_F	
Required:	Optional
Description:	• 1 if there was an additional charge for the mini bar. • 0 or not present if not applicable.

LODGING_EXTRA_RESTAURANT_F	
Required:	Optional
Description:	• 1 if there was an additional charge for the restaurant. • 0 or not present if not applicable.

LODGING_EXTRA_PHONE_F	
Required:	Optional
Description:	• 1 if there was an additional charge for the telephone. • 0 or not present if not applicable.

LODGING_EXTRA_OTHER_F	
Required:	Optional
Description:	• This should not include gift shop, laundry, mini bar, restaurant, or telephone charges. • 1 if there was an additional charge. 0 or not present if not applicable.

LODGING_CHARGE_TYPE	
Required:	Optional
Description:	The code for the description of the lodging charge. If the value is not set then the default value is '1' (Hotel Stay)
Possible Codes:	• 1 = Hotel Stay • 2 = Assured Reservation - No Show • 3 = CAR Deposit • 4 = Delayed Charge • 5 = Express Service • 6 = Assured Reservation

VEHICLE RENTAL (Optional)

The following fields are available for Vehicles Rental processing:

VEHICLE_RENT_AGREE_NUM	
Required:	Yes
Description:	The invoice or rental agreement number of the original vehicle rental agreement.

VEHICLE_PICKUP_DATETIME	
Required:	Yes
Description:	The date the vehicle rental began. The format is "YYYY-MM-DD HH:MM:SS" and Hours, minutes, and seconds are optional.

VEHICLE_DROPOFF_DATETIME	
Required:	Yes
Description:	The date the vehicle rental ended. "YYYY-MM-DD HH:MM:SS" Hours, minutes, and seconds are optional.

VEHICLE_PICKUP_CITY	
Required:	Yes (unless same as merchant's city)
Description:	The city in which the vehicle was rented. Defaults to merchant's city if not set.

VEHICLE_DROPOFF_CITY	
Required:	Yes, (unless same as merchant's city)
Description:	The city in which the vehicle was returned. Defaults to merchant's city if not set.

VEHICLE_PICKUP_STATE	
Required:	Yes, (unless same as merchant's city)
Description:	The 2-letter abbreviation for the state/province in which the vehicle was rented. Defaults to merchant's state if not set. Use the ISO 3166-2 US or CA standard (i.e.: "AL" = Alabama, "AB" = Alberta).

VEHICLE_DROPOFF_STATE	
Required:	Yes, (unless same as merchant's city)
Description:	The 2-letter abbreviation for the state/province in which the vehicle was rented. Defaults to merchant's state if not set. Use the ISO 3166-2 US or CA standard (i.e.: "AL" = Alabama, "AB" = Alberta).

VEHICLE_DROPOFF_LOCATION_ID	
Required:	Yes
Description:	Identification of return location (phone, store ID, etc.), up to 10 characters.

VEHICLE_PICKUP_COUNTRY	
Required:	Yes, (if outside of USA)
Description:	The code for the country in which the vehicle was returned. Defaults to 'USA' if not set. Use the ISO 3166-1 Alpha-3 Country Code (i.e.: USA, CAN).

VEHICLE_DROPOFF_COUNTRY	
Required:	Yes, (if outside of USA)
Description:	The code for the country in which the vehicle was returned. Defaults to 'USA' if not set. Use the ISO 3166-1 Alpha-3 Country Code (i.e.: USA, CAN).

VEHICLE_RENTER_NAME1 & VEHICLE_RENTER_NAME2 / VEHICLE_RENTER_NAME	
Required:	Yes, (if different from billing name (NAME1 & NAME2 / NAME))
Description:	The renter's name may be sent as VEHICLE_RENTER_NAME1 & VEHICLE_RENTER_NAME2 (preferred) or as VEHICLE_RENTER_NAME.

VEHICLE_RENTER_NAME1	
Required:	Optional
Description:	The first name of the renter. Defaults to NAME1 if not set.

VEHICLE_RENTER_NAME2	
Required:	Optional
Description:	The last name of the renter. Defaults to NAME2 if not set.

VEHICLE_RENTER_NAME	
Required:	Optional
Description:	This can be set to the first and last name separated by a space. We prefer VEHICLE_RENTER_NAME1 and VEHICLE_RENTER_NAME2 as separate fields. Defaults to NAME if not set.

VEHICLE_PREF_CUST_F	
Required:	Optional
Description:	1 if the customer is a preferred customer. 0 or not present if not applicable.

VEHICLE_EXTRA_GAS_F	
Required:	Optional
Description:	1 if there was an additional charge for gas. 0 or not present if not applicable.

VEHICLE_EXTRA_MILES_F	
Required:	Optional
Description:	1 if there was an additional charge for extra mileage. 0 or not present if not applicable.

VEHICLE_EXTRA_LATE_F	
Required:	Optional
Description:	• 1 if there was an additional charge for a late return. • 0 or not present if not applicable.

VEHICLE_EXTRA_VIOLATIONS_F	
Required:	Optional
Description:	• 1 if there was an additional charge for a parking violation. • 0 or not present if not applicable.

VEHICLE_ONEWAY_AMOUNT	
Required:	Optional
Description:	The amount charged for a one-way rental.

VEHICLE_NO_SHOW_AMOUNT	
Required:	Optional
Description:	The amount charged if the renter did not claim the vehicle as scheduled.

VEHICLE_TOTAL_EXTRA_AMOUNT	
Required:	Optional
Description:	The total amount of extra charges that have been added.

VEHICLE_TYPE	
Required:	Optional
Description:	The code for the type of vehicle being rented. Defaults to '99' (Miscellaneous) if not set. Refer the Possible codes in the Appendix II.

REDIRECT BY POST

If the amount of data being returned to the destination server exceeds the maximum limit of a GET request POST can be used instead. This is done by the gateway displaying a web page containing a form with the result fields as hidden values. When the page is loaded, the form is immediately posted using Javascript.

RESPONSETYPE	
Required:	Optional
Description:	• **GET (Default):** Results are returned to customer's web browser as the query string of a redirect URL pointing to the destination server with HTTP Status 301. • **POST**: Results are returned to customer's web browser as a web page that automatically posts to the destination server. • **DIRECT:** Results are returned directly to the application that made the transaction request with HTTP Status 200.

RESPONSEVERSION value must be >= 3 to use this option.

RESPONSE_TITLE	
Required:	Optional
Description:	Title displayed on auto-posting web page.

RESPONSE_BODY	
Required:	Optional
Description:	Content to display in the body of the auto-posting web page.

ADDITIONAL MERCHANT-SPECIFIC DATA

MERCHDATA_*	
Required:	Optional
Description:	Any fields sent to the BluePay system that begin with MERCHDATA_ will be stored and returned in the response from the bp10emu API, as well as being available in the Transaction Notification API. Field name appended to MERCHDATA_ will be stored as lower case.

BluePay supports storing up to 4096 bytes of merchant data. However, not all browsers are capable of handling such a long query string in a 302/GET like BluePay will return from this API. Please keep this in mind as you plan your usage.

MERCHANT-DEFINED FIELD VALIDATORS

<table>
<tr><th colspan="2">*_FORMAT</th></tr>
<tr><td>Required:</td><td>Optional</td></tr>
<tr><td>Description:</td><td>Any fields sent to BluePay that end with _FORMAT will be interpreted as validation rules for the associated fieldname. For instance, NAME1_FORMAT contains a validation rule for NAME1. Validation rules are in the form of a (very small) subset of standard Perl-style Regular Expression syntax. Any validation rules provided are used IN ADDITION TO validation rules specified by BluePay, not INSTEAD OF. If a validation rule fails to match on the supplied field data, then BluePay will return an ERROR with the MESSAGE set to "FIELDNAME_FORMAT CHECK FAILED".

The subset of Regular Expression currently allowed does NOT include any literal characters, only classes. The subset of Regular Expression syntax currently allowed includes.

<table>
<tr><th>Regular Expression</th><th>Description</th></tr>
<tr><td>\w, \W</td><td>= a "word character" or the inverse</td></tr>
<tr><td>\s, \S</td><td>= a "whitespace character" or the inverse</td></tr>
<tr><td>\d, \D</td><td>= a "digit character" or the inverse</td></tr>
<tr><td>.</td><td>= any character</td></tr>
<tr><td>^;</td><td>= beginning of string</td></tr>
<tr><td>$</td><td>= end of string</td></tr>
<tr><td>{x}</td><td>= specify exactly this many of preceding specifier</td></tr>
<tr><td>{x,y}</td><td>= specify minimum and maximum for preceding specifier</td></tr>
<tr><td>+</td><td>= specify one or more of preceding specifier</td></tr>
<tr><td>*</td><td>= specify zero or more of preceding specifier</td></tr>
<tr><td>[...]</td><td>= specify a character class composed of \w\s\d\W\S\D, no literal characters allowed</td></tr>
<tr><td>[^...]</td><td>= inverse of above</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th colspan="2">*_FORMAT</th></tr>
<tr><td>Example:</td><td>A few quick examples:
<table>
<tr><th>Regular Expression</th><th>Description</th></tr>
<tr><td>NAME1</td><td>= "Test"</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "\w+", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\w+$", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\d+$", FAIL</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\D+$", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\d+\s+.*$", FAIL</td></tr>
<tr><td>NAME1</td><td>= "1234"</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "\w+", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\w+$", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\d+$", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\D+$", FAIL</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\d+\s+.*$", FAIL</td></tr>
<tr><td>NAME1</td><td>= "1234 Foo Street"</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "\w+", PASS</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\w+$", FAIL</td></tr>
<tr><td>NAME1_FORMAT</td><td>= "^\d+$", FAIL</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\D+$", FAIL</td></tr>
<tr><td>NAME1_FORMAT</td><td>="^\d+\s+.*$", PASS</td></tr>
</table></td></tr>
</table>

TRANSACTION TIMESTAMP

The amount of time allowed between the display of an order form and the submission of payment can be controlled by the use of a time stamp. If more than the allowed amount of time has passed the transaction will return a Result with an ERROR status and a MESSAGE of TRANSACTION IS TOO OLD. Transactions with a STATUS of ERROR are not saved and cannot be seen in the BluePay Manager web site.

TRANS_SUBMIT_TIMESTAMP	
Required:	Optional
Description:	The date/time when the order form was displayed to the customer. Central Time is used for this field. If the TRANS_SUBMIT_TIMESTAMP is in the future the transaction will be allowed.
Format:	YYYY-MM-DD HH:MM:SS

TRANS_EXPIRE_PERIOD	
Required:	Optional
Description:	The number of units from the date/time in TRANS_SUBMIT_TIMESTAMP that submission of payment is allowed. For example, setting TRANS_EXPIRE_PERIOD to 30 MINUTES will give that particular transaction 30 minutes until it is deemed expired. If the unit of measure is missing or invalid the transaction will be allowed. **Possible units of measure:** MINUTES, HOURS, DAYS, MONTHS, YEARS
Format:	XX MINUTES

SWIPE OPTIONS

Method 1 (Primary)

Option	Description
SWIPE	The full output of the magnetic swipe reader. Works for both unencrypted & encrypted swipe (Magtek Dynamag KBE).

Method 2 (Unencrypted)

Option	Description
TRACK2	Track2 of the swiped data

Method 3 (Encrypted only, Magtek Dynamag HID)

Option	Description
KSN	The full output of the magnetic swipe reader. Works for both unencrypted & encrypted swipe (Magtek Dynamag KBE).
TRACK1_ENC	Track 1 data encrypted

Option	Description
TRACK1_EDL	Pre-encryption track 1 length
TRACK2_ENC	Track 2 data encrypted
TRACK2_EDL	Pre-encryption track 2 length

EMV (IDTech Augusta S)

EMV_DATA	
Required:	Optional
Description:	The full output of a keyboard emulation EMV reader.

EMV_DIRECT	
Required:	Optional
Description:	Set to "1" to indicate a direct read from a connected EMV card reader device.

EMV_FALLBACK	
Required:	Optional
Description:	Set to "1" on a swipe transaction after an EMV transaction has failed.

TRANSARMOR TOKENS

Only merchants who are set up to process TransArmor Tokens should include these fields.

When creating a TransArmor Token:

- TRANSACTION_TYPE must be AUTH or SALE
- Include F_TRANSARMOR=2
- Include RESPONSEVERSION with a value of 7 or higher
- TransArmor Token will be returned in the response as TOKEN_TRANSARMOR

When using a TransArmor Token:

- Include F_TRANSARMOR=1
- Include TransArmor Token as the CC_NUM value
- Include CARD_TYPE & CC_EXPIRES

F_TRANSARMOR	
Required:	Yes, If using or creating a TransArmor Token
Description:	Transaction token value
Valid Value:	• 0: or not present if not applicable. • 1: If using a TransArmor Token. • 2: If creating a TransArmor Token.

CARD_TYPE	
Required:	Yes, if F_TRANSARMOR=1; Otherwise, this field will be ignored
Description:	A four-character indicator of the credit card brand represented by the TransArmor Token.
Valid Value:	• AMEX = American Express • DC = Diner's Club • DISC = Discover • JCB = JCB • MC = MasterCard • VISA = Visa

TAMPER PROOF SEAL

TPS_HASH_TYPE	
Required:	Optional
Description:	The algorithm used to compute the TAMPER_PROOF_SEAL and BP_STAMP. Accepted values are 'MD5', 'SHA256', 'SHA512', 'HMAC_SHA256', or HMAC_SHA512'. Merchant's 'Hash Type in APIs' value is used if this parameter is not present.

While MD5 is still available it should not be used in new implementations.

TPS_DEF	
Required:	Optional

TPS_DEF	
Description:	Space-separated list of input field names in the order they are to be used in the calculation of the TAMPER_PROOF_SEAL. If set as blank or not sent, it will default to "MERCHANT PLATFORM_MERCHANT_ID TRANSACTION_TYPE AMOUNT REBILLING REB_FIRST_DATE REB_EXPR REB_CYCLES REB_AMOUNT RRNO AVS_ALLOWED AUTOCAP MODE CAPTCHA_REQUIRED CAPTCHA_TYPE". The merchant's Secret Key is always used in the calculation of the TAMPER_PROOF_SEAL, but should NOT be included in the TPS_DEF.

The use of this field can possibly weaken your security. Please be sure you understand how the TAMPER_PROOF_SEAL works before you consider using this option.

TAMPER_PROOF_SEAL	
Required:	Yes
Description:	Hash for security, using selected algorithm (either TPS_HASH_TYPE or account's 'Hash Type in APIs' value). This is computed by the merchant as follows: **STEP ONE** Concatenate the values of the fields that make up the TPS_DEF in same order that they are listed. Use "" (empty string - no space) as the value for any fields that are empty or unsent. When no TPS_DEF is sent ('+' represents string concatenation, and the field names represent the contents of the respective fields).

message =	MERCHANT + PLATFORM_MERCHANT_ID + TRANSACTION_TYPE + AMOUNT + REBILLING + REB_FIRST_DATE + REB_EXPR + REB_CYCLES + REB_AMOUNT + RRNO + AVS_ALLOWED + AUTOCAP + MODE + CAPTCHA_REQUIRED + CAPTCHA_TYPE

STEP TWO

- If TPS_HASH_TYPE is 'MD5', 'SHA256', or 'SHA512', find the md5sum, sha256sum, or sha512sum of (the merchant's Secret Key + message) in hex format.
- If TPS_HASH_TYPE is 'HMAC_SHA256' or 'HMAC_SHA512', find the HMAC_SHA256 or HMAC_SHA512 of (the merchant's Secret Key, message) in hex format.

Refer "Calculating the TAMPER_PROOF_SEAL" below for further information and examples.

Captcha

BluePay offers a captcha option to assist merchants in preventing velocity, or "card-runner" attacks, where a malicious party continually "tests" stolen credit card numbers for a small dollar amount to establish which ones are valid. This solution, if implemented, only allows a transaction to process if a human reads the letters portrayed in the captcha image and types them into an input box. Using the captcha is extremely simple.

The link to the BluePay captcha handler is embedded in an image tag on the merchant's checkout form. The handler is found here:

```
https://secure.bluepay.com/interfaces/captcha_handler
```

CAPTCHA_REQUIRED	
Required:	Optional
Description:	Set to 1 to require the customer to enter a valid captcha value.

CAPTCHA_TYPE	
Required:	Optional
Description:	Set to 1 to require the customer to enter a valid captcha value.
Value:	• native: - Captcha using a BluePay generated image. • recaptcha: - Google reCAPTCHA using a "I'm not a robot" checkbox.

CAPTCHA_WORD	
Required:	Yes, if CAPTCHA_REQUIRED is set to 1 and CAPTCHA_TYPE is set to native.
Description:	The value displayed in the BluePay generated image.

DEPRECATED OPTIONS

AUTOCAP	
Required:	Optional for TRANSACTION_TYPE of AUTH
Description:	If this is set to 1, and the AUTH is successful, the AUTH will automatically be CAPTURED in 15 minutes.
Valid Value:	• 1 for Automatic Capture. • 0 for regular transaction processing.

This was only used because our system would only perform scrubbing on AUTH transaction. Now it is possible to get scrubbing on SALE as well.

AVS_ALLOWED	
Required:	Optional for TRANSACTION_TYPE of AUTH & SALE; Overrides legacy AVS filter; Does not override Fraud Management settings.
Description:	Allows a string of allowed Address Verification System (AVS) response codes to be set on a per transaction basis. If the resulting AVS response is not in this list, the transaction will be voided and a decline response returned. If set to '#', all AVS responses are considered valid.
Example:	If the merchant wishes to allow AVS responses 'X', 'Y', and 'Z', he sets this to 'XYZ.'

The Fraud Management system or the legacy AVS filter should be used to set AVS filters if possible.

CVV2_ALLOWED	
Required:	Optional for AUTH & SALE; Overrides legacy CVV filter; Does not override Fraud Management settings.
Description:	Allows a string of allowed Card Verification Value (CVV) response codes to be set on a per transaction basis. If the resulting CVV response is not in this list, the transaction will be voided and a decline response returned. If set to '#', all CVV responses are considered valid. For example, if the merchant wishes to allow CVV responses 'X', 'Y', and 'Z', he sets this to 'XYZ'.

The Fraud Management system or the legacy CVV filter should be used to set CVV filters if possible.

OUTPUT FIELDS

RESPONSERVERSION NULL

The following output fields are returned in the API response.

Result	
Description:	The transaction status that resulted from the processing of this transaction.
Valid Value:	• **APPROVED:** Transaction passed format validations and was processed successfully. • **DECLINED:** Transaction did not pass fraud filtering or credit card transaction was not approved by the card issuer. • **ERROR:** Issue with transaction request data or system issue prevented the transaction from being processed. Check MESSAGE field for description of the error. • **MISSING:** Required information was missing from the transaction request. Check MESSAGE field for description of missing information.
⚠ This is the only field programmers should be checking to determine the transaction status. "Result" of a test transaction is determined by the dollar portion of the amount without cents. If the dollars are odd APPROVED is returned. If the dollars are even DECLINED is returned.	

RRNO	
Description:	A 12-digit number identifies the transaction processed. This value is useful when processing subsequent transactions against the transaction or it can serve as a token for the payment information when processing new transactions.

AMOUNT	
Description:	The amount of the transaction.

AUTH_CODE	
Description:	A six-character pseudo-ID from the processing network.

AVS	
Description:	Address Verification System (AVS) response code received on the transaction.
Value:	Refer the AVS response code in the Appendix III

AVS
When a transaction is processed with MODE=TEST if the first character of ADDR1 is one of the possible AVS response codes that value will be returned as the AVS response value.

BANK_NAME	
Description:	Up to 64 characters containing the customer's bank name.

CARD_EXPIRE	
Description:	Expiration date for a credit card in MMYY.
Format:	Four Character

CARD_TYPE	
Description:	A four-character indicator of the credit card type used, if any
Values:	• AMEX = American Express • MC = MasterCard • DISC = Discover • VISA = VISA • JCB = JCB • DCCB = Diner's Club or Carte Blanche • ENRT = EnRoute • BNKC = BankCard • SWTC = Switch • SOLO = Solo

CVV2	
Description:	Card Verification Value 2 response code. Result of the validation of the CVV2 value entered by the payer.
Values:	• _ = Unsupported on this network or transaction type • M = CVV2 Match • N = CVV2 did not match • P = CVV2 was not processed • S = CVV2 exists but was not input • U = Card issuer does not provide CVV2 service • X = No response from association • Y = CVV2 Match (Amex only when processed through Payroc

⚠ When a transaction is processed with MODE=TEST if the first character of ADDR2 is one of the possible CVV2 response codes that value will be returned as the CVV2 response value.

EMV_APPLICATION	
Description:	The EMV Application ID, or AID, to be printed on the EMV receipt. Only output when EMV data is in the transaction request.

EMV_APPNAME	
Description:	The EMV Application Name to be printed on the EMV receipt. Only output when EMV data is in the transaction request.

INVOICE_ID	
Description:	Merchant specified value. Usually, an identifier for the payment from the merchant's system.

MESSAGE	
Description:	Human-readable description of the result of the transaction which describes the transaction result. This description may be used on response pages or customer receipts but may vary between supposedly identical transactions.

⚠ Do not attempt to use this field to determine the exact "Result" of a transaction by attempting to parse the contents of this field. Use the “result” field to determine the true status of a transaction.

MISSING	
Description:	Name of missing field. Only output when Result=MISSING.

ORDER_ID	
Description:	Merchant specified value. Usually, an identifier for the payment from the merchant's system.

PAYMENT_ACCOUNT	
Description:	The payment account used for the transaction. • For credit card, there will be an “x” for each digit of the card number except the last four digits. Example: xxxxxxxxxxxx1234 • For ACH, there will be the account type, routing number and account number separated by colons. The account number will have an “x” for each digit except for the last four digits Example: C:123123123:xxxxxx4321
Format:	Character varying (32) `"<account type>:<routing number>:<x's><last four digits of account number>"`

PAYMENT_ACCOUNT	
Value:	ACH Account Type Values: • 'C' for checking • 'S' for savings • 'G' for general ledger

PAYMENT_TYPE	
Description:	Type of payment associated with the transaction. A Payment Type of "EFT", the Institution Code (3 digits) and the Transit Number (5 digits) should be combined into a single value of 8-digits and be passed in the ACH Routing Number field.
Format:	Up to 8 characters
Value:	• ACH for ACH transactions. • CREDIT for credit card transactions.

REBID	
Description:	A 12-digit identifier for a recurring billing schedule is assigned when a recurring billing schedule is created. This value is useful with the BluePay Rebill Administration (bprebadmin) API to check the status of, and make updates to, a recurring billing schedule.

⚠ This is NOT the RRNO value used when canceling a recurring billing schedule by processing a TRANSATION_TYPE=REBCANCEL transaction on the BluePay Post (bp10emu) API.

RESPONSERVERSION 1 & 2

Following fields are the additional fields returned along with fields in lower RESPONSEVERSIONs.

ACCOUNT_NAME	
Description:	The gateway account name for the account used to process the transaction.

ADDR1	
Description:	The address 1 value submitted on the transaction.

ADDR2	
Description:	The address 2 value submitted on the transaction.

AMOUNT_TAX	
Description:	The tax amount value that submitted in the transaction request.

AMOUNT_TIP	
Description:	Tip amount submitted on the transaction..

AVS_RESULT	
Description:	Alternate name for the AVS field. Refer to the AVS field for description.

BACKEND_ID	
Description:	For credit card transactions this is a transaction tracking number issued by the credit card processing network. For ACH transactions this value will initially be null. When the outbound payment is made the transaction ID for that payment will be put in this field.

BINDATA	
Description:	Tilde (~) separated list of transaction information returned by credit card processing network.
Example:	Character varying (32) `BINDATA = 6~V~X~~~~~~~~A~N~~~Y~C`

<table>
<tr><th colspan="2">BINDATA</th></tr>
<tr><td>Processing Data</td><td>Each character separated by Tilde (~) represents the following processing data
<table>
<tr><th>Character</th><th>Description</th></tr>
<tr><td>binlen:</td><td>Length of the Bank Identification Number (BIN). A value between 1 and 16</td></tr>
<tr><td>cardtype:</td><td>Refer various BINDATA available for cardtypes in Appendix IV</td></tr>
<tr><td>cardusage:</td><td>Refer the various BINDATA available for cardusage in the Appendix IV</td></tr>
<tr><td>networks:</td><td>Refer the various BINDATA available for network types in the Appendix IV for processing</td></tr>
<tr><td>ebt:</td><td>Electronic Benefit Transfer State</td></tr>
<tr><td>fsa:</td><td>Flexible Spending Account</td></tr>
<tr><td>issbin:</td><td>Unused</td></tr>
<tr><td>processbin:</td><td>Unused</td></tr>
<tr><td>ica:</td><td>Unused</td></tr>
<tr><td>prepaid:</td><td>• Not Present - Not a Prepaid Card
• P - Prepaid Card</td></tr>
<tr><td>prodid:</td><td>Card Product Subcategory. Refer the various card product sub category in the Appendix V</td></tr>
<tr><td>regulated:</td><td>• B - ISS Regulated Issuer
• N - ISS Non-Regulated Issuer or Non-US issued card
• 1 - ISS Regulated Issuer with fraud</td></tr>
<tr><td>subtype:</td><td>Refer the various BINDATA available for subtype in the Appendix IV for processing.</td></tr>
<tr><td>largeticket:</td><td>• L - Visa Large Ticket
• Space - Not Visa Large Ticket</td></tr>
<tr><td>acctlylproc:</td><td>• Y - Account Level Processing
• Space - No Account Level Processing
• N - Not applicable for BIN</td></tr>
<tr><td>acctfunsrc:</td><td>Account Fund Source
• C - Credit
• D - Debit</td></tr>
</table></td></tr>
</table>

BINDATA	
	Character / **Description** • P - Prepaid • H - Charge • R - Deferred Debit (Visa Only) • Space - Network Only

BP_STAMP_DEF	
Description:	List of fields used in the BP_STAMP calculation. If no BP_STAMP_DEF was sent in the request, then the default value will be used (RRNO ISSUE_DATE Result).

BP_STAMP	
Description:	• BP_STAMP verifies that transaction result data came from the BluePay gateway and has not been modified. It is a hash of the merchant's Secret Key and a string concatenation of the values of various fields returned from BluePay. It uses the same methodology as the TAMPER_PROOF_SEAL in transaction requests. • The exact fields and their order are determined by BP_STAMP_DEF, and the hash type used is the same algorithm that the merchant used to calculate the TAMPER_PROOF_SEAL (the TPS_HASH_TYPE provided in the response from BluePay). • A merchant should properly split BP_STAMP_DEF on whitespace, then create a message string by looping through the field names in the split BP_STAMP_DEF and concatenating each one's value to the end of the previous fields. • If a field is absent from the post, concatenate nothing (the null string) for its value. • This is computed by the merchant as follows (using the default BP_STAMP_DEF value): `message = RRNO + ISSUE_DATE + Result`

Where '+' represents string concatenation and the field names represent the contents of the respective fields - or "" (empty string - no space) if empty or absent from the post.

- If TPS_HASH_TYPE is 'MD5', 'SHA256', or 'SHA512': Find the md5sum, sha256sum, or sha512sum of (the Merchant's Secret Key + message) in hex format.

- If TPS_HASH_TYPE is 'HMAC_SHA256' or 'HMAC_SHA512': Find the HMAC_SHA256 or HMAC_SHA512 of (the Merchant's Secret Key, message) in hex format.

- Finally, the merchant should take the result and compare it to the value of BP_STAMP.If they match, the response is genuine. If they do not, the response has been modified.

CARD_COUNTRY	
Description:	Country of credit card issuer

CITY	
Description:	The city value submitted on the transaction.

COMPANY_NAME	
Description:	The company name value submitted on the transaction.

CUST_EMAIL	
Description:	The email address submitted on the transaction. Same as EMAIL.

CUST_PHONE	
Description:	The phone number submitted on the transaction. Same as PHONE.

CUSTOM_ID	
Description:	The CUSTOM_ID value submitted on the transaction.

CUSTOM_ID2	
Description:	The CUSTOM_ID2 value submitted on the transaction.

CUSTOMER_CODE	
Description:	Customer or Purchase Order number.

CVV2_RESULT	
Description:	The CVV2 result code returned by the validation of the CVV2 value submitted. Same as CVV2. Refer to CVV2 for possible response codes.

CVV2_STATUS	
Description:	The status response of credit card transactions.
Valid Value:	• 1: if a CVV2 value was supplied on the transaction. • 0: if a CVV2 value was not supplied.

DOC_TYPE	
Description:	ACH documentation type of agreement with payer. It also known as Standard Entry Class (SEC) Code.

EMAIL	
Description:	The email address submitted on the transaction. Same as CUST_EMAIL.

F_CARD_PRESENT	
Description:	• 1: for a swiped transaction. • 0: or not present for a non-swiped transaction.

F_REBILL_MASTER	
Description:	• 1: if transaction will be used as the master transaction for a rebilling. • 0: if the transaction is a single stand alone transaction.

F_WILL_CAPTURE (Deprecated)	
Description:	• 1: if AUTOCAP is enabled on the transaction request. Instead of AUTOCAP, a TRANSACTION_TYPE of SALE should be used.

FANCY_STATUS	
Description:	"Error" or "Approved" or "Declined" or "Missing"; Check MESSAGE field for description of missing information.
Valid Values:	• APPROVED - Transaction passed format validations and was processed successfully. • DECLINED - Transaction did not pass fraud filtering or credit card transaction was not approved by the card issuer. • ERROR - Issue with transaction request data or system issue prevented the • transaction from being processed. Check MESSAGE field for description of the error. • MISSING - Required information was missing from the transaction request.

"Result" of a test transaction is determined by the dollar portion of the amount without cents. If the dollars are odd APPROVED is returned. If the dollars are even DECLINED is returned.

FLAGS	
Description:	Up to 5 characters denoting internal processing flags used internally in the gateway. • A: Transaction was created by an auto-capture (deprecated) • B: Business, f_corporate flag set on transaction • C: Auto-Capture set on transaction (deprecated) • R: Transaction is a rebilling master • T: Test Transaction • W: Transaction is part of a rebilling

FRAUD_SCORE	
Description:	For each filter that the transaction does not pass 100 is added to the fraud score. The FRAUD_SCORE will be the total accumulated score.

FRAUD_LOG	
Description:	A space separated list of the fraud filters that did not pass with the assigned score for each unpassed filter. Filter code and score separated by colon. Refer the available Fraud Filter Codes list in the Appendix V

F_REBILL_MASTER	
Description:	The master rebilling transaction
Value:	• 1: if transaction will be used as the master transaction for a rebilling. • 0: if the transaction is a single standalone transaction.

ID	
Description:	The transaction ID of the transaction. Same as RRNO and TRANS_ID.

ISSUE_DATE	
Description:	Date and time from the Central Time zone when the transaction was processed.
Format:	yyyy-mm-dd hh:mm:ss

LOGIN_ACCOUNT_ID	
Description:	Gateway account ID that the transaction was run on.

MEMO	
Description:	The COMMENT value submitted on the transaction. Same as the output COMMENT.

MERCHDATA	
Description:	All the MERCHDATA values combined into a single field.

MERCHDATA_XXXXXXXXXXX	
Description:	Each individual MERCHDATA value will be returned in the output.

MODE	
Description:	Mode the transaction was processed as, LIVE or TEST.

NAME1	
Description:	The NAME1 value submitted on the transaction.

NAME2	
Description:	The NAME2 value submitted on the transaction.

ORIGIN	
Description:	The gateway interface that the transaction was processed through. Will always be bp10emu for transactions processed through bp10emu.

OWNER_USER_ID	
Description:	Login account ID number that the transaction was run as.

PHONE	
Description:	The phone number submitted on the transaction. Same as CUST_PHONE.

PROCESSOR_ID	
Description:	ID of the processor on the account that processed the transaction.

REMOTE_IP	
Description:	Either the REMOTE_IP value received in the transaction request or the customer's IP address when the post came from the customer's web browser.

STATE	
Description:	The state value submitted in the transaction request.

STATUS	
Description:	Code for the final status of the transaction.
Value:	• 1 for approved • 0 for declined • E for error.

TRANS_ID	
Description:	The transaction ID of the transaction. Same as RRNO and ID.

TRANS_TYPE	
Description:	The TRANSACTION_TYPE value received in the transaction request.

ZIP	
Description:	The ZIPCODE value received in the transaction request.

RESPONSERVERSION 3

Following are the additional fields that return along with the fields in lower RESPONSEVERSIONs.

*_FORMAT	
Description:	Merchant defined field validation rules sent to BluePay for the associated fieldname.

ACCT_UPDATE_ID	
Description:	The ID of any updated account data for the payment account used for the transaction. Updated account data only provided for merchants using Account Updater.

ACH_ACCOUNT_TYPE	
Description:	The ACH account type received in the transaction request.

ACH_ROUTING	
Description:	The ACH routing number received in the transaction request..

AMOUNT_FOOD	
Description:	The AMOUNT_FOOD value received in the transaction request.

AMOUNT_MISC	
Description:	The AMOUNT_MISC value received in the transaction request.

APPROVED_URL	
Description:	The APPROVED_URL value received in the transaction request.

AUTOCAP	
Description:	The AUTOCAP value received in the transaction request.

AVS_ALLOWED	
Description:	The AVS_ALLOWED value received in the transaction request.

BIC	
Description:	The bank ID of the customer's bank received in the transaction request.

CAPTCHA_REQUIRED	
Description:	1 if CAPTCHA_REQUIRED was enabled on the transaction.

CAPTCHA_WORD	
Description:	The captcha handler value submitted on the transaction.

CC_EXPIRES	
Description:	The CC_EXPIRES value received in the transaction request.

CC_EXPIRES_MONTH	
Description:	The CC_EXPIRES_MONTH value received in the transaction request.

CC_EXPIRES_YEAR	
Description:	The CC_EXPIRES_YEAR value received in the transaction request.

COMMENT	
Description:	The COMMENT value received in the transaction request. Same as MEMO.

CONNECTED_IP	
Description:	The IP address of the computer that accessed the BluePay system.

COUNTRY	
Description:	The COUNTRY value received in the transaction request.

CUSTOMER_IP	
Description:	The IP address of the customer's computer submitted on the transaction.

CVV2_ALLOWED	
Description:	The string of allowed CVV response codes submitted on the transaction.

DECLINED_URL	
Description:	The DECLINED_URL value received in the transaction request.

DUPLICATE_OVERRIDE	
Description:	1 if DUPLICATE_OVERRIDE was enabled on the transaction.

F_CORPORATE	
Description:	1 if IS_CORPORATE was set to 1 on the transaction.

F_UNHELD	
Description:	1 if the TRANSACTION_TYPE value received in the transaction request was 'UNHOLD' or 'UNHOLD'.

IBAN	
Description:	The bank account ID of the customer's bank account received in the transaction request.

IS_CORPORATE	
Description:	The IS_CORPORATE value received in the transaction request.

LEVEL_3_DATA	
Description:	All the LV3_ITEMx_* values combined into a single field.

LV3_ITEMx_*	
Description:	Any of the following values received in the transaction request will be output individually: • LV3_ITEMx_ITEM_DESCRIPTOR • LV3_ITEMx_ITEM_DISCOUNT • LV3_ITEMx_LINE_ITEM_TOTAL • LV3_ITEMx_MEASURE_UNITS • LV3_ITEMx_PRODUCT_CODE • LV3_ITEMx_QUANTITY • LV3_ITEMx_TAX_AMOUNT • LV3_ITEMx_TAX_RATE • LV3_ITEMx_UNIT_COST

MANDATE_DATE	
Description:	The MANDATE_DATE submitted in the transaction request.

MANDATE_ID	
Description:	The MANDATE_ID submitted in the transaction request.

MASTER_ID	
Description:	The TRANS_ID of a previous (master) transaction.

MERCHANT	
Description:	The gateway account ID value received in the transaction request.

MISSING	
Description:	The names of any missing required input fields.

MISSING_URL	
Description:	The MISSING_URL value received in the transaction request.

NAME	
Description:	The NAME value received in the transaction request.

ORIG_AMOUNT	
Description:	The AMOUNT of the previous (master) transaction.

ORIG_IS_SETTLED	
Description:	• 1: If the previous (master) transaction is settled. • 0: If it is not settled.

ORIG_TTYPE	
Description:	The TRANSACTION_TYPE of the previous (master) transaction.

REB_AMOUNT	
Description:	The REB_AMOUNT value received in the transaction request.

REB_CYCLES	
Description:	The REB_CYCLES value received in the transaction request.

REB_EXPR	
Description:	The REB_EXPR value received in the transaction request. "XX UNITS" Relative date as explained below. Format for REB_EXPR expressions is: "XXX DAY" or "XXX MONTH" or "XXX YEAR" or "XXX HOUR" or "XXX MINUTE".

Where XXX is any valid integer. For example, "10 MINUTE" represents, of course, 10 minutes.

REB_FIRST_DATE	
Description:	The REB_FIRST_DATE value received in the transaction request. Explained further below. The valid formats are: • "YYYY-MM-DD HH:MM:SS" (Hours, minutes, and seconds are optional). • "XX UNITS" Relative date as explained below (Marked from the time of the transaction, for example: 10 DAYS, 1 MONTH, 1 YEAR). • "XX UNITS" Relative date as explained below. Format for REB_FIRST_DATE expressions is: "XXX DAY" or "XXX MONTH" or "XXX YEAR" or "XXX HOUR" or "XXX MINUTE"

Where XXX is any valid integer. For example, "10 MINUTE" represents, of course, 10 minutes.

REBILLING	
Description:	The REBILLING value received in the transaction request.

REBILLING_ID	
Description:	The rebilling ID if REBILLING was set to 1 on the transaction.

RESPONSEVERSION	
Description:	The RESPONSEVERSION value received in the transaction request.

STATEMENT_DBA	
Description:	The STATEMENT_DBA submitted in the transaction request.

STATEMENT_PHONE	
Description:	The STATEMENT_PHONE submitted in the transaction request.

TRANS_EXPIRE_PERIOD	
Description:	The TRANS_EXPIRE_PERIOD submitted in the transaction request.

TRANS_SUBMIT_TIMESTAMP	
Description:	The TRANS_SUBMIT_TIMESTAMP submitted in the transaction request.

TPS_DEF	
Description:	The TPS_DEF value received in the transaction request.

TPS_HASH_TYPE	
Description:	The algorithm used to validate the TAMPER_PROOF_SEAL and to generate the BP_STAMP. This is either the TPS_HASH_TYPE value received in the transaction request or the merchant's 'Hash Type in APIs' value.

TRANSACTION_TYPE	
Description:	The TRANSACTION_TYPE value received in the transaction request.

ZIPCODE	
Description:	The ZIPCODE value received in the transaction request.

RESPONSERVERSION 4

Following fields are returned along with fields in lower RESPONSEVERSIONs.

BIN	
Description:	Bank Identification Number, the first six digits of a credit card number.

RESPONSERVERSION 5

Following fields are returned along with fields in lower RESPONSEVERSIONs.

CUST_TOKEN	
Description:	The Customer Token that may be used in the future to reuse this customer's payment information.

RESPONSERVERSION 6

Following fields are returned along with fields in lower RESPONSEVERSIONs.

LODGING_DATA	
Description:	All the LODGING* values combined into a single field.

LODGING_*	
Description:	Any of the following values received in the transaction request will be output individually. • LODGING_FOLIO_NUM • LODGING_ARRIVAL_DATE • LODGING_DEPART_DATE • LODGING_LOCAL_PHONE • LODGING_COUNTRY • LODGING_PREF_CUST_F • LODGING_EXTRA_GIFTSHOP_F • LODGING_EXTRA_LAUNDRY_F • LODGING_EXTRA_MINIBAR_F • LODGING_EXTRA_RESTAURANT_F • LODGING_EXTRA_PHONE_F • LODGING_EXTRA_OTHER_F • LODGING_CHARGE_TYPE

VEHICLE_RENTAL_DATA	
Description:	All the VEHICLE_* values combined into a single field.

VEHICLE_*	
Description:	Any of the following values received in the transaction request will be output individually: • VEHICLE_RENT_AGREE_NUM • VEHICLE_PICKUP_DATETIME • VEHICLE_DROPOFF_DATETIME • VEHICLE_PICKUP_CITY • VEHICLE_DROPOFF_CITY • VEHICLE_DROPOFF_LOCATION_ID • VEHICLE_PICKUP_COUNTRY • VEHICLE_DROPOFF_COUNTRY • VEHICLE_RENTER_NAME1 • VEHICLE_RENTER_NAME2 • VEHICLE_PREF_CUST_F • VEHICLE_EXTRA_GAS_F • VEHICLE_EXTRA_MILES_F • VEHICLE_EXTRA_LATE_F • VEHICLE_EXTRA_VIOLATIONS_F • VEHICLE_ONEWAY_AMOUNT • VEHICLE_NO_SHOW_AMOUNT • VEHICLE_TOTAL_EXTRA_AMOUNT • VEHICLE_TYPE

RESPONSERVERSION 7

Following fields are returned along with fields in lower RESPONSEVERSIONs.

TOKEN_TRANSARMOR	
Description:	The TransArmor Token that was generated for the credit card used in this transaction. This is only included when F_TRANSARMOR=2 was submitted in the transaction request.

RESPONSERVERSION 8

Following fields are returned along with fields in lower RESPONSEVERSIONs.

STORED_ID	
Description:	The transaction identifier generated after passing the STORED_INDICATOR flag as 'F' to store the customer's payment information. This value should be stored by the merchant and must be included in the STORED_ID input field for any subsequent stored credential transactions processed with STORED_INDICATOR = 'S'.

RESPONSERVERSION 9

Following fields are returned along with fields in lower RESPONSEVERSIONs.

ACH_SAME_DAY_FUNDING	
Description:	The flag value (0 or 1) received in the transaction request that controls same-day funding.'

ACH_NOC_ID	
Description:	A 12-digit Notification of Change (NOC) ID generated and displayed when a transaction is processed using the corrected details received in the ACH NOC. If there is no ACH_NOC_ID value then no Notice Of Change information was used.

VALIDATION_RESULT		
Description:	Validation result or status value when ACH bank account validation was performed.	
Possible Value:	• '15': Known bad bank account. The transaction is immediately declined (validation fee charged for LIVE) • '20': Unknown bank account but with a valid format (validation fee charged for LIVE transactions)) • '25': bank account (validation fee charged for LIVE transactions)) • '35': Bank account found but pending transaction settlement (validation fee charged for LIVE transactions)) • '45': Known good bank account (validation fee charged for LIVE transactions)	• 'B': Account validation bypassed (no validation fee charged) • 'R': Bank Account received was known bad previously (no validation fee charged) • 'E': Error, Bank account validation failed (no validation fee charged) • null: Account validation not performed (no validation fee charged)

See ACH_DESCRIPTION in the Input section for method to trigger specific results on MODE=TEST transactions.

RESPONSERVERSION 10

The following fields are returned along with fields in lower RESPONSEVERSIONs.

REB_IS_CREDIT	
Description:	Flag value that indicates whether the rebill transaction is SALE or CREDIT.
Possible Value:	• 1: For CREDIT • 0: For SALE

RESPONSERVERSION 11

The following fields are returned along with fields in lower RESPONSEVERSIONs.

PLATFORM_MERCHANT_ID	
Description:	The Platform Merchant ID associated with the transaction.

FORMATS

INPUT FORMAT

Input is expected in the form of a standard HTTP POST operation. All parameters are encoded in the body of the POST.

OUTPUT FORMAT.

- The output format is dependent on the RESPONSETYPE received in the input.
- RESPONSETYPE GET (Default) will have HTTP Status 302 with the result string appended to the destination URL as the query string. The destination URL comes from MISSING_URL, APPROVED_URL or DECLINED_URL depending on the transaction results.
- RESPONSETYPE DIRECT will have HTTP Status 200 with the result string in the HTTP BODY.
- RESPONSETYPE POST will be a form post of the result values from the payer's web browser.

Ensure the following points for the RESPONSETYPE GET.
- Programmers should note that the HTTP BODY contains undefined information.
- Do not rely on the body for return parameters. Instead, read the parameters from the HTTP HEADER, specifically in the "Location" header.
- These parameters will be appended to the selected redirection URL in URL-encoded format, and you can safely parse them from the header.
- We may add additional fields in the future, so do not rely on the order or number of fields.
- By following these guidelines and keeping the URL submission simple to change, your code will continue to work with future versions of the BluePay API without changes.

TAMPER_PROOF_SEAL & BP_STAMP

BluePay uses cryptographic hash (or "digest") functions as a means of both protecting transaction data from being altered and ensuring that the transaction is genuine. A cryptographic hash function is an algorithm that maps data of any size to a bit string of a fixed size that cannot be deciphered.

When using a program or function to calculate the TAMPER_PROOF_SEAL and BP_STAMP, make sure that it will accept a text string (or "message") argument and will return the hashed string (or "message digest") in hexadecimal form.

EXAMPLE: (Using Perl. Perl is available for free for Windows as "ActivePerl" from ActiveState.)

```
message = "This is my message."

key = "Secret key"

Run the following in the command line to create a message digest using MD5:

      % perl -e 'use Digest::MD5; print

      Digest::MD5::md5_hex("Secret KeyThis is my message.") . "\n";'

      4e87c3c67c588e3be33b8278f4f62cdf>

      %

Run the following in the command line to create a message digest using HMAC_SHA256:

      % perl -e 'use Digest::SHA; print

      Digest::SHA::hmac_sha256_hex("This is my message.", "Secret Key") . "\n";'

      8d985033f8058ae59211eced3fbfacab6e04c836ef3730345dab0b7a5d8c485d>

      %
```

All merchants have a default hash type assigned to their account. This can be viewed and updated on the merchant's Account Admin page of BluePay's Gateway (https://secure.bluepay.com) under "Hash Type in APIs". Merchants may override their default by including the TPS_HASH_TYPE field in the transaction request.

TPS_HASH_TYPE

The default hash type and the TPS_HASH_TYPE may be any of the following algorithms (in hexadecimal form):

Hash Type	Description	# of Hexadecimal Characters in Result
MD5	Use md5sum or a similar program to calculate a 128-bit hash, then convert it into hexadecimal form; result is 32 hexadecimal characters.	32
SHA256	Use sha256sum or a similar program to calculate a 256-bit hash, then convert it into hexadecimal form; result is 64 hexadecimal characters.	64
SHA512	Use sha512sum or a similar program to calculate a 512-bit hash, then convert it into hexadecimal form; result is 128 hexadecimal characters.	128
HMAC_SHA256	A 128-bit hash, resulting in a sequence of 64 hexadecimal characters.	64
HMAC_SHA512	A 128-bit hash, resulting in a sequence of 128 hexadecimal characters included.	128

Steps to find the HMAC of either SHA256 (HMAC_SHA256) or SHA512 (HMAC_SHA512):

1. Compare the length of the key (the merchant's Secret Key) to the hash's input blocksize. SHA256 blocksize = 64, SHA512 blocksize = 128.
    - If length of key is > blocksize, set the key's value to the hash of the original key.
    - If length of key is < blocksize, pad the key to the right with zeros until its length equals the blocksize.
2. Create the inner key (inner_key):
    - Create an inner padding value of 0x36 repeated the blocksize number of times.
    - Perform a bitwise exclusive-OR (XOR) on the key and the inner padding to create the inner key.
3. Create the outer key (outer_key):
    - Create an outer padding value of 0x5c repeated the blocksize number of times.
    - Perform a bitwise exclusive-OR (XOR) on the key and the outer padding to create the outer key.

4. Calculate the hash of the inner key concatenated with the text string, then calculate the hash of the outer key concatenated with the previous hash result: hash (outer_key + hash (inner_key + string))
5. Convert the result into a hex string.

A simple merchant who only sells one item from a static page, passing the information through the customer's browser as they will with the basic Weblink system, can calculate the TAMPER_PROOF_SEAL once and put it into their static HTML form, and be secure.

A more advanced merchant with a programmer or a shopping cart using the Weblink as an API can recalculate it for each transaction on the fly, allowing differing transaction amounts.

Calculating the TAMPER_PROOF_SEAL

Merchant A's account information is as follows:

Secret Key	= "abcdabcdabcdabcd"
Account ID	= "123412341234"
Hash Type in APIs (default hash type)	= "HMAC_SHA512"

The default hash type has been changed from MD5 to HMAC_SHA512 for new accounts only. Please view the default value Hash Type in APIs in your account.

If Merchant A set their TPS_DEF to "MERCHANT AMOUNT COMPANY_NAME MODE" and wanted to run a $10.00 TEST transaction, the transaction would include:

TPS_DEF	= "MERCHANT AMOUNT COMPANY_NAME MODE"
MERCHANT	= "123412341234"
Amount	= "10.00"
MODE	= “TEST”

STEP 1: Concatenate the TPS_DEF Value

Concatenate the values in the TPS_DEF to create a message string. Remember, if the field isn't sent or if the value is undefined, use an empty string as that field's value.

EXAMPLE:

message	= MERCHANT + AMOUNT + COMPANY_NAME + MODE = "123412341234" + "10.00" + "" + "TEST" = "12341234123410.00TEST"

STEP 2: Perform the Hash Type

This step will vary depending on which TPS_HASH_TYPE is sent (if any).

- If TPS_HASH_TYPE = "" or was not sent, the merchant's default hash type must be used.

TAMPER_PROOF_SEAL	= HMAC_SHA512(Secret Key + message) in hex format = HMAC_SHA512("abcdabcdabcdabcd" + "12341234123410.00TEST") in hex format = "3ff5f71100f5972015dc35f1d4b20d2bfe39402e09a4e0c5656f1fab8 db96445270e873f90dde5924cd14b8744c8945754b7907850 6987cb23a2322f9e1505f3"

- If TPS_HASH_TYPE = "SHA256".

TAMPER_PROOF_SEAL	= sha256sum(Secret Key + message) in hex format = sha256sum("abcdabcdabcdabcd" + "12341234123410.00TEST") in hex format = "DC8F5AC97C00CB83D93E74C26D0CF042377AA88D484424352B9 D96B42F89C949"

- If TPS_HASH_TYPE = "HMAC_SHA256".

TAMPER_PROOF_SEAL	= HMAC_SHA256(Secret Key, message) in hex format = HMAC_SHA256("abcdabcdabcdabcd", "12341234123410.00TEST") in hex format = “A89259FF6997496EA55360978B62F799309B22BF8E33B78B 1952CE57FDBD9DAB"

Calculating the BP_STAMP

Merchant A's account information is as follows:

Secret Key	= "abcdabcdabcdabcd"
Account ID	= "123412341234"
Hash Type in APIs (default hash type)	= "HMAC_SHA512"

If Merchant A sent the transaction described in the previous section and included their BP_STAMP_DEF set as "RRNO CUST_PHONE CARD_TYPE AVS Result", the transaction would include:

BP_STAMP_DEF	= "RRNO CUST_PHONE CARD_TYPE AVS Result"
MERCHANT	= "123412341234"

Amount	= "10.00"
MODE	= “TEST”
TPS_HASH_TYPE	= (we'll look at 3 examples)

Merchant A's response post might include the following output fields:

RRNO	= "987654321001"
AVS	= "M"
Result	= "APPROVED"
BP_STAMP_DEF	= "RRNO CUST_PHONE CARD_TYPE AVS Result"
CARD_TYPE	= "AMEX"
TPS_HASH_TYPE	= ? (TPS_HASH_TYPE sent with transaction OR the merchant's default hash type)
BP_STAMP	= ? (this differs based on the TPS_HASH_TYPE used)

STEP 1: Concatenate the BP_STAMP_DEF Value

Concatenate the values in the BP_STAMP_DEF to create a message string. Remember, if the field isn't returned or if the value is undefined, use an empty string as that field's value.

EXAMPLE:

message	= RRNO + CUST_PHONE + CARD_TYPE + AVS + Result = "987654321001" + "" + "AMEX" + "M" + "APPROVED" = "987654321001AMEXMAPPROVED"

STEP 2: Perform the BP_STAMP

Calculate the expected BP_STAMP and compare that value to the BP_STAMP in the response to verify that the response is genuine. This step will vary depending on which TPS_HASH_TYPE is included in the response.

- If TPS_HASH_TYPE = "" or wasn't sent in the transaction request, the response would include:

TPS_HASH_TYPE	= “HMAC_SHA512”
BP_STAMP	= "1d54db4494cd371ba50bcee7c36be7321312177084c6fb37f557 1a99a6ece5c9dcf69e5d85114507ca0f495f8840f2e42c9858c0119f 5210b7259eac64eea25e"

Expected BP_STAMP	= HMAC_SHA512(Secret Key + message) in hex format = HMAC_SHA512("abcdabcdabcdabcd" + "987654321001AMEXMAPPROVED") in hex format ="1d54db4494cd371ba50bcee7c36be7321312177084c6fb37f55 71a99a6ece5c9dcf69e5d85114507ca0f495f8840f2e42c9858c011 9f5210b7259eac64eea25e"

- If TPS_HASH_TYPE = "SHA256" in the transaction request, the response would include:

TPS_HASH_TYPE	= "SHA256"
BP_STAMP	= "d29ac974f13215c7d3f7b7467325b18fbe3517d329334d5c802dccd5f571b750"
Expected BP_STAMP	= sha256sum(Secret Key + message) in hex format = sha256sum("abcdabcdabcdabcd" + "987654321001AMEXMAPPROVED") in hex format = d29ac974f13215c7d3f7b7467325b18fbe3517d329334d5c802dccd5f571b750"

- If TPS_HASH_TYPE = "HMAC_SHA256" in the transaction request, the response would include:

TPS_HASH_TYPE	= "HMAC_SHA256"
BP_STAMP	= " 505b2b7680dee6ced9f5d14c71d3cd8270a1f977feba4a33b73d89670eec3414"
Expected BP_STAMP	= HMAC_SHA256(Secret Key, message) in hex format = HMAC_SHA256("abcdabcdabcdabcd", "987654321001AMEXMAPPROVED") in hex format = 505b2b7680dee6ced9f5d14c71d3cd8270a1f977feba4a33b73d89670eec3414"

Since the expected BP_STAMP matches the BP_STAMP from the response, the response is genuine.

How Rebilling Works

Following are the working processes of the BluePay Gateway rebilling engine:

- The rebilling engine runs transactions at regular intervals. The information for the transaction (customer name, test transaction status, credit card number, etc.) comes from a "template transaction."
- Merchants can use the BluePay Rebill Administration (bprebadmin) API to change the template transaction used in a rebilling sequence.
- The system requires the template transaction to process successfully to be valid for rebilling. If the template transaction fails or returns as DECLINED, the rebilling sequence will neither run nor be created.

- The merchant submits the transaction as an AUTH or SALE and the transaction is processed as usual. If it is successful, the system will create a rebilling sequence with this transaction as its template.
- The first rebilling will occur on REB_FIRST_DATE or "now + REB_FIRST_DATE" if REB_FIRST_DATE is an expression.
- Each time the system runs a rebilling, it pulls the transaction parameters from the template transaction, with the exception that the TRANSACTION_TYPE of a rebilling is always a SALE, even if the template was an AUTH.
- The AMOUNT of a rebilling is set to the amount specified as the REB_AMOUNT in the template transaction. If none is specified, it defaults to the template transaction's AMOUNT.
- If a number of cycles is set in the template transaction, the system will decrement that number with each rebilling, and the sequence will stop when the count reaches 0. If no cycles are specified, the rebilling will continue until canceled through a REBCANCEL transaction. A rebill status change using the gateway website or the BluePay Rebill Administration (bprebadmin) API.

Rebilling Examples:

The Secret Key used below is not indicative of an actual Secret Key. Secret Keys will be 32 random alphanumeric characters. Merchant B's account information is as follows:

Field Parameter	Sample Value
Secret Key	"raouhc.jbefiougb"
Account ID	"123412341234"
Hash Type in APIs (default hash type)	"HMAC_SHA512"

If any of the TPS_DEF input fields are missing, they get skipped, or you can think of it as entering them as "" (an empty string).

Remember, if the TPS_DEF is not included, the default value is used: "MERCHANT PLATFORM_ MERCHANT_ID TRANSACTION_TYPE AMOUNT REBILLING REB_FIRST_DATE REB_EXPR REB_CYCLES REB_AMOUNT RRNO AVS_ALLOWED AUTOCAP MODE CAPTCHA_REQUIRED CAPTCHA_TYPE".

EXAMPLE ONE:

Merchant B sends a $10.00 sale transaction without a TPS_DEF or TPS_HASH_TYPE.

They include these input fields with the transaction:

Field Parameter	Sample Value
MERCHANT	"demo"
TRANSACTION_TYPE	"SALE"

Field Parameter	Sample Value
AMOUNT	"10.00"
REBILLING	"" (or not sent)
REB_FIRST_DATE	"" (or not sent)
REB_EXPR	"" (or not sent)
REB_CYCLES	"" (or not sent)
REB_AMOUNT	"" (or not sent)
TAMPER_PROOF_SEAL	'2a0cc23b2cbd5162aa18ecc7e684975ad6635b4618f2a2161dac b69da5606273baa120ec99f39be7b83806a6331aca5ce3a2cca9b 86c6febc46aacf3a256312f'

⚠ TAMPER_PROOF_SEAL was calculated as - HMAC_SHA512("raouhc.jbefiougb" + "demoSALE10.00")

EXAMPLE TWO:

A membership site wants to offer 1 free month to a new member, then charge $39.99 per month until the customer cancels.

They include these input fields with the transaction:

Field Parameter	Sample Value
MERCHANT	"demo"
TRANSACTION_TYPE	"AUTH"
AMOUNT	"1.00"
REBILLING	"1"
REB_FIRST_DATE	"1 MONTH"
REB_EXPR	"" (or not sent)
REB_CYCLES	"" (or not sent)
REB_AMOUNT	"" (or not sent)
TPS_HASH_TYPE	"HMAC_SHA256"
TAMPER_PROOF_SEAL	“1d107b91f0dc39ebd1b557f93dbce6af508a626fb4e3b0f13bc671e3a59fd35a”

Field Parameter	Sample Value
⚠ TAMPER_PROOF_SEAL was calculated as - HMAC_SHA256("demoAUTH1.0011 MONTH1 MONTH39.99", "raouhc.jbefiougb")	

EXAMPLE THREE:

A cable TV company charges $150.00 upfront and $12.00 per month for a one-year contract, which explicite renewal.

They include these input fields with the transaction:

Field Parameter	Sample Value
MERCHANT	"demo"
TRANSACTION_TYPE	"SALE"
AMOUNT	"150.00"
REBILLING	"1"
REB_FIRST_DATE	"1 MONTH"
REB_EXPR	"1 MONTH"
REB_CYCLES	"11" (they are counting the first month at $150 as one of the contract months. Otherwise, this would be 12)
REB_AMOUNT	"12.00"
TPS_HASH_TYPE	"SHA256"
TAMPER_PROOF_SEAL	“B56DEB2A7BF13D12E87E76D9A2DCFA054E5E38B05A2F8CBE534CFDF84606FA8C”
⚠ TAMPER_PROOF_SEAL was calculated as - sha256sum("raouhc.jbefiougb" + "demoAUTH1.0011 MONTH1 MONTH1112.00")	

Static HTML Example and TAMPER_PROOF_SEAL

Now, Mom and Pop Merchant want to create a simple HTML payment form for their simple website. Normally, they'd make something like this:

(The below transaction information is for a $10.00 AUTH at the time of "sale" and then $5.00 rebillings for the next 11 months. Not a very realistic schedule, perhaps.)

```
<!-- HTML BEGINS HERE -->

    <html>

      <head><title>Mom and Pop Merchants</title></head>

      <body>

        <form action="https://secure.bluepay.com/interfaces/bp10emu" method=POST>

          <input type=hidden name=MERCHANT value="demo">

          <input type=hidden name=TRANSACTION_TYPE value="AUTH">

          <input type=hidden name=REBILLING value="1">

          <input type=hidden name=REB_FIRST_DATE value="1 MONTH">

          <input type=hidden name=REB_EXPR value="1 MONTH">

          <input type=hidden name=REB_CYCLES value="11">

          <input type=hidden name=REB_AMOUNT value="5.00">

          <input type=hidden name=TPS_HASH_TYPE value="SHA256">

          <input type=hidden name=TAMPER_PROOF_SEAL value="????">

          <input type=hidden name=AMOUNT value="10.00">

          <input type=hidden name=Order_ID value="UNUSED">

          <input type=hidden name=APPROVED_URL
          value=http://momandpop.com/goodpay.html">

          <input type=hidden name=DECLINED_URL
          value="http://momandpop.com/badpay.html">

          <input type=hidden name=MISSING_URL
          value="http://momandpop.com/error.html">

          <input type=hidden name=COMMENT value="SUBSCRIPTION TO WIDGET OF THE
          MONTH CLUB, 1YR">

          <table>

            <tr><td>Card number</td><td><input type=text name=CC_NUM
            value=""></td></tr>

            <tr><td>CVV2</td><td><input type=text name=CVCCVV2 value=""></td></tr>

            <tr><td>Expiration Date</td><td><input type=text name=CC_EXPIRES
```

```
            value=""></td></tr>

            <tr><td>NAME</td><td><input type=text name=NAME value=""></td></tr>

            <tr><td>Address</td><td><input type=text name=Addr1
            value=""></td></tr>

            <tr><td>Address 2</td><td><input type=text name=Addr2></td></tr>>

            <tr><td>City</td><td><input type=text name=CITY value=""></td></tr>

            <tr><td>State</td><td><input type=text name=STATE value=""></td></tr>

            <tr><td>Zipcode</td><td><input type=text name=ZIPCODE
            value=""></td></tr>

            <tr><td>Phone</td><td><input type=text name=PHONE value=""></td></tr>

            <tr><td>Email</td><td><input type=text name=EMAIL value=""></td></tr>

            <tr><td colspan=2><input type=SUBMIT value="Test it."
            name=SUBMIT></td></tr>

          </table>

        </form>

      </body>

    </html>

<!-- HTML ENDS HERE -->
```

Now we see the need for the TAMPER_PROOF_SEAL field, but Mom and Pop need to know what to set it to. First, they get their Secret Key from BluePay, by whatever method they've decided to provide it. In mom and pop's case, their **Secret Key is: "sakldjhflaskjfhasllsdkjfh"**

Now, how do Mom and Pop use the Secret Key to make their TAMPER_PROOF_SEAL correct for this transaction?

1. First, they go online and find a SHA256 hex hash generator, since they set their TPS_HASH_ TYPE to "SHA256". Then they need to enter a message string based on their Secret Key and the default TPS_DEF: "sakldjhflaskjfhasllsdkjfhdemoAUTH10.0011 MONTH1 MONTH115.00"
2. Notably, they have to put it in without the quotes and without hitting ENTER at the end (which may add a new line which may change the checksum value).
3. After submitting the message, the online tool should generate the following message digest: 996d9383ff8a2088570159be34ab0ddfe89fdeae42f4a1b96446e2c336137938
4. Now mom and pop know they should set their TAMPER_PROOF_SEAL for this transaction to: 996d9383ff8a2088570159be34ab0ddfe89fdeae42f4a1b96446e2c336137938

If Mom and Pop want to change the price of their product later on, they will have to make a new TAMPER_PROOF_SEAL for it.

```
<!-- HTML BEGINS HERE -->

 <html>

  <head><title>Mom and Pop Merchants</title></head>

  <body>

   <form action="https://secure.bluepay.com/interfaces/bp10emu" method=POST>

    <input type=hidden name=MERCHANT value="demo">

    <input type=hidden name=TRANSACTION_TYPE value="AUTH">

    <input type=hidden name=REBILLING value="1">

    <input type=hidden name=REB_FIRST_DATE value="1 MONTH">

    <input type=hidden name=REB_EXPR value="1 MONTH">

    <input type=hidden name=REB_CYCLES value="11">

    <input type=hidden name=REB_AMOUNT value="5.00">

    <input type=hidden name=TAMPER_PROOF_SEAL
    value="996d9383ff8a2088570159be34ab0ddfe89fdeae42f4
    a1b96446e2c336137938">

    <input type=hidden name=ORDER_ID value="UNUSED">

    <input type=hidden name=APPROVED_URL
    value="http://momandpop.com/goodpay.html">

    <input type=hidden name=DECLINED_URL
    value="http://momandpop.com/badpay.html">

    <input type=hidden name=MISSING_URL
    value="http://momandpop.com/error.html">

    <input type=hidden name=COMMENT value="SUBSCRIPTION TO WIDGET OF THE MONTH
    CLUB,1YR">

    <table>

     <tr><td>Card number</td><td><input type=text name=CC_NUM
     value=""></td></tr>

     <tr><td>CVV2</td><td><input type=text name=CVCCVV2 value=""></td></tr>

     <tr><td>Expiration Date</td><td><input type=text name=CC_EXPIRES
     value=""></td></tr>

     <tr><td>NAME</td><td><input type=text name=NAME value=""></td></tr>

     <tr><td>Address</td><td><input type=text name=Addr1 value=""></td></tr>

     <tr><td>Address 2</td><td><input type=text name=Addr2></td></tr>>

     <tr><td>City</td><td><input type=text name=CITY value=""></td></tr>

     <tr><td>State</td><td><input type=text name=STATE value=""></td></tr>
```

```
        <tr><td>Zipcode</td><td><input type=text name=ZIPCODE value=""></td></tr>

        <tr><td>Phone</td><td><input type=text name=PHONE value=""></td></tr>

        <tr><td>Email</td><td><input type=text name=EMAIL value=""></td></tr>

        <tr><td colspan=2><input type=SUBMIT value="Test it."
        name=SUBMIT></td></tr>

      </table>

    </form>

  </body>

 </html>

<!-- HTML ENDS HERE -->
```

Appendix I - Level 3 Measurement Units

Unit	Description
5B	Batch
ACT	Activity
ANN	Year
AU	Activity Unit
DAY	Day
DZN	Dozen
E49	Working Day
EA	Each
FOT	Foot
FTQ	Cubic Foot
GLL	Gallon
GRM	Gram
HUR	Hour
INH	Inch
KGM	Kilogram
KMT	Kilometer
LBR	Pound
LTR	Liter
MIN	Minute
MMT	Millimeter
MON	Month
MTQ	Cubic Meter
MTR	Meter
ONZ	Ounce
SEC	Second
SMI	Mile

Unit	Description
STN	Ton
TNE	Metric Ton
WEE	Week
XBG	Bag
XBX	Box
XCR	Crate
XCS	Case
XCT	Carton
XPK	Package
XPX	Pallet
XSX	Set
YRD	Yard

Appendix II - Vehicle Rental Vehicle Types

Code	Vehicle Type
01	Mini
02	Subcompact
03	Economy
04	Compact
05	Midsize
06	Intermediate
07	Standard
08	Full size
09	Luxury
10	Premium
11	Minivan
12	Van (12-passenger)
13	Van (moving)
14	Van (15-passenger)
15	Van (cargo)
16	Truck (12-foot)
17	Truck (20-foot)
18	Truck (24-foot)
19	Truck (26-foot)
20	Moped
21	Stretch
22	Regular
23	Unique
24	Exotic
25	Truck (small/medium)
26	Truck (large)

Code	Vehicle Type
27	SUV (small)
28	SUV (medium)
29	SUV (large)
30	SUV (exotic)
99	Miscellaneous

Appendix III - Credit Card Address Verification Service (AVS) Response Codes

AVS Code	Description
A	Partial match - Street Address matches, ZIP Code does not
B	International street address match, postal code not verified due to incompatible formats
C	International street address and postal code not verified due to incompatible formats
D	International street address and postal code match
E	Not a mail or phone order
F	Address and Postal Code match (UK only)
G	Service Not supported, non-US Issuer does not participate
I	Address information not verified for international transaction
M	Address and Postal Code match
N	No match - No Address or ZIP Code match
P	International postal code match, street address not verified due to incompatible format
Q	Bill to address did not pass edit checks/Card Association can't verify the authentication of an address
R	Retry - Issuer system unavailable, retry later
S	Service not supported
W	Partial match - ZIP Code matches, Street Address does not
U	Unavailable - Address information is unavailable for that account number, or the card issuer does not support
X	Exact match, 9-digit zip - Street Address, and 9-digit ZIP Code match
Y	Exact match, 5-digit zip - Street Address, and 5-digit ZIP Code match
Z	Partial match - 5-digit ZIP Code match only
1	Cardholder name matches
2	Cardholder name, billing address, and postal code match
3	Cardholder name and billing postal code match
4	Cardholder name and billing address match

AVS Code	Description
5	Cardholder name incorrect, billing address and postal code match
6	Cardholder name incorrect, billing postal code matches
7	Cardholder name incorrect, billing address matches
8	Cardholder name, billing address, and postal code are all incorrect

Appendix IV - BINDATA Values

BINDATA	Description
cardtype	
V	Visa
M	MasterCard
A	American Express
D	Discover
N	PIN Only (Non-Visa/MasterCard/Amex/Discover)
cardusage	
A	Unknown
C	Credit Hybrid (meaning it has pin capability also)
D	Unknown
E	PIN Only Debit with Electronic Benefit Transfer
H	Debit Hybrid (PIN and Signature)
J	USA Commercial Debit, Signature Only, No PIN Access
K	USA Commercial Debit, PIN Capable
L	Non USA Consumer Debit, No PIN Access
M	Non USA Commercial Debit, No PIN Access
N	Non USA Consumer Debit, PIN Capable
O	Non USA Commercial Debit, PIN Capable
P	PIN Only Debit without Electronic Benefit Transfer
R	Private Label Credit (MasterCard)
S	Signature only Debit, No PIN Access
T	Unknown
U	Reloadable Prepaid - Amex only (obsolete)
V	Stored Value Prepaid - Amex only (obsolete)
X	True credit, No PIN/Signature capability
networks	

BINDATA	Description
A5	PAVD FISERV
01	SHAZAM
02	NYCE
03	PULSE
05	STAR-WEST
07	STAR-EAST
14	STAR NE
15	PAVD
16	CNB-ENID
17	ATH-SCOTIA BANK
18	ATH-DIRECT
19	INTERLINK
20	INTERAC EVERLINK
21	EVERLINK INTERAC ICU
25	MAESTRO DIRECT
26	MCX
29	EBT ACS
30	MAESTRO STAR WEST
31	STAR WEST OPTION
33	JEANIE STAR WEST
34	JEANIE DIRECT
35	AFFN DIRECT
36	CU24-DIRECT
37	EBT TEAM OF TEXAS
39	EBT JPCHASE
41	SVS
42	EXXON

BINDATA	Description
43	BASE 24/INTERAC
46	CU24-FIDELITY
47	EBT EFUNDS
48	AFFN FIDELITY
49	EBT NORTHRUP GRUMMAN
51	CU24-FISERV
52	AFFN FISERV
56	STAR-EAST FISERV
57	PULSE FISERV
59	NYCE FISERV
62	STAR-WEST FISERV
63	STAR NE FISERV
64	MAESTRO FISERV
65	INTERLINK FISERV
67	EBT-FISERV
68	ACCEL FISERV
69	OPTION-FISERV
A	PINLESS POS
B	PINNED POS, PINLESS BILLPAY
C	PINLESS BILLPAY, PINLESS POS
D	PINNED POS, PINLESS POS
E	PINNED POS, PINLESS BILLPAY, PINLESS POS
L	PINLESS BILLPAY
P	PINNED POS
ebt	Electronic Benefit Transfer State
fsa	Flexible Spending Account
issbin	Unused

BINDATA	Description
processbin	Unused
ica	Unused
prepaid	
Not Present	Not a Prepaid Card
P	Prepaid Card
prodid	
A	Visa Traditional
B	Visa Traditional Rewards
C	Visa Signature
D	Visa Signature Preferred
E	Visa Proprietary ATM
F	Visa Classic
G	Visa Business
G1	Visa Signature Business (non-US)
G1	Visa Business Tier 3 (US)
G3	Visa Business Enhanced (non-US)
G3	Visa Business Tier 2 (US)
G4	Visa Infinite Business (non-US)
G4	Visa Business Tier 4 (US)
G5	Visa Business Rewards
I^	Visa Infinite
I1	Visa Infinite Privilege
I2	Visa Ultra High Net Worth
J3	Visa Healthcare
K^	Visa Corporate T&E
K1	Visa GSA Corporate T&E (US)?Visa Government Corporate T&E (global)
L^	Visa Electron

BINDATA	Description
N^	Visa Platinum
N1	Visa Rewards
N2	Visa Select
P^	Visa Gold
Q^	Visa Private Label
Q2	Visa Private Label Basic
Q3	Visa Private Label Standard
Q4	Visa Private Label Enhanced
Q5	Visa Private Label Specialized
Q6	Visa Private Label Premium
R^	Visa Proprietary
S^	Visa Purchasing
S1	Visa Purchasing with Fleet
S2	Visa GSA Purchasing (US)?Visa Government Purchasing (Global)
S3	Visa GSA Purchasing with Fleet (US)?Visa Government Purchasing With Fleet (global)
S4	Visa Government Services Loan
S5	Visa Commercial Transport (EBT)
S6	Visa Business Loan
U^	Visa Travel Money
V^	Visa V Pay
BPD	MasterCard Business Premium Debit
DAG	MasterCard Gold Debit MasterCard Salary
DAP	MasterCard Platinum Debit MasterCard Salary
DAS	MasterCard Standard Debit MasterCard Salary
DLG	MasterCard Gold Delayed Debit
DLH	MasterCard World Embossed Delayed Debit
DLP	MasterCard Platinum Delayed Debit

BINDATA	Description
DLS	MasterCard Card Delayed Debit
DOS	MasterCard Standard Debit Social
MAB	MasterCard World Elite for Business
MAC	MasterCard World Elite Corporate Card
MAP	MasterCard MAP Commercial Payments Account
MAQ	MasterCard Prepaid Commercial Payments Account
MBB	MasterCard Prepaid Consumer
MBC	MasterCard Prepaid Voucher
MBD	MasterCard Professional Debit Business Card
MBE	MasterCard Electronic Business Card
MBF	MasterCard Prepaid Food
MBK	MasterCard Black Card
MBM	MasterCard Prepaid Meal
MBP	MasterCard Corporate Prepaid
MBS	MasterCard B2B Product
MBT	MasterCard Corporate Prepaid Travel
MBW	MasterCard World MasterCard Black Edition Debit
MCB	MasterCard Business Card
MCC	MasterCard Credit Card (Mixed BIN)
MCD	MasterCard Debit Card
MCE	MasterCard Electronic Card
MCF	MasterCard Fleet Card
MCG	MasterCard Gold Card
MCH	MasterCard Premium Charge
MCO	MasterCard Corporate Card
MCP	MasterCard Purchasing Card
MCS	MasterCard Standard Card

BINDATA	Description
MCT	MasterCard Titanium Card
MCV	MasterCard Merchant-Branded Program
MCW	MasterCard World Card
MDB	MasterCard Business Debit Card
MDG	MasterCard Gold Debit Card
MDH	MasterCard Other Embossed Debit Card
MDJ	MasterCard Debit Enhanced
MDL	MasterCard Business Debit Other Embossed
MDO	MasterCard Debit Other
MDP	MasterCard Platinum Debit Card
MDR	MasterCard Debit Brokerage 1
MDS	MasterCard Debit Card
MDT	MasterCard Commercial Debit Card
MDW	MasterCard World Black Debit (LAC region excluding Mexico)
MEB	MasterCard Executive Business Card
MEC	MasterCard Electronic Commercial
MEF	MasterCard Electronic Payment Account
MEO	MasterCard Corporate Executive Card
MFB	MasterCard Flex World Elite
MFD	MasterCard Flex Platinum
MFE	MasterCard Flex Charge World Elite
MFH	MasterCard Flex World
MFL	MasterCard Flex Charge Platinum
MFW	MasterCard Flex Charge World
MGF	MasterCard Government Commercial Card
MHA	MasterCard Healthcare Prepaid Non-Tax
MHB	MasterCard HSA Substantiated

BINDATA	Description
MHD	MasterCard HELOC Debit Standard
MHH	MasterCard HSA Non-Substantiated
MHK	MasterCard Magna Health Access Card
MHL	MasterCard HELOC Debit Gold
MHM	MasterCard HELOC Debit Platinum
MHN	MasterCard HELOC Debit Premium
MIA	MasterCard Prepaid Unembossed Student Card
MIP	MasterCard Prepaid Student Card
MIU	MasterCard Debit Unembossed (Non-US)
MLA	MasterCard Central Travel Solutions Air Card
MLB	MasterCard Brazil Benefit for Home Improvement
MLD	MasterCard Distribution Card
MLE	MasterCard Brazil General Benefits
MLF	MasterCard Agro
MLL	MasterCard Central Travel Solutions Land Card
MNF	MasterCard Public Sector Commercial Card
MNW	MasterCard World Card
MOC	MasterCard Standard Maestro Social
MOG	MasterCard Maestro Gold Card
MOP	MasterCard Maestro Platinum
MOW	MasterCard World Maestro
MPA	MasterCard Prepaid Debit Standard Payroll
MPB	MasterCard Preferred Business Card
MPF	MasterCard Prepaid Debit Standard Gift
MPG	MasterCard Debit Standard Prepaid General Spend
MPH	MasterCard Cash
MPJ	MasterCard Prepaid Debit Voucher Meal/Food Card

BINDATA	Description
MPK	MasterCard Prepaid Government Commercial Card
MPL	MasterCard Platinum Card
MPM	MasterCard Prepaid Debit Standard Consumer Incentive
MPN	MasterCard Prepaid Debit Standard Insurance
MPO	MasterCard Prepaid Debit Standard Other
MPP	MasterCard Prepaid Card
MPR	MasterCard Prepaid Debit Standard Travel
MPT	MasterCard Prepaid Debit Standard Teen
MPV	MasterCard Prepaid Debit Standard Government
MPW	MasterCard Debit Business Card Prepaid Workplace Business to Business
MPX	MasterCard Prepaid Debit Standard Flex Benefit
MPY	MasterCard Prepaid Debit Standard Employee Incentive
MPZ	MasterCard Prepaid Debit Standard Government Consumer
MRC	MasterCard Prepaid Electronic Card (Non-US)
MRF	MasterCard Standard Deferred
MRG	MasterCard Prepaid Card (Non-US)
MRH	MasterCard Platinum Prepaid Travel Card
MRJ	MasterCard Prepaid MasterCard Voucher Meal/Food Card
MRK	MasterCard Prepaid MasterCard Public Sector Commercial Card
MRO	MasterCard Rewards Only
MRL	MasterCard Prepaid Business Preferred
MRP	MasterCard Standard Retailer Centric Payments
MRW	MasterCard Prepaid Business Card (Non-US)
MSA	MasterCard Prepaid Maestro Payroll Card
MSB	MasterCard Maestro Small Business Card
MSF	MasterCard Prepaid Maestro Gift Card
MSG	MasterCard Prepaid Maestro Consumer Reloadable Card

BINDATA	Description
MSI	MasterCard Maestro Card
MSJ	MasterCard Maestro Prepaid Voucher Meal and Food Card
MSM	MasterCard Maestro Prepaid Consumer Promotion Card
MSN	MasterCard Maestro Prepaid Insurance Card
MSO	MasterCard Maestro Prepaid Other Card
MSQ	MasterCard Reserved for future use
MSR	MasterCard Prepaid Maestro Travel Card
MST	MasterCard Prepaid Maestro Teen Card
MSV	MasterCard Prepaid Maestro Government Benefit Card
MSW	MasterCard Prepaid Maestro Corporate Card
MSX	MasterCard Prepaid Maestro Flex Benefit Card
MSY	MasterCard Prepaid Maestro Employee Incentive Card
MSZ	MasterCard Prepaid Maestro Emergency Assistance Card
MTP	MasterCard World Domestic Affluent
MUW	MasterCard World Domestic Affluent
MWB	MasterCard World for Business Card
MWD	MasterCard World Deferred
MWE	MasterCard World Elite
MWO	MasterCard World Elite Corporate Card
MWR	MasterCard World Retailer Centric Payments
OLB	MasterCard Maestro Small Business Delayed Debit
OLG	MasterCard Maestro Gold Delayed Debit
OLP	MasterCard Maestro Platinum Delayed Debit
OLS	MasterCard Maestro Delayed Debit
OLW	MasterCard World Maestro Delayed Debit
PMC	MasterCard Proprietary Credit Card (Sweden domestic)
PMD	MasterCard Proprietary Debit Card (Sweden domestic)

BINDATA	Description
PSC	MasterCard Common Proprietary Swedish Credit Card
PSD	MasterCard Common Proprietary Swedish Debit Card
PVA	MasterCard Private Label A
PVB	MasterCard Private Label B
PVC	MasterCard Private Label C
PVD	MasterCard Private Label D
PVE	MasterCard Private Label E
PVF	MasterCard Private Label F
PVG	MasterCard Private Label G
PVH	MasterCard Private Label H
PVI	MasterCard Private Label I
PVJ	MasterCard Private Label J
PVL	MasterCard Private Label L
SAG	MasterCard Gold Salary Immediate Debit
SAL	MasterCard Standard Maestro Salary
SAP	MasterCard Platinum Salary Immediate Debit
SAS	MasterCard Standard Salary Immediate Debit
SOS	MasterCard Standard Social Immediate Debit
SUR	MasterCard Prepaid Unembossed Card (Non-US)
TBE	MasterCard Electronic Business Immediate Debit
TCB	MasterCard Corporate Executive Business Card Immediate Debit
TCC	MasterCard (mixed BIN) Immediate Debit
TCE	MasterCard Electronic Immediate Debit
TCF	MasterCard Fleet Card Immediate Debit
TCG	MasterCard Gold Card Immediate Debit
TCO	MasterCard Corporate Immediate Debit
TCP	MasterCard Purchasing Card Immediate Debit

BINDATA	Description
TCS	MasterCard Standard Card Immediate Debit
TCW	MasterCard World Signia Card Immediate Debit
TEB	MasterCard Executive Business Card
TEC	MasterCard Electronic Commercial Immediate Debit
TEO	MasterCard Corporate Executive Card Immediate Debit
TNF	MasterCard Public Sector Commercial Card Immediate Debit
TNW	MasterCard New World Immediate Debit
TPB	MasterCard Preferred Business Card Immediate Debit
TPL	MasterCard Platinum Immediate Debit
TWB	MasterCard World Black Edition Immediate Debi
WBE	MasterCard World Black Edition
regulated	
B	ISS Regulated Issuer
N	ISS Non-Regulated Issuer or Non-US Issued Card
1	ISS Regulated Issuer with fraud
subtype	
AC	Brazil Agriculture Maintenance Account/Custeio
AE	Brazil Agriculture Debit Account/Electron
AG	Brazil Agriculture
AI	Brazil Agriculture Investment Loan/Investimento
CG	Brazil Cargo
CS	Construction
DS	Distribution
HC	Healthcare
LP	Visa large-Purchase Advantage
MA	Visa Mobile Agent
MB	Interoperable Mobile Branchless Banking

BINDATA	Description
MG	Visa Mobile General
VA	Brazil Food or Supermarket
VF	Brazil Fuel / Flex Visa Vale
VR	Brazil Food or Restaurant
largeticket	
L	Visa Large Ticket
Space	Not Visa Large Ticket
acctlylproc	
Y	Account Level Processing
Space	No Account Level Processing
N	Not applicable for BIN
Acctfunsrc (Account Fund Source)	
C	Credit
D	Debit
P	Prepaid
H	Charge
R	Deferred Debit (Visa Only)
Space	Network Only

Appendix V - Fraud Filter Codes

Fraud Filter Codes	Description
MINMAX	Transaction Amount (Min/Max): Transaction amount outside the range configured on the Threshold filters.
NDB_ACCT	Account Negative Database: Account number blocked by negative database.
NDB_EMAIL	Email Negative Database: Email address blocked by negative database.
NDB_IP	IP Negative Database: IP address blocked by negative database.
NDB_ADDR	Address Negative Database: Billing address blocked by negative database.
VELO_COUNT	Velocity Transaction Count: Exceeded the maximum number of transactions allowed in a 60 minute time period.
VELO_IP	Velocity Transaction Count per IP: Exceeded the maximum number of transactions allowed from a specific IP in a 60 minute time period.
VELO_AMOUNT	Velocity Total Amount: Exceeded the maximum dollar amount allowed for all Transactions in a 60 minute time period.
VELO_SALE	Sale Velocity: Exceeded the maximum dollar amount allowed for all Sales in a 60 minute time period.
VELO_CREDIT	Credit Velocity: Exceeded the maximum dollar amount allowed for all Credits in a 60 minute time period.
BIN	Card Issuing Country: Country of card issuer is not allowed.
GEO	GEO IP Tracking: Country that IP address is associated with is not allowed.
AVS	Address Verification System: Address Verification System response did not meet the AVS filter requirements.
CVV	Card Verification Value: Incorrect Card Verification Value supplied.

Revision History

Version	Revision Date	Reason for Change
1.1	April 2025	• Updated the document layout and format.