BluePay 2.0 Daily Report V2 Interface

BluePay Reporting API Documentation

January 2022/Version 2.15

About this Document

This document provides technical guidance to onboard and manage BluePay Payment Gateway merchant account.

Intended Audience

This document is written for merchants, partners, and developers who are responsible for integrating payment processing functionality with the BluePay Payment Gateway. This document provides specification on BluePay Gateway reporting API.

Assistance and Feedback

For assistance on the BluePay Payment Gateway integration, contact the Integration Support Team using the following contact information:

BluePay Integration Team

- 1-800-350-2684 (Toll Free)
- 1-630-300-0682 (Standard)

bluepay-integration@fiserv.com

Support hours are Monday through Friday 8:00a.m. to 5:00p.m. (CST UTC-6). To provide feedback on this document, or the BluePay Payment Gateway, write us at bluepay-integration@fiserv.com.

Revision History

The following versions are as per bpdaily2report API releases:

Date Revision	Version	API Updates	Document Updates
	2.00	Initial Release	
	2.01 through 2.14	Response Version 1 through Response Version 14	
January 2022	2.15	• Added ach_same_day_funding flag as a response parameter • Response Version 15 • Added BOC option in the doc_type parameter	• Redesigned the layout • Rewritten descriptions for each API • Rewritten the content as per standard writing style

Contents

Overview

The “bpdailyreport2” API is the reporting interface that retrieves transaction history based on search criteria, such as date range.

URL

https://secure.bluepay.com/interfaces/bpdailyreport2.

Input Format

This web service takes the input as the standard HTTP "POST" request. The parameters sent to the service are URI-encoded in the body of the request.

Output Format

This web service returns the output as the standard HTTP response format with header and body of the response separated by an empty line. The header contains the standard HTTP response status codes. For example, 200 indicates a success and 400 indicates an error or other request failure.

- If successful, the response body contains comma-separated transaction data
- If failed, the output contains a single line, containing an error message that displays the reason for failure

Note:

Except for the case when DO_NOT_ESCAPE option is used, existing commas within the transaction data are preceded by a backslash (“\”) character. Commas **“,”** becomes backslash commas “**\,**”.

Sample Request and Response

Use the following fields to run the API:

- Request method: POST
- Requested URL: https://secure.bluepay.com/interfaces/bpdailyreport2
- Content-Type: application/x-www-form-urlencoded

Request Format

```
Content-Type: application/x-www-form-urlencoded
Content-Length: 292
Host: secure.bluepay.com
User-Agent: Transaction Processing Application
Accept: */*

ACCOUNT_ID=100009229785&REPORT_START_DATE=2022-01-
17%2000%3A00%3A00&REPORT_END_DATE=2022-01-
19%2000%3A00%3A00&MODE=TEST&TPS_HASH_TYPE=HMAC_SHA512&TAMPER_PROOF_SEAL=456b25c0f6b0
01a6c22d4ae1a05b81c25fca3c3d3bb0ea5f8437b9d33f8c17fecc730d3c5d0c5b52f4bf2339b34cc006
e8b98c92e039feeafda688aadc1d2d04
```

Response Format

```
Date: Fri, 04 Feb 2022 15:35:20 GMT
Content-Type: text/html; charset=ISO-8859-1
Transfer-Encoding: chunked
Connection: keep-alive
X-item-count: 2
Vary: Accept-Encoding
CF-Cache-Status: DYNAMIC
Expect-CT: max-age=604800, report-uri=https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct
X-Content-Type-Options: nosniff
Server: cloudflare
CF-RAY: 6d84f7df0d1d8108-ORD

"id","payment_type","trans_type","amount","card_type","payment_account","order_id","invoice_id","custom_id","custom_id2","master_id","status","f_void","message","origin","issue_date","settle_date","rebilling_id","settlement_id","card_expire","bank_name","addr1","addr2","city","state","zip","phone","email","auth_code","name1","name2","company_name","memo","backend_id","doc_type","f_captured"

"101233145802","CREDIT","SALE","1.00","VISA","xxxxxxxxxxxx1111","101203057676","101203057676","","","101219299271","1","","Approved Sale","BATCH","2022-01-18 08:28:13","2022-01-19 02:24:08","","101233547902","1227","","25 W Randolph St","Apt 1720","Chicago","IL","60601","6303002365","John.Doe@Fiserv.com","ALVSGO","John","Doe","Fiserv, Inc.","","262578219586","",""

"101233170130","CREDIT","SALE","1.00","VISA","xxxxxxxxxxxx1111","101233170130","101233170130","","","","1","","Approved Sale","BATCH","2022-01-18 09:23:53","2022-01-19 02:24:08","","101233547902","1225","","","","","","","","","ZJUDQD","","","","","599909803219","",""
```

Note:

To understand the step-wise calculation of the Tamper Proof Seal, refer to the Appendix section.

Error Responses

An unsuccessful request returns a response with the HTTP error code ”400 Bad Request.“ The cause of the error is included in the body of the response. It gives a brief description of the error. Contact the BluePay Integration team to get assistance in resolving any errors.

The following table lists some of the common error messages.

Message	Description	Example
“Security Error"	Issue related to authentication, permissions, or some other security check.	“SECURITY ERROR”
“Missing “ + an INPUT PARAMETER	Required field is missing. If the field is included in the request, it may have been incorrectly formatted.	“Missing ACCOUNT_ID”

Input Fields

The following input fields are used to fetch the daily transaction report:

ACCOUNT_ID

- **Required**: Yes
- **Maximum Length**: 12
- **Description**: BluePay-assigned account number associated with the current reporting request

REPORT_START_DATE, REPORT_END_DATE

- **Required**: Yes
- **Description**: Date and time span for which you want to retrieve the transaction history. Pass the date and time (start and end) values, in the YYYY-MM-DD HH:MM:SS format
- **Example**: To get the list of all the statuses updated on 2021-08-08, set the REPORT_START_DATE to 2021-08-08 00:00:00 and REPORT_END_DATE to 2021-08-08 23:59:59

RESPONSEVERSION

- **Required**: No
- **Default**: 1
- **Latest**: 15
- **Description**: Version related to response fields. If not set, response contains only the version 1 (RESPONSEVERSION 1) fields

TAMPER_PROOF_SEAL

- **Required**: Yes
- **Description**: Hash for security. Compute the Tamper Proof Seal as follows:

```
Hash(The Merchant's Secret Key + ACCOUNT_ID + REPORT_START_DATE +
REPORT_END_DATE)
```

Note:

In the hex format, '**+**' represents string concatenation and the field names represent the contents of the respective fields or **""** (empty string with no space) if empty or unsent. For more information, refer to the Appendix section.

TPS_DEF

- **Required**: No
- **Description**: Space-separated list of input field names in the exact order they are to be used for calculating the TAMPER_PROOF_SEAL. If not set or blank, TPS_DEF defaults to:

```
TPS_DEF= "ACCOUNT_ID REPORT_START_DATE REPORT_END_DATE"
```

Note:

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.

DO_NOT_ESCAPE

- **Required**: No
- **Description**: Flag value that indicates whether to remove all commas and quotes from the output data and retrieve only comma-separated results
- **Values**: 0 or 1
- **Default**: 0

QUERY_BY_SETTLEMENT

- **Required**: No
- **Description**: Fetches the report based on the settlement date. By default, the **bpdailyreport2** queries by the transaction date, also known as the issue date

  Note:

  By using this option, you receive only the settled transactions in the output and not any authorization or void details in the output.
- **Values**: 0 or 1
- **Default**: 0

QUERY_BY_HIERARCHY

- **Required**: No
- **Description**: Flag that indicates to include child accounts in the output. The **account_id** field is returned in the output as the last column, regardless of RESPONSEVERSION
- **Values**: 0 or 1
- **Default**: 0

EXCLUDE_ERRORS

- **Required**: No
- **Description**: Flag that indicates to include only approvals and declines and remove any errors from the report. As billing is not done for error transactions, this parameter returns the exact count as our billing numbers
- **Values**: 0 or 1
- **Default**: 0

MODE

- **Required**: No
- **Description**: Limits transactions returned to either LIVE or TEST transactions. Live transactions actually moved funds and the Test transactions have a simulated response
- **Values**: LIVE or TEST
- **Default**: LIVE

Aggregate Fields

You can request to view an aggregated report instead of a detailed report. When you use the aggregation fields, the output displays count of transactions, sum of their amounts along with other selected columns.

AGG_QUERY

- **Required**: No
- **Description**: Flag value that indicates whether the aggregation feature is enabled or not. To enable the aggregation reporting feature, set the value to '1'
- **Values**: 0 or 1
- **Default**: 0

AGG_FIELDS

- **Default**: N/A
- **Description**: Space separated list of fieldnames. You cannot aggregate fields by "id" or "amount"
- **Example**: After URI-Encoding, `"AGG_FIELDS=payment_type%20trans_type%20card_type"`

Miscellaneous Filters

Use the additional filters to retrieve a subset of transactions or to filter the results to fit your purpose. Also, a few filters support searching for blank values by submitting a key without a value. For example, `"KEY="`

COMPANY_NAME

- **Maximum Length**: 64
- **Description**: Filter transactions with a specific company name
- **Example**: To search transactions for a company, "abcxyz", apply filters `COMPANY_NAME="abcxyz"`

FIRST_NAME

- **Maximum Length**: 32
- **Description**: Filter transactions with a specific first name
- **Example**: To filter transactions with the first name, "abc", apply filters `FIRST_NAME="abc"`

LAST_NAME

- **Maximum Length**: 32
- **Description**: Filter transactions with a specific last name
- **Example**: To filter transactions with the last name, "xyz", apply filters `LAST_NAME="xyz"`

EMAIL

- **Maximum Length**: 128
- **Description**: Filter transactions with a specific email address
- **Example**: To filter transactions with an email address, apply filters "`EMAIL=username%40mail.com`"

ORIGIN

- **Description**: Filter transactions based on the API or BluePay service, from which the transactions were originally initiated
- **Valid Values**: The valid values are:
  - "bp10emu"
  - "asbyemu"
  - "REJECT"
  - "bp20post"
  - "a.net-aim"
  - "FIXER"

- "PAYOUT"
- "REBILL"
- "AGG"
- "BATCH"
- "CAPQUEUE"
- "FRAUDSCRUB"
- "IVR"
- "VTerm"

- **Example**: Filter transactions from bp10emu "ORIGIN=bp10emu"

TRANSACTION_ID

- **Maximum Length**: 12
- **Description**: Filter transactions with a specific transaction number
- **Example**: To filter transactions with a transaction ID, apply filters "`TRANSACTION_ID=100000123456`"

MASTER_ID

- **Maximum Length**: 12
- **Description**: Filter transactions based on a specific master ID or a transaction ID token
- **Example**: To filter transactions with the Master ID, apply filters "`MASTER_ID=100000123456`"

REBILLING_ID

- **Maximum Length**: 12
- **Description**: Filter transactions by the Rebilling ID
- **Example**: To filter transactions with the Rebill ID ", apply filters "`REBILLING_ID=100000123456`"

SETTLEMENT_ID

- **Maximum Length**: 12
- **Description**: Filter transactions by the transaction's settlement ID
- **Example**: To filter transactions with the Settlement ID ", apply filters "`SETTLEMENT_ID=100000123456`"

PROCESSOR_ID

- **Maximum Length**: 12
- **Description**: Filter transactions by the processor ID
- **Example**: To filter transactions by the Processor ID ", apply filters "`PROCESSOR_ID=100000123456`"

STATUS

- **Description**: Search transactions by the transaction status
- **Valid Values**: You can pass one of the following values:
  - For Approved, "1"
  - For Declined, "0"
  - For Error, "E"
- **Example**: To fetch all the approved transactions, apply the filter "`STATUS=1`"

TRANS_TYPE

- **Description**: Search transactions by transaction type

- **Valid Values**: You can pass one of the following values:
  - AUTH
  - VOID
  - SALE
  - CAPTURE
  - REFUND
  - CREDIT
  - UPDATE
- **Example**: To fetch all the sale transactions, apply the filter "`TRANS_TYPE=SALE`"

PAYMENT_TYPE

- **Description**: Search transactions payment type
- **Valid Values**: You can pass the one of the following values:
  - CREDIT
  - ACH
  - SEPA (Single Euro Payments Area)
- **Example**: To fetch all the transactions by the Credit Card payment mode, apply the filter "`PAYMENT_TYPE=CREDIT`"

CARD_TYPE

- **Description**: Fetch the transactions by the card type
- **Valid values**: You can pass the one of the following values
  - VISA
  - MC
  - DISC
  - DCCB
  - AMEX
  - ACH
  - JCB
  - ENRT
  - BNKC
  - SWTC
  - SOLO
- **Example**: To fetch all the VISA Transactions, apply the filter "`CARD_TYPE=VISA`"

CARD_PRESENT

- **Description**: Fetch the transactions based on a flag value (card present or card not present)
- **Valid values**: You can pass one of the following values
  - To search a transaction without the card present, enter "0"
  - To search a transaction with the card present, enter "1"
- **Example**: To fetch all the transactions with a credit card present, apply the filter "`CARD_PRESENT=1`"

CUSTOM_ID

- **Maximum Length**: 16
- **Description**: Filter transactions by the custom ID

- **Example**: To filter transactions with the custom ID, apply filters "`CUSTOM_ID=abc`"

CUSTOM_ID2

- **Maximum Length**: 64
- **Description**: Filter transactions by the second custom ID
- **Example**: To filter transactions with the custom ID, apply filters "`CUSTOM_ID2=abc`"

ORDER_ID

- **Maximum Length**: 128
- **Description**: Filter transactions by the order ID
- **Example**: To filter transactions with the order ID, apply filters "`ORDER_ID=abc`"

INVOICE_ID

- **Maximum Length**: 64
- **Description**: Filter transactions by the invoice ID
- **Example**: To filter transactions with the invoice ID, apply filters "`INVOICE_ID=100234567654442`"

BACKEND_ID

- **Maximum Length**: 64
- **Description**: Filter transactions by the backend ID
- **Example**: To search for transactions with a specific backend ID, enter "`BACKEND_ID=100000123456`"

AUTH_CODE

- **Maximum Length**: 8
- **Description**: Filter transactions by a specific authorization code
- **Example**: To search for transactions with a specific bank authorization code, enter "`AUTH_CODE=abc123`"

AMOUNT

- **Maximum Length**: 9
- **Description**: Filter transactions by a specific transaction amount associated with the current transaction
- **Example**: To search for $100 transactions, enter "`Amount=100.00`"

Output Fields

The output fields are displayed as per response version. By default, you can see the limited output fields defined in the RESPONSEVERSION 1. Set RESPONSEVERSION to higher values to receive additional data.

Tip:

To view all the response fields and include future updates, set the Response Version to an arbitrarily high value (for example “99”). If you do not want the response data to change as new versions are released, set the RESPONSEVERSION to an existing version.

If the requested RESPONSEVERSION is greater than the default, the output includes all the fields defined for that version, including fields from the lower versions. For example, if the requested version is RESPONSEVERSION 3, the output fields will include all the fields defined in the response versions 1, 2 and 3.

Merchant Information

This section contains the response fields related to gateway account on which the transaction is processed:

account_id

- **Maximum Length**: 12
- **Description**: 12-digit BluePay 2.0 Account ID
- **Response Version**: 1 or higher

Note:

This field displays only when QUERY_BY_HIERARCHY is set as “1” and it will always display as the last column regardless of RESPONSEVERSION.

account_name

- **Maximum Length**: 12
- **Description**: Merchant’s BluePay Gateway account name
- **Response Version**: 9 or higher

owner_id

- **Description**: Unique ID number of the user associated with the transaction. The transactions that are processed using an API display the default BluePay account user
- **Response Version**: 9 or higher

connected_ip

- **Maximum Length**: 15
- **Description**: IP address of the computer that submitted the transaction request to the BluePay system
- **Response Version**: 4 or higher

Transaction Information

mode

- **Maximum Length**: 4
- **Description**: Valid values are **LIVE** or **TEST**
- **Response Version**: 9 or higher

origin

- **Maximum Length**: 16
- **Description**: Origination source of the transactions. The valid values are:
  - "**bp10emu**": Post and Redirect API
  - "**bp20post**": Post API
  - "**asbyemu**": AssureBuy Emulation/XML
  - "**a.net-aim**": Authorize.net Emulator
  - "**VTerm**": Virtual Terminal
  - "**AGG**": Aggregation System
  - "**BATCH**": File Upload
  - "**FRAUDSCRUB**": Fraud Management System
  - "**REBILL**": Recurring Billing System
  - "**REJECT**": Automatic reject or chargeback from the bank
- Response Version: 1 or higher

issue_date

- **Maximum Length**: 19
- **Description**: Timestamp (date and time) when the transaction entered in the BluePay system. The timestamp format is "YYYY-MM-DD HH:MM:SS"
- **Example**: The sample issue date is "`2020-10-05 04:37:00`"
- **Response Version**: 1 or higher

trans_type

- **Maximum Length**: 8
- **Description**: Transaction type for the current transaction. The possible values are:
  - AUTH
  - VOID
  - SALE
  - CAPTURE
  - REFUND
  - CREDIT
  - UPDATE

payment_type

- **Maximum Length**: 8
- **Description**: Type of payment.
- **Valid Values**: The valid values are:
  - "ACH" for Automated Clearing House transactions
  - "SEPA" for Single Euro Payments Area transactions
  - "CREDIT" for credit card transactions
- **Response Version**: 1 or higher

payment_account

- **Maximum Length**: 32
- **Description**: Payment account used for the transaction.
  - For "CREDIT" transactions, the API masks all the preceding account digits with "x" and displays only the last-four account digits
  - For "ACH" transactions, the API returns the payment account in the following format

    "<account type>:<routing number><x's and last four digits of the account number>"
- **Example**:
  - For credit card transactions, for a 16-digit account number, the payment account displays, "xxxxxxxxxxxx1111"
  - For check transactions (ACH), the payment account displays "C:123123123:xxxxxx4321," where "C" stands for Checking account and S stands for Savings account.
- **Response Version**: 1 or higher

amount

- **Maximum Length**: 9
- **Description**: Transaction amount
- **Response Version**: 1 or higher

amount_tip

- **Maximum Length**: 9
- **Description**: Tip (monetary amount) paid for this transaction
- **Response Version**: 10 or higher

master_id

- **Maximum Length**: 64
- **Description**: Master transaction ID or token transaction ID
- **Response Version**: 1 or higher

rebilling_id

- **Maximum Length**: 12
- **Description**: ID of the recurring billing schedule that initiated the transaction

- **Response Version**: 1 or higher

f_corporate

- **Maximum Length**: 1
- **Description**: Flag that indicates if the transaction is a corporate transaction or a personal transaction. Possible values are:
  - **1**: If the transaction is a corporate transaction
  - **0**: If the transaction is not a corporate transaction
- **Response Version**: 9 or higher

Transaction Response Information

id

- **Maximum Length**: 12
- **Description**: 12-digit transaction ID assigned to a transaction by BluePay
- **Response Version**: 1 or higher

status

- **Maximum Length**: 1
- **Description**: Flag value that indicates transaction status.
- **Valid Values**: The possible values are:
  - For Approved, “1”
  - For Declined, “0”
  - For Error, “E”
- **Response Version**: 1 or higher

message

- **Maximum Length**: 64
- **Description**: A short description of the transaction result
- **Response Version**: 1 or higher

bank_name

- **Maximum Length**: 64
- **Description**: Bank name associated with the payment method
- **Response Version**: 1 or higher

processor_id

- **Maximum Length**: 12
- **Description**: ID of the payment configuration that processed the transaction
- **Response Version**: 8 or higher

backend_id

- **Maximum Length**: 2048
- **Description**:
  - For credit card transactions, this is a transaction tracking number issued by the credit card processing network
  - For Third Party Sender ACH (TPS), it is the funding event ID
- **Response Version**: 1 or higher

settlement_id

- **Maximum Length**: 12
- **Description**: Settlement ID for the transaction

settle_date

- **Maximum Length**: 19
- **Description**: Date and time for the settlement of the transaction. The date format is "YYYY-MM-DD HH:MM:SS"
- **Example**: The sample settlement date is "`2020-11-05 08:37:00`"
- **Response Version**: 1 or higher

f_captured

- **Maximum Length**: 1
- **Description**: Flag value to identify if a transaction is captured. It applies only to authorization (AUTH) transactions. The possible values are:
  - "null", for transactions that are not captured
  - "1", for transactions that are captured
- **Response Version**: 1 or higher

f_refunded

- **Maximum Length**: 1
- **Description**: Flag value to identify if a transaction was refunded. The possible values are:
  - "null", for transactions that are not refunded
  - "1", for transactions that are refunded
- **Response Version**: 8 or higher

f_void

- **Maximum Length**: 1
- **Description**: Flag value to identify if the transaction was voided. The possible values are:
  - "null", for transactions that are not voided
  - "1", for transactions that are voided
- **Response Version**: 1 or higher

f_rebill_master

- **Maximum Length**: 1

- **Description**: Flag value that indicates if this transaction is the master transaction of a recurring billing schedule. The possible values are:
  - **1**: True
  - **0**: False
- **Response Version**: 9 or higher

f_will_capture

- **Maximum Length**: 1
- **Description**: Flag value to indicate if the transaction was auto-captured. The possible values are:
  - **1**: True
  - **0**: False
- **Response Version**: 9 or higher

f_unheld

- **Maximum Length**: 1
- **Description**: Flag value to indicate if the transaction was on hold or not. The possible values are:
  - **0**: If the transaction is never on hold
  - **1**: If the transaction initially is on hold, but later released or declined
- **Response Version**: 9 or higher

unhold_id

- **Maximum Length**: 12
- **Description**: Original transaction ID before the transaction was put on hold
- **Response Version**: 9 or higher

f_transarmor

- **Maximum Length**: 1
- **Description**: Flag value to indicate if a TransArmor token was used. The possible values are:
  - **0** or **NULL** if a TransArmor token was used
  - **1** if a TransArmor token was used
- **Response Version**: 11 or higher

Credit Card Payment Information

auth_code

- **Maximum Length**: 8
- **Description**: Contains the credit card authorization code in the case of a successful transaction. This field displays the reject code on voids created from ACH rejection notices
- **Response Version**: 1 or higher

card_type

- **Maximum Length**: 4
- **Response Version**: 1 or higher
- **Description**: Type of card network. The possible values are:
  - "**ACH**": Automated Clearing House (only for ACH transactions)
  - **AMEX**": American Express
  - "**MC**": Master Card
  - "**DISC**": Discover
  - "**VISA**": Visa
  - "**JCB**": JCB Co. (formerly known as Japan Credit Bureau)
  - "**DCCB**": Diner's Club or Carte Blanche
  - "**ENRT**": EnRoute
  - "**BNKC**": BankCard
  - "**SWTC**": Switch
  - "**SOLO**": Solo
- **Response Version**: 1 or higher

card_expire

- **Maximum Length**: 4
- **Description**: Credit card expiration date in the MMYY format
- **Example**: If the expiration date of a credit card is November 2026, the response displays "1126"
- **Response Version**: 1 or higher

card_present

- **Maximum Length**: 1
- **Description**: Flag value that checks whether the card was swiped at a terminal or was a non-swiped transaction
- **Values**: The possible values are:
  - "1" for a swiped transaction
  - "0" for a non-swiped transaction
- **Response Version**: 2 or higher

avs_result

- **Maximum Length**: 1
- **Description**: Address Verification System (AVS) response code received on the transaction
- **Valid Values**: The following values are valid:
  - "**A**": Street match, zip no match
  - "**N**": No match
  - "**S**": AVS not supported for this card type
  - "**U**": AVS not available for this card type
  - "**W**": Zip match 9, street no match
  - "**X**": Zip match 9, street match
  - "**Y**": Zip match 5, street match
  - "**Z**": Zip match 5, street no match
  - "**E**": Not eligible
  - "**R**": System unavailable
  - "_": Not supported for this network or transaction type

There are some international extensions. The following are the valid values for these international extensions:

- "**B**": Street match, Zip not verified
- "**C**": Street and Zip not verified
- "**D**": Street and Zip match
- "**U**": Zip match 9, street no match
- "**M**": Street and Zip match
- "**G**": Issuer does not support AVS
- "**I**": Not verified
- "**P**": Street no match, Zip match

- **Response Version**: 2 or higher

cvv_result

- **Maximum Length**: 1
- **Description**: Card Verification Value 2 response code. After the payer enters the CVV2 value, it returns the validation result. The possible values are:
  - "_" = Unsupported for 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)
- **Response Version**: 2 or higher

cvv2_status

- **Maximum Length**: 1
- **Description**: Flag that checks whether the cvv2 value was provided in the request
- **Values**: The possible values are:
  - **1**: If a CVV2 value was supplied on the transaction
  - **0**: If a CVV2 value was not supplied on the transaction
- **Response Version**: 9 or higher

acct_update_id

- **Maximum Length**: 12
- **Description**: ID of update record used on transaction. Available only for merchants using the Card Account Updater service
- **Response Version**: 6 or higher

ACH Payment Information

ach_check_num

- **Maximum Length**: 15
- **Description**: Check number of a paper check. It is used when paper checks are converted to ACH transfers

- **Response Version**: 13 or higher

doc_type

- **Maximum Length**: 3
- **Description**: ACH Service Entry Class code for the current transaction. See NACHA guidelines for specific requirements related to the usage of each type. The possible values are:
  - PPD: Prearranged Payment and Deposit
  - CCD: Corporate Credit or Debit Entry
  - WEB: Internet Initiated/Mobile Entry
  - TEL: Telephone Initiated Entry
  - CTX: Corporate Trade Exchange Entry
  - ARC: Account Receivable Entry
  - POP: Point of Purchase Entry
  - POS: Point of Sale Entry
  - BOC: Back Office Conversion Entry

ach_description

- **Maximum Length**: 10
- **Description**: Alphanumeric field to identify the type of ACH transaction being performed
- **Response Version**: 12 or higher

ach_payout

- **Maximum Length**: 12
- **Description**: Transaction ID for a payout transaction
- **Response Version**: 14 or higher

ach_reject

- **Maximum Length**: 12
- **Description**: Transaction ID of the corresponding void transaction if a reject notification is received from the bank
- **Response Version**: 14 or higher

ach_noc_id

- **Maximum Length**: 12
- **Description**: Notification of change ID when a transaction is processed using the corrected details received in the ACH NOC file. If ACH_NOC_ID field is blank, then no notification of change information was used to process the transaction
- **Response Version**: 14 or higher

ach_trace_number

- **Maximum Length**: 15
- **Description**: Unique 15-digit trace control number assigned to a transaction whenever it is added to the NACHA settlement file. Originators require this to identify the individual entries.

  - **0-7**: The first eight digits of the merchant's routing number that is stored in Immediate Origin on the processor
  - **8-14**: The remaining seven digits are sequentially numbered from "1" to "9999999" across multiple files on multiple days. The sequential number starts again at "1" once it reaches its maximum limit of "9999999."
- **Response Version**: 14 or higher

ach_same_day_funding

- **Maximum Length**: 1
- **Description**: Flag value that indicates if a transaction is funded the same-day. The possible values are:
  - **"0**" or "**NULL"**: The transaction is not funded the same-day
  - **"1"**: The transaction is funded the same-day
- **Response Version**: 15 or higher

validation_result

- **Maximum Length**: 2
- **Description**: Status value that includes bank account validation results from the ACH Account Validation service. The possible values are:
  - '**15**': Known bad bank account. The transaction is immediately declined
  - '**20**': Unknown bank account but with a valid format
  - '**25**': Unknown bank account
  - '**35**': Bank account found but pending transaction settlement (within 5 days)
  - '**45**': Known good bank account
  - '**B**': Account validation bypassed
  - '**R**': Bank account validation had failed previously
  - '**E**': Error, bank account validation failed
  - **Null**: Account validation not performed
- **Response Version**: 14 or higher

Customer Information

name1

- **Maximum Length**: 32
- **Description**: First name of the customer
- **Response Version**: 1 or higher

name2

- **Maximum Length**: 32
- **Description**: Last name or surname of the customer
- **Response Version**: 1 or higher

fancy_name

- **Description**: First name and Last name of the customer combined into a single field
- **Response Version**: 9 or higher

company_name

- **Maximum Length**: 64
- **Description**: Name of the company on the check or the credit card
- **Response Version**: 1 or higher

addr1

- **Maximum Length**: 64
- **Description**: Street address of the customer
- **Response Version**: 1 or higher

addr2

- **Maximum Length**: 64
- **Description**: Second address line of the customer
- **Response Version**: 1 or higher

city

- **Maximum Length**: 32
- **Description**: City name in which the customer resides
- **Response Version**: 1 or higher

state

- **Maximum Length**: 16
- **Description**: State or province in which the customer resides
- **Response Version**: 1 or higher

zip

- **Maximum Length**: 16
- **Description**: Zip or postal code where the customer resides
- **Response Version**: 1 or higher

country

- **Description**: Name of the country where customer resides
- **Response Version**: 9 or higher

phone

- **Maximum Length**: 16
- **Description**: Registered phone number of the customer
- **Response Version**: 1 or higher

email

- **Maximum Length**: 64
- **Description**: Email address of the customer
- **Response Version**: 1 or higher

remote_ip

- **Maximum Length**: 15
- **Description**: Remote IP address captured in the transaction request or the customer's IP address when a POST request is sent through the customer's web browser
- **Response Version**: 4 or higher

Additional Transaction Information

order_id

- **Maximum Length**: 128
- **Description**: Merchant-supplied or system supplied order ID
- **Response Version**: 1 or higher

invoice_id

- **Maximum Length**: 64
- **Description**: Merchant-supplied or system supplied invoice ID
- **Response Version**: 1 or higher

custom_id

- **Maximum Length**: 16
- **Description**: Merchant-supplied value for custom ID
- **Response Version**: 1 or higher

custom_id2

- **Maximum Length**: 64
- **Description**: Merchant-supplied value for custom ID 2

memo

- **Maximum Length**: 4096
- **Description**: Comment associated with the current transaction
- **Response Version**: 1 or higher

merchdata

- **Maximum Length**: 4096
- **Description**: Additional merchant defined fields containing the merchant supplied data
- **Response Version**: 2 or higher

level_3_data

- **Maximum Length**: 4096
- **Description**: Order and item details related to the card transaction.
  - All the field values starting with “LV2” contain Level 3 order details. For example, shipping amount, discount amount, tax rate and so on
  - All the field values starting with “LV3” contain Level 3 item details. For example, item stock-keeping-unit, item descriptor, commodity code and so on
- **Response Version**: 3 or higher

level_2_data

- **Maximum Length**: 4096
- **Description**: Purchase identification number along with the tax applied on the transaction
- **Response Version**: 5 or higher

vehicle_rental_data

- **Description**: All vehicle rental fields combined into a single field. For example, the value contains vehicle type, vehicle rental agreement number, rent amount, vehicle pick-up and drop-off details and so on
- **Response Version**: 7 or higher

lodging_data

- **Description**: All lodging fields combined into a single field. For example, the value contains lodge folio number, country, extra charges related to Laundry, restaurant, minibar and so on
- **Response Version**: 7 or higher

Appendix

Tamper Proof Seals

For a secure transaction, merchants send `TAMPER_PROOF_SEAL` value in the API request to BluePay. The TAMPER_PROOF_SEAL is used to both authenticate the request and prevent changes in request data. BluePay uses cryptographic hash or "digest" function to calculate the `TAMPER_PROOF_SEAL` value.

To calculate the hash value, merchants can use any standard hash encoder. Make sure the hash results are in hexadecimal format.

TPS Hash Types

TPS hash type can be any of the following algorithms in hexadecimal form:

Hash Type	Description	# of Hexadecimal Characters in Result
SHA256	Use sha256sum or a similar program to calculate a 256-bit hash, then convert it into hexadecimal form.	64
SHA512	Use sha512sum or a similar program to calculate a 512-bit hash, then convert it into hexadecimal form.	128
HMAC_SHA256	Use any standard program to calculate a 256-bit hash, then convert it into hexadecimal form.	64
HMAC_SHA512	Use any standard program to calculate a 512-bit hash, then convert it into hexadecimal form.	128

Calculating Tamper Proof Seal

Step 1: Build the pre-hash string

To build pre-hash, concatenate field values in the same order that they are listed in TPS_DEF. If there is no TPS_DEF value in the API request, the use "`ACCOUNT_ID REPORT_START_DATE REPORT_END_DATE`" as the TPS_DEF value. Use "" (empty string without space) for any fields that are empty or unsent.

Example: If `TPS_DEF="ACCOUNT_ID REPORT_START_DATE REPORT_END_DATE"`, then `TPS_PRE_HASH = ACCOUNT_ID + REPORT_START_DATE + REPORT_END_DATE`, where field names represent the values of the respective fields and '+' represents string concatenation.

Step 2: Perform the hash

`TAMPER_PROOF_SEAL = HASH(Account Secret Key, TPS_PRE_HASH)` where HASH is a function that performs the desired hash type

Note:

If `TPS_HASH_VALUE` is "" (empty string) or not sent, then the hash function is determined by the "`Hash Type in APIs`" value on the Account Admin page of the BluePay website.

Example

Assume the following account information for merchant 'A':

- `Account Secret Key = "abcdabcdabcdabcd"`

- `Account ID = "123412341234"`
- `Hash Type in APIs (DEFAULT_HASH_TYPE) = "HMAC_SHA512”`

If merchant A sets `TPS_DEF` to `"ACCOUNT_ID MODE REPORT_START_DATE REPORT_END_DATE EXCLUDE_ERRORS"` and wants to generate transaction report that excludes transaction errors, then the request includes the following parameters:

- `TPS_DEF = "ACCOUNT_ID MODE REPORT_START_DATE REPORT_END_DATE EXCLUDE_ERRORS"`
- `ACCOUNT_ID = “123412341234”`
- `EXCLUDE_ERRORS = "1"`
- `REPORT_START_DATE = "2021-02-28"`
- `REPORT_END_DATE = "2021-03-01"`
- `TPS_HASH_TYPE = ?`
- `TAMPER_PROOF_SEAL = ?`

To calculate the `TAMPER_PROOF_SEAL`, merchant ‘A’ can perform the following steps:

Step 1:

Concatenate the values in the `TPS_DEF` to create a pre-hash string.

```
TPS_PRE_HASH= ACCOUNT_ID + MODE + REPORT_START_DATE + REPORT_END_DATE + EXCLUDE_ERRORS
= "123412341234" + "" + "2021-02-28" + "2021-03-01" + "1"
= "1234123412342021-02-282021-03-011"
```

Step 2:

Calculate the TPS value in hexadecimal format using the applicable hash type.

`TAMPER_PROOF_SEAL = HMAC_SHA512 (Account Secret Key, TPS_PRE_HASH)` in hex format

```
= HMAC_SHA512( “abcdabcdabcdabcd”, “1234123412342021-02-282021-03
011”)

="ed0ccd231939d461cb205e7283a0bba09201d20b516b733f18b578b4cfe0b151daba5884b879442a153b
d43814efae1805191eb01e8aee913810aa1204cdae07"
```

Tip:

To calculate the TPS and retrieve transaction report, you can use sample BluePay code.

Quick Response Field Reference

The following table summarizes the response fields, expected length and minimum response version:

Fieldname	Response Version	Maximum Length
id	1	12
payment_type	1	8
trans_type	1	8

Fieldname	Response Version	Maximum Length
amount	1	9
card_type	1	4
payment_account	1	32
order_id	1	128
invoice_id	1	64
custom_id	1	16
custom_id2	1	64
master_id	1	12
status	1	1
f_void	1	1
message	1	64
origin	1	16
issue_date	1	19
settle_date	1	19
rebilling_id	1	12
settlement_id	1	12
card_expire	1	4
bank_name	1	64
addr1	1	64

Fieldname	Response Version	Maximum Length
addr2	1	64
city	1	32
state	1	16
zip	1	16
memo	1	4096
phone	1	16
email	1	64
auth_code	1	8
name1	1	32
name2	1	32
company_name	1	64
backend_id	1	2048
f_captured	1	1
account_id[1]	1	12
avs_result	2	1
cvv_result	2	1
merchdata	2	4096
card_present	2	1

[1] This field displays only when QUERY_BY_HIERARCHY is set as "1." and it always displays as the last column regardless of RESPONSEVERSION.

Fieldname	Response Version	Maximum Length
level_3_data	3	
remote_ip	4	
connected_ip	4	
level_2_data	5	
acct_update_id	6	12
vehicle_rental_data	7	
lodging_data	7	
f_refunded	8	1
processor_id	8	12
fancy_name	9	
country	9	
owner_id	9	12
mode	9	4
f_rebill_master	9	1
f_will_capture	9	1
f_corporate	9	1
cvv2_status	9	1
account_name	9	
update_id	9	12
f_unheld	9	1

Fieldname	Response Version	Maximum Length
unhold_id	9	12
amount_tip	10	9
f_transarmor	11	1
ach_reject	12	12
ach_payout	12	12
ach_check_num	13	15
ach_noc_id	14	12
validation_result	14	2
ach_same_day_funding	15	1