# ACTIVATE

This command activates a Gift or Merchandise Credit card with an initial (non-zero) balance.

**Device UI Required:** No

**Request Packet**

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| FUNCTION_TYPE | Required | Static value | N/A | N/A | PAYMENT | Type of function. |
| COMMAND | Required | Static value | N/A | N/A | ACTIVATE | Command name |
| PAYMENT_TYPE | Required | List | N/A | N/A | GIFT MERCH_CREDIT | Payment type field for Gift or Merchandise Credit. **NOTE:** PAYMENT_TYPE field is mandatory for card token based transactions. |
| PAYMENT_SUBTYPE | Optional | List | N/A | N/A | | Payment subtype field for Gift or Merchandise Credit. **Example:** BLACKHAWK |
| TRANS_AMOUNT | Required | Floating point number | 1(2) | 6(2) | | Total transaction amount. This field must be a non-zero amount. **Example:** 5.00 |
| MANUAL_ENTRY | Optional | Binary | N/A | N/A | TRUE FALSE | This field defines the transaction mode as Manual if it is set to TRUE. This instructs Point to collect the account information through the keypad on the device. |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| MANUAL_PROMPT_OPTIONS | Optional | Character | 1 | 50 | NOEXP | This field is applicable if MANUAL_ENTRY = TRUE. When this is present, SCA will not prompt for expiration. |
| ENCRYPT | Conditional | Binary | N/A | N/A | TRUE FALSE | This field is required to encrypt the PAN details before passing it on to processor/gateway. In case of P2PE encryption, this field value will be TRUE as default value. **NOTE:** If this field is not present, then the application will internally treat this field as a value TRUE when the device encryption is ADE/VSD. |
| POS_RECON | Optional | Character | 1 | 30 | | POS reconciliation. POS Reconciliation field to be echoed back in response to POS. **Example:** RetailPOS1 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|-------|------|------|---------|---------|----------|-------------|
| BANK_USERDATA | Conditional | Character | 1 | 50 | | Returned with CARD_TOKEN. Whatever comes back with BANK_USERDATA in the response for the token should also be sent in the request. **Example:** 01/00/02/Visa/ |
| CARD_TOKEN | Conditional | Character | 1 | 40 | | Card token is processor-based or gateway-based and can represent a unique card. Refer to Two Way Card Token section. **Example:** 7987654321098 765 |
| TKN_RENEW | Conditional | Character | | 1 | Valid value: 1 | Application will send this field to the Gateway, requesting for Token renewal. As of this publication, this is applicable for UGP only. |
| CASHIER_ID | Optional | Character | 1 | 10 | | This indicates the Cashier ID performing the transaction ID. **Example:** 7987654321098 765 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| CAPTURECARD_EARLYRETURN | Optional | Binary | N/A | N/A | TRUE FALSE | If the sending value is TRUE, then the application returns card data to POS before processing. PCI BIN checking in place to return full PAN or masked PAN BIN range level. **NOTE:** SCA will cache data from the swipe, however, will only use in immediately subsequent CAPTURE request containing explicit tender type – else, it will discard. |
| EMV_TAGS_REQD | Character | Binary | | | Valid values: Y/N | EMV tags detail required. This field is sent in request to return the EMV tags in the response, only in case of CAPTURECARD_EARLYRETURN is sent as TRUE. |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| CDD_DATA | Optional | Character | 1 | 30 | | Customer Defined Data. This field is optional, and the datatype is String. It is a pass-through field, and it is passed in the host request if this field is present in the POS request and also returned in POS response. This field is applicable for all payment transactions. **Example:** `<CDD_DATA>INV200471</CDD_DATA>` |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| COL_3, COL_4, COL_5, COL_6, COL_7, COL_8, COL_9, COL_10 | Optional | Character | 1 | 255 | | These fields represent Column 3 to Column 10. These fields are expected for the Merchants internal POS System, which will record any additional data and link those to the PWC CLIENT_ID and CTROUTD. When a value for COL_n is passed in, that same value will be returned in the response. These COL_n values are not indexed, or searchable in any command report. These fields are not sent to any payment processor. These fields are sent in SAF transaction request.<br>**Example:** Merchant defined data |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| INVOICE | Required | Character | 1 | 40 | | Merchant invoice number. Maximum 40 characters (A-Z, a-z, 0-9) and these are not case sensitive. All the special characters are supported in INVOICE. **NOTE:** POS, integrated with the application should handle maximum invoice applicable for the host used in the environment. Vantiv allows numeric only; the value may not be all zeroes. **Example:** TA1234 |
| LANE | Optional | Numeric | 1 | 8 | | This field is used to identify the retail lane. **Example:** 1 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| NOTIFY_SCA_EVENTS | Optional | Boolean | | | TRUE FALSE | If this field is set to TRUE, then it returns event broadcasts as play by play of what is occurring on the device when interacting with consumer and host as Unsolicited Consumer Selection response. This is applicable to Vantiv Direct Engage Devices Only (as of this publication). **NOTE:** POS_IP and POS_PORT must be sent. Broadcast events should be used as information only and not as a condition for stopping or resuming a transaction. The exception would be in case where consumer interaction has lapsed for certain duration and POS elects to CANCEL on secondary port. |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| POS_IP | Optional | Character | | | | This indicates the POS listening IP address. This is for Consumer Unsolicited responses to POS. **Example:** 192.168.31.100 |
| POS_PORT | Optional | Numeric | 4 | 4 | | This indicates the POS listening port. This is for Consumer Unsolicited responses to POS. **Example:** 5016 |
| PURCHASE_ID | Conditional | Character | 1 | 25 | | Purchase ID. This is required for Level II processing. P.O. Number or Customer Code. All the special characters are supported in PURCHASE_ID. **Example:** 1 |
| SERVER_ID | Optional | Numeric | 1 | 10 | | This indicates the Server ID, performing the transaction. **Example:** 560 |
| SHIFT_ID | Optional | Character | 1 | 1 | | This indicates the Shifts at the store. **Example:** 2 |
| STORE_NUM | Optional | Character | 1 | 6 | | Store number. **Example:** 203 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|-------|------|------|---------|---------|----------|-------------|
| TABLE_NUM | Optional | Numeric | 1 | 5 | | Table number. Collected for receipts. **Example:** 10 |
| TRAINING_MODE | Optional | List | 1 | 3 | OFF ON | This field is included to turn on Training Mode for the session. Transactions are routed to HIF Test for host simulation and results are mocked for approvals. **NOTE:** When DEMO parameter is 1 (enabled), transactions will be performed in Training Mode without the need to pass <TRAINING_MODE>ON</TRAINING_MODE> from POS. |
| COUNTER | Required | Numeric | 1 | 10 | | COUNTER is used for a given MAC label. Each COUNTER should be higher than the last one. This is used to authenticate the POS. **Example:** 100 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|-------|------|------|---------|---------|----------|-------------|
| MAC | Required | Base64 Encoded Data | N/A | N/A | N/A | Message Authentication Code. This is used to authenticate the POS. |
| MAC_LABEL | Required | Character | 1 | 50 | | Associated label that tells the device which MAC_KEY to use to decrypt the value of MAC. This is used to authenticate the POS. **Example:** REG1 |

**Keyed Account Information**

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|-------|------|------|---------|---------|----------|-------------|
| ACCT_NUM | Optional | Numeric | 1 | 25 | PAYMENT | This field is used to enter the account number manually. For this MANUAL_ENTRY must be set to TRUE. Pre-swipe data will not be honored. **Example:** 6782345678131 3 |

| Field | Rule | Type | Minimum | Maximum | Value(s) | Description |
|---|---|---|---|---|---|---|
| CARD_EXP_MONTH | Required | Numeric | 2 | 2 | | Card expiry month. **NOTE:** If the encryption is set to TRUE, then SCI will use 12 as default value if this field is not passed. **Example:** 12 |
| CARD_EXP_YEAR | Required | Numeric | 2 | 2 | | Card expiry year. **NOTE:** If encryption is set to TRUE, SCI will use 49 as default value if this field is not passed. **Example:** 49 |
| BARCODE | Optional | Character | 1 | 100 | | Barcode scanning option. |
| PIN_CODE | Required | Numeric | 1 | 12 | | Gift PIN code. **Example:** 5.00 |
| CVV2 | Optional | Numeric | 1 | 10 | | Card Verification Value 2. |
| SAF_FLAG | Optional | Boolean | N/A | N/A | | Upon returning 15 or 74 result code to POS, if customer decides to SAF the Activation Process, then activate command should be resend with this extra field <SAF_FLAG>. |

**Example**

Following is an example of request packet

```
<TRANSACTION>
<FUNCTION_TYPE>PAYMENT</FUNCTION_TYPE>
<COMMAND>ACTIVATE</COMMAND>
<COUNTER>1</COUNTER>
<MAC> … </MAC>
<MAC_LABEL>REG2</MAC_LABEL>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<PAYMENT_SUBTYPE>BLACKHAWK</PAYMENT_SUBTYPE>
<BARCODE>88590991979</BARCODE>
<ENCRYPT>TRUE</ENCRYPT>
<TRANS_AMOUNT>100.00</TRANS_AMOUNT>
</TRANSACTION>
```

Following is an example of request packet - **First leg(Capture Card Early Return)**

```
TRANSACTION>
<FUNCTION_TYPE>PAYMENT</FUNCTION_TYPE>
<COMMAND>ACTIVATE</COMMAND>
<TRANS_AMOUNT>100.00</TRANS_AMOUNT>
<CAPTURECARD_EARLYRETURN>TRUE</CAPTURECARD_EARLYRETURN>
<MANUAL_ENTRY>FALSE</MANUAL_ENTRY>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<FORCE_FLAG>FALSE</FORCE_FLAG>
<MAC_LABEL>P_EJIOKG</MAC_LABEL>
<COUNTER>14</COUNTER>
<MAC>pWMpzmKh4CwI1OYFRtWOK1Xw332YUoGzbROyaMku6ZI=</MAC>
</TRANSACTION>
```

Following is an example of request packet - **Second leg(Capture Card Early Return)**

```
<TRANSACTION>
<FUNCTION_TYPE>PAYMENT</FUNCTION_TYPE>
<COMMAND>ACTIVATE</COMMAND>
<TRANS_AMOUNT>100.00</TRANS_AMOUNT>
<MANUAL_ENTRY>FALSE</MANUAL_ENTRY>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<FORCE_FLAG>FALSE</FORCE_FLAG>
<MAC_LABEL>P_EJIOKG</MAC_LABEL>
<COUNTER>15</COUNTER>
<MAC>Dqutu+H7fEndQ4D/O9KJ/xGMXiy/8n8WJ1psDVpwLH4=</MAC>
</TRANSACTION>
```

**Response Packet**

| Field | Type | Value | Description |
|-------|------|-------|-------------|

| | | | |
|---|---|---|---|
| RESPONSE_TEXT | Character | | Processor response text. **Example:** CAPTURED. |
| RESULT | Character | | This indicates the Result details. Commonly CAPTURED or DECLINED. |
| RESULT_CODE | Numeric | Expected result code: 4, 59074, 59040 | This indicates the result code. |
| TERMINATION_STATUS | Character | SUCCESS and FAILURE | This indicates the transaction termination status. This is the overall status of the transaction irrespective of approved or declined. Like, if the output is generated then the status is SUCCESS and if no output is generated then the status will be FAILURE. |
| BATCH_TRACE_ID | Character | | Batch Trace ID. This is conditional field. **Example:** 12cc7b17-4b45-4344-b412-5432 |
| EMBOSSED_ACCT_NUM | Numeric | | Card number conditionally returned if present in the SSI response. Returned if payment type = GIFT and returnembossednumforgift is enabled. **Example:** 6499991111115789 |
| COUNTER | Numeric | | Echoes counter sent in the request. **Example:** 100 |
| MERCHID | Numeric | | Merchant ID. **Example:** 900000000123 |
| TERMID | Numeric | | Merchant ID. **Example:** 001 |

| | | | |
|---|---|---|---|
| TRANS_SEQ_NUM | Numeric | | Processor/Batch transaction sequence number. **NOTE:** For private label transaction (ADS), PT_SEQ_NUM field will be mapped to TRANS_SEQ_NUM and TROUTD fields back to SCA. **Example:** 5 |
| INTRN_SEQ_NUM | Numeric | | PWC transaction ID. **Example:** 123456789 |
| AUTH_CODE | Character | | Processor authorization number. **Example:** 123456 |
| TROUTD | Numeric | | Transaction routing ID. **Example:** 123456789 |
| CTROUTD | Numeric | | CTROUTD is a sequence number for PAYMENT transactions (always enabled) that is generated per Client ID. Each Client ID has its own CTROUTD sequence counter. **NOTE:** For private label transaction (ADS), PT_CTROUTD field will be mapped to CTROUTD field back to SCA. **Example:** 45 |
| LPTOKEN | Numeric | | LP Token is a non-sensitive unique number assigned to each unique card number processed with the UGP gateway. This value will automatically increment by one for each unique card number. This is a conditional field. **NOTE:** Refer to Responses from Point section in Message Format. **Example:** 12457 |
| PAYMENT_MEDIA | Character | | Mode of payment. Commonly VISA/MC/DISC/AMEX/DEBIT. **Example:** : GIFT/MERCHANDISE |
| PAYMENT_TYPE | Character | | Payment type returned, like Gift. **Example:** GIFT/MERCH_CREDIT |

| | | | |
|---|---|---|---|
| ACCT_NUM | Numeric | | Returned the masked account number. **NOTE:** If UNMASKEDPANFORNONPCI=1 then the account number will be sent back to POS as unmasked for non PCI cards. Refer to GSC Parameters for more details on the parameter. **Example:** 600649******9147 |
| CARDHOLDER | Character | | Returned for swiped transactions. **Example:** TEST PROCESSOR |
| CARD_EXP_MONTH | Numeric | | Card expiry month. **Example:** 12 |
| CARD_EXP_YEAR | Numeric | | Card expiry year. **Example:** 20 |
| CARD_ENTRY_MODE | Character | | Returns card entry mode values. **Example:** 123123 **NOTE:** Refer to Card Entry Mode for details on possible values. |
| EMV_TAGS | Character | | This is returned for Early Card Capture payment flows for Non PCI BIN ranges, only when EMV_TAGS_REQD is sent as Y. |
| CARD_CLASS | Numeric | | This field is returned to identify the card type of the gift transaction. **Example:** 0 |
| PIN_CODE | Numeric | | Gift PIN code. This is a conditional field. This field will return in POS response if GIFTPINTOPOS parameter is enabled. Refer to Application Parameters for more details on the parameter. |
| APPROVED_AMOUNT | Floating point number | | The amount which got approved. **Example:** 5.00. |

| AVAILABLE_BALANCE | Floating point number | | Available balance on the card used for transaction. This field will be returned to POS, when the Host returns the Available Balance data. SCA application sends <BALANCE_ENQ> as Host request field and based on the processor, it returns the Available Balance, and SCA will send it back to POS. **Example:** 100.00. |
|---|---|---|---|
| PREVIOUS_BALANCE | Floating point number | | Previous balance on card. **Example:** 200.00. |
| RECEIPT_DATA | Character | | Receipt Data. |
| TRANS_DATE | Character | | Transaction date returned. **Example:** 2016.09.20 |
| TRANS_TIME | Character | | Transaction time returned. **Example:** 09:16:25 |
| VSP_CODE | Numeric | | If present, returns the VSP code. **Example:** 100 |
| VSP_RESULTDESC | Boolean | SUCCESS or FAILURE | If present, returns the VSP result description. |
| VSP_TRXID | Numeric | | If present, returns the VSP transaction ID. **Example:** 012345678901234567 |

| | | | |
|---|---|---|---|
| TRAN_LANG_CODE | Character | en – English<br>fr – French<br>es – Spanish | This field contains the language code for the current transaction which is finalized based on the configured language on terminal and language preference from the card. This field will be returned only whenever the Card data is captured from cardholder during transaction flow. If Language code is not available from card, then terminal language will be returned. This field needs to be added for the below transaction flows. |
| TRANS_CURRENCY_CODE | Numeric | | This is the currency code of the transaction. This field is sent from POS to identify if it is US or Canada transaction. **Example:**<br>• For USA, POS response is: `<TRANS_CURRENCY_CODE>0840</TRANS_CURRENCY_CODE>`<br>• For Canada, POS response: `<TRANS_CURRENCY_CODE>0124</TRANS_CURRENCY_CODE>` |
| AUTH_REF_NUMBER | Character | Example: 123456789012345 Or It can be empty | This tag returns in the host response with the value for the particular transaction. This is used by some merchants to refer to the transaction at the host side. Currently this is applicable only for Worldpay processor. |
| COL_3, COL_4, COL_5, COL_6, COL_7, COL_8, COL_9, COL_10 | Character | | Column 3 to Column 10 fields value will be echoed in POS response. These fields are not sent to any payment processor. |

**Processor-Based Token (Conditional)**

| Field | Type | Value | Description |
|---|---|---|---|
| CARD_TOKEN | Character | | Card Token field is returned in most of the GIFT administrative transactions. **NOTE:** Refer to Card Tokens section in Point Integration Best Practices. **Example:** 7987654321098765 |
| TOKEN_SOURCE | Character | | Source of the token. **Example:** PWC |

**Example**

Following is an example of response packet

```
<RESPONSE>
<APPROVED_AMOUNT>100.00</APPROVED_AMOUNT>
<AVAILABLE_BALANCE>100.00</AVAILABLE_BALANCE>
<ACCT_NUM>600649******9147</ACCT_NUM>
<AUTH_CODE>ABC001</AUTH_CODE>
<CARDHOLDER>PROCESSOR/GIFT</CARDHOLDER>
<CTROUTD>141</CTROUTD>
<INTRN_SEQ_NUM>569230</INTRN_SEQ_NUM>
<PAYMENT_MEDIA>GIFT</PAYMENT_MEDIA>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<RESPONSE_TEXT>CAPTURED</RESPONSE_TEXT>
<RESULT>CAPTURED</RESULT>
<RESULT_CODE>4</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<TRANS_SEQ_NUM>19</TRANS_SEQ_NUM>
<TROUTD>569230</TROUTD>
<TRAN_LANG_CODE>en</TRAN_LANG_CODE>
</RESPONSE>
```

Following is an example of response packet - **first leg(Capture Card Early Return)**

```
<RESPONSE>
<RESPONSE_TEXT>CAPTURE EARLY CARD NOTIFICATION</RESPONSE_TEXT>
```

```
<RESULT>OK</RESULT>
<RESULT_CODE>-1</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<COUNTER>14</COUNTER>
<CARD_TRACK1>B60105*********************************************************** ?.
</CARD_TRACK1>
<CARD_TRACK2>601056*************************4680</CARD_TRACK2>
<ACCT_NUM>601056******6057</ACCT_NUM>
<TRANS_AMOUNT>100.00</TRANS_AMOUNT>
<CARD_EXP_MONTH>**</CARD_EXP_MONTH>
<CARD_EXP_YEAR>**</CARD_EXP_YEAR>
<CARDHOLDER>RAPI*****************</CARDHOLDER>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<PAYMENT_MEDIA>GIFT</PAYMENT_MEDIA>
<CARD_ENTRY_MODE>Swiped</CARD_ENTRY_MODE>
<INVOICE>123456</INVOICE>
</RESPONSE>
```

Following is an example of response packet - **second leg(Capture Card Early Return)**

```
<RESPONSE>
<RESPONSE_TEXT>04: Inactive account. </RESPONSE_TEXT>
<RESULT>DECLINED</RESULT>
<RESULT_CODE>6</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<COMMAND>ACTIVATE</COMMAND>
<BATCH_TRACE_ID>0954ba24-3c56-4c53-b1c1-35a7bc48909a</BATCH_TRACE_ID>
<TROUTD>4016100327</TROUTD>
<CTROUTD>75065</CTROUTD>
<CARD_ENTRY_MODE>Swiped</CARD_ENTRY_MODE>
<EMBOSSED_ACCT_NUM>286831******4680</EMBOSSED_ACCT_NUM>
<TRANS_SEQ_NUM>20927</TRANS_SEQ_NUM>
<INTRN_SEQ_NUM>4016100327</INTRN_SEQ_NUM>
<TRANS_DATE>2022.07.12</TRANS_DATE>
<TRAN_LANG_CODE>en</TRAN_LANG_CODE>
<TRANS_TIME>02:40:21</TRANS_TIME>
<CARD_EXP_MONTH>**</CARD_EXP_MONTH>
<CARD_EXP_YEAR>**</CARD_EXP_YEAR>
<PAYMENT_MEDIA>GIFT</PAYMENT_MEDIA>
<PAYMENT_TYPE>GIFT</PAYMENT_TYPE>
<ACCT_NUM>601056******6057</ACCT_NUM>
<CARDHOLDER>RAPI*****************</CARDHOLDER>
<TRANS_AMOUNT>100.00</TRANS_AMOUNT>
<BANK_USERDATA>GIFT</BANK_USERDATA>
<VSP_CODE>200</VSP_CODE>
<VSP_RESULTDESC>[204] BIN Excluded from encryption</VSP_RESULTDESC>
<VSP_TRXID>637932048220875278</VSP_TRXID>
<CARD_ABBRV>GF</CARD_ABBRV>
<TRAINING_MODE>OFF</TRAINING_MODE>
<COUNTER>15</COUNTER>
</RESPONSE>
```