Process payments using third-party network token processing

DOCS CURRENT ADVANCED Last updated: September 6th 2023, @ 12:43:42 pm

PayPal supports third-party network token processing for merchants and partners.

Availability

This integration is available:

- In the US.
- In beta in the following countries:
  - Australia
  - Canada
  - European Union
  - United Kingdom

For more information, contact Sales.

Tokenization

Tokenization keeps payment information private by turning card numbers into unique tokens, which are stored securely and used instead of the original card.

Tokenization creates a unique credential, or token, for a card that is different from its 15- or 16-digit primary account number. The merchant only sends the token, rather than the underlying account number. A network token only works for a specific card and merchant.

The benefits of using network tokens include:

- Improved authorization rates.
- Increased security by reducing opportunities for data theft and using cryptograms to protect credentials.
- Potentially reduced transaction-processing costs.
- Simplified payment processing.
- Helps maintain Payment Card Industry Data Security Standard compliance.

Third-party network token

A third-party network token represents a payment method that is either:

- Saved in-house by the partner or merchant.
- Saved by an external Token Service Provider (TSP).

Third-party network token processing happens when PayPal processes a payment using a token that PayPal didn't create.

How it works

Integrate with third-party network token payments as follows:

1. Save and tokenize a payer's payment method.
2. Get a token number and expiration date to use for payments.
3. Use this token when sending a payment through PayPal.
4. PayPal processes the payment as a regular credit or debit card purchase.

Note: Third-party tokens aren't stored and can't be created, mapped, unmapped, or validated against their primary account number.

Know before you code

- You'll need an advanced credit and debit card payments integration.
- You acknowledge and agree that you are solely responsible for any third-party vaulting functionality (not provided by PayPal) that you use and PayPal will, under no circumstances, be responsible or liable for any damages, losses or costs whatsoever suffered or incurred by you as a result of using a third-party vaulting functionality on PayPal's platforms (including Products), services or APIs.

Note: Third-party network token integrations don't support reference or future transactions. Don't pass a reference transaction ID using the `payment_source.token.id` parameter.

Using third-party network tokens with PayPal

Review this section to learn how to integrate third-party network tokens in your PayPal integration.

This code sample shows a third-party network token in the body of a `POST` call to the Create order endpoint of the Orders v2 API. This request creates a new order and completes the payment in a single step by declaring the `intent` as `CAPTURE`:

Sample request

The payment request includes the new `network_token` and `stored_credential` objects:

```
curl -v -X POST https://api-m.sandbox.paypal.com/v2/checkout/orders \
-H 'Content-Type: application/json' \
-H 'PayPal-Request-Id: REQUEST-ID' \
-H 'Authorization: Bearer ACCESS-TOKEN' \
-d '{
  "intent": "CAPTURE",
  "purchase_units": [
    {
      "reference_id": "REFID-000-1001",
      "amount": {
        "currency_code": "USD",
        "value": "100.00"
      }
    }
  ],
  "payment_source": {
      "card": {
          "name": "Firstname Lastname",
          "network_token": {
              "number": "4420567383232198",
              "expiry": "2030-11",
              "eci_flag": "NON_3D_SECURE_TRANSACTION",
              "token_requestor_id": "12324"
          },
          "stored_credential": {
              "payment_initiator": "MERCHANT",
              "payment_type": "UNSCHEDULED",
              "usage": "SUBSEQUENT",
              "previous_network_transaction_reference": {
                  "id": "123456789016848",
                  "network": "VISA"
              }
          }
      }
  }'

```

- Lines 19-24: The `payment_source.card.network_token` object contains details about the third-party network token. PayPal passes this information to the issuer. See `network_token` for more details.
- Line 22: Token service providers give each third-party network token a 2-digit Electronic Commerce Indicator (ECI) code. When you make a payment using a third-party network token, your integration needs to change the 2-digit ECI code to the corresponding string from the table below. Pass this value using the `payment_source.card.network_token.eci_flag` parameter. This value is required for customer-initiated payments and optional for merchant-initiated payments:

Numeric ECI code	String
`00`	`MASTERCARD_NON_3D_SECURE_TRANSACTION`
`07`	`NON_3D_SECURE_TRANSACTION`

- Lines 25-33: The `payment_source.card.stored_credential` object contains details about the type of card-on-file payment. See `stored_credential` for more details.

Sample response

The HTTP `201` response includes the new `bin_details` and `network_transaction_reference` objects:

```
{
    "id": "1C882901H6992113A",
    "status": "COMPLETED",
```

```
    "payment_source": {
        "card": {
            "name": "Firstname Lastname",
            "last_digits": "7287",
            "expiry": "2030-11",
            "brand": "VISA",
            "available_networks": [
                "VISA"
            ],
            "type": "CREDIT",
            "bin_details": {
                "bin": "43999450",
                "issuing_bank": "CREDIT UNION OF OHIO",
                "bin_country_code": "US",
                "products": [
                      "CONSUMER"
                ]
             }
        }
    },
    "purchase_units": [
        {
            "reference_id": "REFID-000-1001",
            "payment_instruction": {
                ...
            },
            "shipping": {
                ...
            },
            "payments": {
                "captures": [
                    {
                        "id": "3BJ29575M0175253F",
                        "status": "COMPLETED",
                        "amount": {
                            "currency_code": "USD",
                            "value": "50.00"
                        },
                        "final_capture": true,
                        "disbursement_mode": "DELAYED",
                        "seller_protection": {
                            "status": "NOT_ELIGIBLE"
                        },
                        "seller_receivable_breakdown": {
                            ...
                        },
                        "invoice_id": "INVID-21-07-2023-05-56-55",
                        "custom_id": "CUSTOMID-1001",
                        "links": [
                            ...
                        ],
                        "create_time": "2023-07-21T12:27:00Z",
                        "update_time": "2023-07-21T12:27:00Z",
                        "processor_response": {
                            "avs_code": "Y",
                            "cvv_code": "X",
                            "response_code": "0000"
                        },
                        "network_transaction_reference": {
                            "id": "054706134140378",
                            "network": "VISA"
                        }
                    }
                ]
            }
        }
    ],
    "links": [
        ...
    ]
}
```

- Lines 14-18: The `payment_source.card.bin_details` object contains the bank identification number (BIN) information. See `bin_details` for more details.
- Lines 59-62: The `purchase_units.payments.captures.network_transaction_reference` object includes the `id` and `network` name. See `network_transaction_reference` for more details.