Ecommerce API Guide v2.2

Simplified 3DS Integration

January 20, 2022

Table of Contents

Change Log

Document Version	Description	Release Date
1.0	Initial Version	September 25, 2021
2.0	Final Draft	December 10, 2021
2.1	Added PowerTranz-GatewayKey to section 4 Added numbering throughout document Removed PowerTranzId and PowerTranzPassword from section 7.3	December 24, 2021
2.2	- Added clarification to allowed values for CardholderName , FirstName ,LastName, Line1, Line2 ,City, State, PhoneNumber, PostalCode in section 5 - Removed address reference in section 9.4 - Removed Token from section 6 and added to section 5 - Replaced token with SpiToken in section 2.2 and 2.3 - Added format to PanToken in section 6 - Added Tokenize to section 5	January 20, 2022

1. Introduction

This document is a developer's guide for integrating PowerTranz payment processing within a merchant's website. This integration guide covers the simplified 3DS integration method for 3DS e-commerce transactions with or without utilizing a hosted payment page.

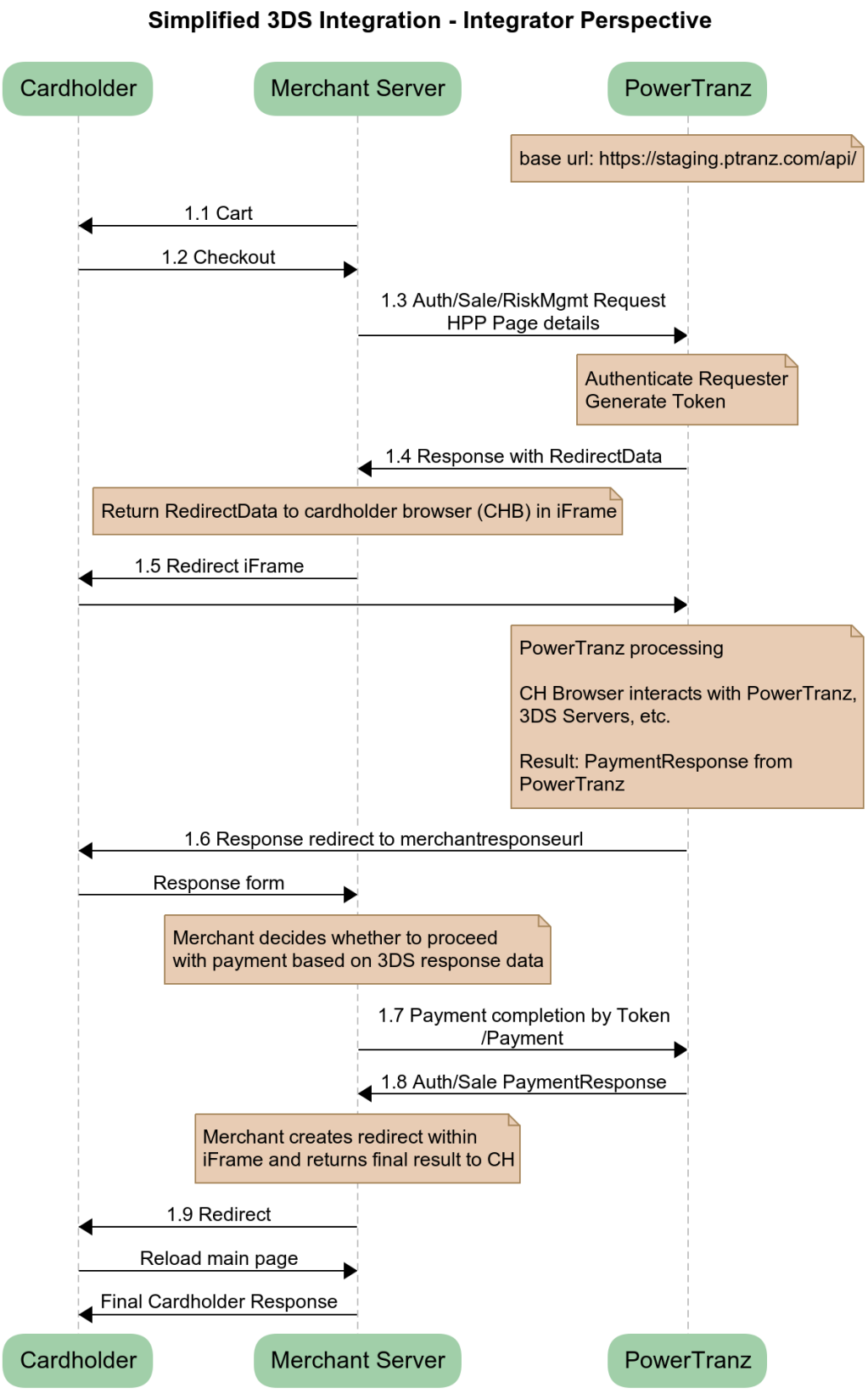
2. E-Commerce with 3D-Secure Overview

The PowerTranz gateway supports EMV 3D-Secure versions 2.x with fallback to 3DS version 1.0 for cardholder authentication and sends financial requests (authorization, sale, refund or void) to the payment networks for cardholder authorization.

A 3D-Secure Request is initiated by using the Auth, Sale or RiskMgmt API methods with the 3D-Secure flag enabled. PowerTranz will query the supported version of 3D-Secure based on the provided card number and the issuing bank's capabilities. The simplified 3DS integration method will handle the required interactions for a 3DS 2.0 authentication which may be frictionless, include device fingerprinting, a challenge flow or if 3DS 2.0 is not supported then a fallback attempt to 3DS version 1.0

Using this integration method, there will be a pre-authentication followed by a payment completion depending on the pre-authentication result. Payment information is submitted directly from the merchant's payment page or the PowerTranz hosted payment page (HPP). The transaction is processed transparently by the PowerTranz server which will notify the merchant with the 3D-Secure authentication result and the merchant will then decide whether or not to proceed with the financial request.

2.1 Simplified 3DS Integration Diagram



2.2 Simplified 3DS Integration High Level Process Flow

1.1 The merchant webserver displays the finalized shopping cart to the cardholder.
1.2 The cardholder checks out.
1.3 Depending on the integration method used:
    a. The merchant collects the cardholder payment information and sends an Auth, Sale or RiskMgmt request (that includes the relevant cardholder and payment details) with the 3DS flag enabled to the PowerTranz server; or
    b. The merchant sends an Auth, Sale or RiskMgmt request to the PowerTranz server which includes a hosted page set and name where the relevant cardholder and payment details will be collected on the hosted page.
1.4 PowerTranz authenticates the request coming from the merchant, generates a SpiToken and replies to the merchant server with Redirect Data.
1.5 The Redirect Data is contained in the response from the Auth/Sale/RiskMgmt endpoint. It contains an HTML form with JavaScript that, when injected into an iFrame, will display the hosted page (HPP) if being used or a challenge flow if required by the issuing bank. During this stage the iFrame in the cardholder browser interacts with PowerTranz and the required 3DS servers depending on the type of 3DS authentication required. This could be a fully frictionless flow or the cardholder could be presented with a challenge during this time. When complete, the iFrame is redirected to the MerchantResponseUrl and the Merchant application resumes control of the flow. See code sample in Appendix.
1.6 PowerTranz responds with the 3DS authentication result to the merchant server via the cardholder browser. Note that this is not a financial transaction and is the result of the 3DS authentication only.
1.7 Based on the 3DS authentication result, the merchant determines if they want to proceed with payment Completion. If the merchant chooses to proceed with the transaction, a payment completion is sent using the SpiToken and sending a Payment request. The authorization request is then sent from the PowerTranz server to the processor and on the issuing bank.
1.8 PowerTranz returns the Auth/Sale payment response to the merchant server.
1.9 The merchant server then displays the final results to the cardholder browser. If the merchant originally called a Sale, the financial transaction is now complete and then, following settlement (controlled by PowerTranz), the cardholder will be billed and the merchant account will be credited. If the merchant called an Auth, there will be a hold on funds but a Capture must be sent when the merchant is ready to finalize the transaction and bill the cardholder.

2.3 Merchant API Calls – Additional Details

Within the simplified 3DS implementation, the merchant will make multiple calls to endpoints in the PowerTranz API. The first request (Auth, Sale or RiskMgmt) will initiate the authentication process and return an authorization SpiToken to be used in subsequent requests. Subsequent, optional, requests can then be made to the "Payment", "Capture", "Void" and "Refund" endpoints to either cancel or complete the transaction as required.

A 3D-Secure **authentication** request is initiated by merchant using "Auth/Sale/RiskMgmt" REST endpoints with the 3DS flag enabled

- "/Auth" or "/Sale" financial request endpoint is called when merchant is intending to perform an online financial request after 3DS authentication. If the Payment completion is successful, an Auth requires a follow up capture to be sent to complete the transactions. A Sale will both Authorize and flag the transaction to be captured without an additional call.

- "/RiskMgmt" non-financial request endpoint is called only when merchant intends to authenticate card holder (3DS2 Authentication only).

If merchant decides to proceed with an online **authorization** (for financial requests) payment transaction is initiated by calling "/Payment".

Notes:

- During Auth/Sale/RiskMgmt call, merchant should pass "MerchantResponseURL" which is the merchant server endpoint that PowerTranz will send final transaction result.
- Calls to the PowerTranz API are performed by using REST with JSON over HTTPS as the transport protocol.
- Externally accessible BASE URLs for the PowerTranz SPI/HPP endpoints are:

  Staging: https://staging.ptranz.com/api/spi/<endpoint>

  Prod: https://*TBD*.ptranz.com/api/spi/<endpoint>

- Merchants can call "/Capture", "/Void" or "/Refund" for a successfully authorized transaction. External base URLs for these endpoints are:

  Staging: https://staging.ptranz.com/api/<endpoint>

  Prod: https://*TBD*.ptranz.com/api/<endpoint>

3. PowerTranz Gateway Endpoints and Operations

PowerTranz exposes a set of financial and nonfinancial endpoints for merchant transaction processing. The table below shows endpoints with a brief description of their usage and their URL.

Endpoint	Description	Type	Method	URL
Alive	Gateway status	Non-financial	GET	<API Root>/api/alive
Auth	Performs an SPI authorization securing funds for later capture.	Financial	POST	<API Root>/api/spi/auth
Sale	Performs an SPI authorization with capture.	Financial		<API Root>/api/spi/sale
RiskMgmt	Non-financial transaction. Use this to pre-authenticate a 3DS only transaction.	Non-financial	POST	<API Root>/api/spi/riskmgmt
Payment	Payment Completion for 3DS pre-authenticated sale or authorization transactions.	Financial	POST	<API Root>/api/spi/payment
Capture	Capture a previously authorized transaction.	Financial	POST	<API Root>/api/capture
Refund	Refund a previously authorized transaction.	Financial	POST	<API Root>/api/refund
Void	Void an authorization.	Financial	POST	<API Root>/api/void

The Swagger page for PowerTranz API provides parameter information in JSON format.

https://staging.ptranz.com/api/swagger/index.html

4. PowerTranz Request Header Requirements

All requests to endpoints are HTTP POST requests over TLS with JSON payloads in the body. It is mandatory that the http header includes merchant authentication parameters (e.g. PowerTranzId and Password). Merchants should call PowerTranz API endpoints using a HTTP POST and send request parameters in JSON format.

Field Name	Req	Format	Length Max/Value	Notes
PowerTranz-PowerTranzId	M	AN	25	Merchant identifier for the merchant's account with PowerTranz. Example : 99901066
PowerTranz-PowerTranzPassword	M	AN	100	This is the merchant's unique processing password. Example : m9mOPK@vpUM
PowerTranz-GatewayKey	C	GUID (string)	36	Additional token assigned by Powertranz Do not send until value is provided by PowerTranz

5. Request Parameters details– Auth, Sale, RiskMgmt

(M)andatory, (O)ptional, (C)onditional

Parameter Name	Req	Format	Length Max/Value	Description
TransactionIdentifier	M	GUID (string)	36	Unique identifier assigned by merchant application Example : f62c3e58-1983-4165-8535-fe5bb6ba6127
TotalAmount	M	DEC	18,3	Total authentication amount
CurrencyCode	M	N	4	Must use numeric currency code (ISO 4217)
ThreeDSecure	M	BOOL		
Tokenize	C	BOOL		PanToken will be returned if Tokenize is set to true
Source				Mandatory nested object within message body (see Data Subset below)
CardPan	M	N	19	Card number
CardCvv	O	N	4	Card verification value
CardExpiration	M	N	4	Expiry date in YYMM format
CardholderName	M	AN	2-45	Cardholder name – required for 3DS transactions -see section 9.4 for allowed characters
Token	O	AN	100	PanToken returned in previous response
OrderIdentifier	M	AN	2**55**	Order ID assigned by the merchant
BillingAddress				Mandatory nested object within message body (see Data Subset below)
FirstName	O	AN	30	First Name (note for 3DS authentication, CardholderName in Source object must be populated) -see section 9.4 for allowed characters
LastName	O	AN	30	Last Name (note for 3DS authentication, CardholderName in Source object must be populated) -see section 9.4 for allowed characters
Line1	O	AN	30	Address line 1 (required for AVS) No special characters, no accents, no special symbols (Example: æ é à ñ * + & : ;) and best to avoid all symbols but basic punctuation is acceptable such as periods and dashes (. and -)
Line2	O	AN	50	Address line 2 No special characters, no accents, no special symbols (Example: æ é à ñ * + & : ;) and best to avoid all symbols but basic punctuation is acceptable such as periods and dashes (. and -)
City	O	AN	25	City No special characters, no accents, no special symbols (Example: æ é à ñ * + & : ;) and best to avoid all symbols but basic punctuation is acceptable such as periods and dashes (. and -)
County	O	AN	25	County
State	O	AN	25	State – if supplied must be the country subdivision code defined in ISO 3166-2.

				For US addresses only correct abbreviations are allowed valid samples :FL ,CA
PostalCode	O	AN	10	Postal or Zip code (required for AVS) Strictly Alphanumeric only - No special characters, no accents, no spaces, no dashes…etc.
CountryCode	C	AN	3	Must contain valid numeric country code (ISO 3166) Must be supplied if State is populated.
EmailAddress	O	AN	50	Email address
PhoneNumber	O	AN	20	Valid phone number including country code Valid examples: 35301176543210 35301176543210 01176543210 (must include CountryCode)
PhoneNumber2	O	AN	20	Mobile phone (see above validations)
PhoneNumber3	O	AN	20	Work phone (see above validation)
ShippingAddress				Optional nested object within message body (see Data Subset below) Note the same validations for BillingAddress apply)
FirstName	O	AN	30	First Name (note for 3DS authentication, CardholderName in Source object must be populated) -see section 9.4 for allowed characters
LastName	O	AN	30	Last Name (note for 3DS authentication, CardholderName in Source object must be populated) -see section 9.4 for allowed characters
Line1	O	AN	30	Address line 1 (required for AVS) No special characters, no accents, no special symbols (Example: æ é à ñ * + & : ;) and best to avoid all symbols but basic punctuation is acceptable such as periods and dashes (. and -)
Line2	O	AN	50	Address line 2 No special characters, no accents, no special symbols (Example: æ é à ñ * + & : ;) and best to avoid all symbols but basic punctuation is acceptable such as periods and dashes (. and -)
City	O	AN	25	City
County	O	AN	25	County
State	O	AN	25	State
PostalCode	O	AN	10	Postal or Zip code (required for AVS) Strictly Alphanumeric only - No special characters, no accents, no spaces, no dashes…etc.
CountryCode	O	AN	3	Must contain valid numeric country code (ISO 4217)
EmailAddress	O	AN	50	Email address
PhoneNumber	O	AN	20	Home phone
PhoneNumber2	O	AN	20	Mobile phone
PhoneNumber3	O	AN	20	Work phone
AddressMatch	O	BOOL		If ‘true’ shipping address and billing address match
ExtendedData				Mandatory nested object within message body
ThreeDSecure				Mandatory nested object within ExtendedData (see Data Subset below)

ChallengeWindowSize	M	AN	1	Merchants preferred sized of challenge window presented to cardholder 1 – 250 x 400 2 – 390x400 3 – 500x600 4 – 600x400 5 – 100%
MerchantResponseURL	M	AN	255	Response URL for merchant
ChallengeIndicator	O	N	2	Conditional value – if supported 01 = No preference 02 = No challenge requested 03 = Challenge requested: 3DS Requestor Preference 04 = Challenge requested: Mandate Default value if not provided is that ACS would interpret as: 01 = No preference.
HostedPage				Nested object within ExtendedData (see Data Subset below) if using HPP
PageSet	O	AN	50	HPP PageSet
PageName	O	AN	50	HPP PageName

6. Response Parameters – All Transaction Types

(P)resent, (C)onditional

Parameter Name	Req	Format	Length Max/Value	Description
TransactionType	P	numeric	2	Transaction type indicator is returned (1-Auth, 2-Sale, 3-Capture, 4-Void, 5-Refund)
Approved	P	BOOL		Status of the transaction
AuthorizationCode	C	AN	6	Authorization code of the authorization or sale transaction
TransactionIdentifier	P	GUID (string)	36	Unique identifier assigned by merchant application Example : f62c3e58-1983-4165-8535-fe5bb6ba6127
TotalAmount	P	DEC	18,3	Amount of the transaction processed
CurrencyCode	P	N	3	Currency of the transaction
CardBrand	P	AN	255	Brand of the card for informational purposes
IsoResponseCode	P	AN	3	Main response code to indicate approval, decline or failure
ResponseMessage	P	AN	255	Descriptive response of IsoResponseCode
RRN	P	string	12	Retrieval reference number
OriginalTrxnIdentifier	C	GUID (string)	36	Transaction Identifier of the original transaction returned in Capture, Refund or Void response
RiskManagement				
CvvResponseCode	C			CVV2 result
ThreeDSecure	P	BOOL		
Eci	C	AN	2	Provided if AuthenticationStatus is Y or A
Cavv	C	AN	100	Provided if AuthenticationStatus is Y or A
Xid	P	AN	100	3DS transaction ID
AuthenticationStatus	P	AN	1	See possible responses here: 3DS Authentication Results
RedirectData	C	HTML Form		Contains the redirect form to send to the cardholder's browser in the case of response codes 3D4,3D5,3D6
AuthenticateUrl	C	AN	100	Required for 3DS2/Authenticate with device fingerprinting
CardEnrolled	P	AN	1	Status of card enrolment
ProtocolVersion	P	AN	8	3DS protocol version supported by the issuer
FingerprintIndicator	C	AN	1	Status of fingerprinting. Possible values U, Y or N
StatusReason	C	AN	2	Provides information on why the Transaction Status field has the specified value for N, U or R AuthenticationStatus. See possible responses here: StatusReason
DsTransID	P	AN	36	Universally unique transaction identifier assigned by the directory server to identify a single transaction.
CardholderInfo	C	AN	255	Additional information optionally provided to the cardholder from the issuer bank ACS
PanToken	C	AN	100	PAN token
OrderIdentifier	P	AN	255	OrderIdentifier from request
SpiToken	C			SPI token
Errors	C			
Code	C	AN	2	Error code
Message	C	AN	255	Descriptive test of error code
BillingAddress	C	AN		Nested object with billing information from request

7. PowerTranz 3DS2 Auth, Sale and RiskMgmt Request Examples

The Auth, Sale and RiskMgmt requests all inherit from the same base and they share the same parameters.

Note: Some parameters can/must be excluded depending on the nature of the request.

Below is a Json sample of the Auth-Payment-Capture flow that a Merchant might implement using their own payment page or the hosted payment page (HPP).

7.1 Auth Request – Merchant Payment Page

Auth Request

```
POST #AuthUrl# HTTP/1.1
Accept: application/json
PowerTranz-PowerTranzId: #PowerTranzPasswordId#
PowerTranz-PowerTranzPassword: #PowerTranzPassword#
Content-Type: application/json; charset=utf-8
Host: staging.ptranz.com
Content-Length: TBD
Expect: 100-continue
Connection: Keep-Alive

{
  "TransactionIdentifier": "5ee7d7c1-af78-4d7e-9386-
abb280822b73",
  "TotalAmount": 1,
  "CurrencyCode": "978",
  "ThreeDSecure": true,
  "Source": {
    "CardPan": "5115010000000001",
    "CardCvv": "",
    "CardExpiration": "2512",
    "CardholderName": "John Doe"
  },
  "OrderIdentifier": "INT-95e75078-7d58-40e8-8053-
c3d488f05f59-Orc 3569",
  "BillingAddress": {
    "FirstName": "John",
    "LastName": "Smith",
    "Line1": "1200 Whitewall Blvd.",
    "Line2": "Unit 15",
    "City": "Boston",
    "State": "NY",
    "PostalCode": "200341",
    "CountryCode": "840",
    "EmailAddress": "john.smith@gmail.com",
    "PhoneNumber": "211-345-6790"
  },
  "AddressMatch": false,
  "ExtendedData": {
    "ThreeDSecure": {
      "ChallengeWindowSize": 4,
      "ChallengeIndicator": "01"
    },
  "MerchantResponseUrl":
"https://localhost:5001/Final"
  }
}
```

Auth Response

```
{
  "TransactionType": 1,
  "Approved": false,
  "TransactionIdentifier": "5ee7d7c1-af78-4d7e-9386-
abb280822b73",
  "IsoResponseCode": "SP4",
  "ResponseMessage": "SPI Preprocessing complete",
  "OrderIdentifier": "INT-95e75078-7d58-40e8-8053-
c3d488f05f59-Orc 3569",
  "RedirectData": "[HTML FORM DATA TRUNCATED FOR
BREVITY]",
  "SpiToken":
"vlf80fset61e73ml9toqu2kqtn5sddelqk9r7kao51kut6h3o-
iseenw5eb"
   }
```

Notes:

The highlighted script is a self posting script, it is returned in the **RiskMgmt, Auth and Sale** response.

- The highlighted script will have to be rendered in the Card Holder Browser.
- It is recomended to include the above mentioned script in an iFrame.

iFrame	Authentication Response

iFrame
 	
iFrame – Redirect From Server to MerchantResponseURL
	{
	"TransactionType": 1,
	"Approved": false,
	"TransactionIdentifier": "5ee7d7c1-af78-4d7e-9386-abb280822b73",
	"TotalAmount": 1.00,
	"CurrencyCode": "978",
	"CardBrand": "MasterCard",
	"IsoResponseCode": "3D0",
	"ResponseMessage": "3D-Secure complete",
	"RiskManagement": {
	"ThreeDSecure": {
	"Eci": "02",
	"Cavv": " kBMAAAAAnEYBUwH06nACcJeBRfOZ",
	"Xid": " 7cac2981-3732-4ae9-a7c9-8d07ec6726f7",
	"AuthenticationStatus": "Y",
	"CardEnrolled": "Y",
	"ProtocolVersion": "2.1.0",
	"ResponseCode": "3D0"
	}
	},
	"PanToken": "1ra0yl1pp1uo9b98fqkf16d93rgw629x01rm2cpq58s82e8u03",
	"OrderIdentifier": "INT-95e75078-7d58-40e8-8053-c3d488f05f59-Orc 3569",
	"SpiToken": "vlf80fset61e73ml9toqu2kqtn5sddelqk9r7kao51kut6h3o-iseenw5eb",
	"BillingAddress": {
	"FirstName": "John",
	"LastName": "Smith",
	"Line1": "1200 Whitewall Blvd.",
	"Line2": "Unit 15",
	"City": "Boston",
	"State": "NY",
	"PostalCode": "200341",
	"CountryCode": "840",
	"EmailAddress": "john.smith@gmail.com",
	"PhoneNumber": "211-345-6790"
	}
	}

7.2 Auth Request – Hosted Payment Page (HPP)

Auth Request

```
POST #AuthUrl# HTTP/1.1
Accept: application/json
PowerTranz-PowerTranzId: #PowerTranzPasswordId#
PowerTranz-PowerTranzPassword: #PowerTranzPassword#
Content-Type: application/json; charset=utf-8
Host: staging.ptranz.com
Content-Length: TBD
Expect: 100-continue
Connection: Keep-Alive

{
  "TransactionIdentifier": "89876ff5-a44a-4e1f-bf71-
8f224823c439",
  "TotalAmount": 1,
  "CurrencyCode": "978",
  "ThreeDSecure": true,
  "Source": {
     },
  "OrderIdentifier": "INT-245d0301-5170-406c-abb7-
750aadce9173-Orc 3570",
  "BillingAddress": {
    "FirstName": "John",
    "LastName": "Smith",
    "Line1": "1200 Whitewall Blvd.",
    "Line2": "Unit 15",
    "City": "Boston",
    "State": "NY",
    "PostalCode": "200341",
    "CountryCode": "840",
    "EmailAddress": "john.smith@gmail.com",
    "PhoneNumber": "211-345-6790"
  },
  "AddressMatch": false,
  "ExtendedData": {
    "ThreeDSecure": {
      "ChallengeWindowSize": 4,
      "ChallengeIndicator": "01"
    },
    "MerchantResponseUrl":
"https://localhost:5001/Final",
    "HostedPage": {
      "PageSet": "GFRHPP",
      "PageName": "HPPBilling1"
    }
  }
}
```

Auth Response

```
{
  "TransactionType": 1,
  "Approved": false,
  "TransactionIdentifier": "89876ff5-a44a-4e1f-bf71-
8f224823c439",
  "IsoResponseCode": "SP4",
  "ResponseMessage": "SPI Preprocessing complete",
  "OrderIdentifier": "INT-245d0301-5170-406c-abb7-
750aadce9173-Orc 3570",
  "RedirectData": "[HTML FORM DATA TRUNCATED FOR
BREVITY]",
  "SpiToken":
"cq8gqlirt6ce2tmmphf09x5kxhndvh2zi25y7owm3m60fhy2l-
iseenw5eb"
   }
```

Notes:

The highlighted script is a self posting script, it is returned in the **RiskMgmt, Auth and Sale** response.

- The highlighted script will have to be rendered in the Card Holder Browser.
- It is recomended to include the above mentioned script in an iFrame.
- If the merchant is using a hosted page (HPP) the Hosted Page node will need to be included. If HPP is not being used, exclude the Hosted Page node.

7.3 Payment Completion

To complete the payment portion of the transaction, merchants should call "/payment" and pass the SpiToken in an HTTP Post request. PowerTranz will send the transaction to the relevant payment network and will reply back to merchant. Please see below sample JSON in response body data from PowerTranz. Note until the payment completion is sent there has been no financial authorization and no funds have been held. The result of the payment completion can be an issuer approval or decline or an error.

Payment Request	**Payment Response**

POST https://dev.ptranz.com/Api/spi/Payment HTTP/1.1
Host: staging.ptranz.com
Accept: text/plain
Request-Id: |8f9d8e1f-482cd3595d7a08db.
Content-Type: application/json-patch+json
Content-Length: TBD
 	
"vftpo8xbb4l36v9d63wx74uejlsfui5btkjw4yv6v4ojbcw8k-iseenw5eb"
 	
Notes:
 	
For the Payment Request, the body of the request is simply the SpiToken value and not Json.
	{
	"TransactionType": 1,
	"Approved": true,
	"AuthorizationCode": "123456",
	"TransactionIdentifier": "12c37d56-07fe-4941-be69-026981fc1dc3",
	"TotalAmount": 1,
	"CurrencyCode": "978",
	"RRN": "125315159423",
	"CardBrand": "Visa",
	"IsoResponseCode": "00",
	"ResponseMessage": "Transaction is approved.",
	"PanToken": "1d3qljq1yt4dk6vagncxd07usvjr6p6sqeo78b36bed9ebh7u8",
	"OrderIdentifier": " 912b-43ef-a2ee-2c83d4bd59d4"
	}

7.4 Capture Request

If the initial request was sent to the Sale endpoint and the Payment Completion returned an approval (IsoResponseCode: "00") then the transaction will be submitted for settlement. If it was sent to the Auth endpoint the transaction must be captured to complete the sale and bill the cardholder. Note there are additional transaction modification endpoints that can be used when required for refunds and voids.

Capture Request	**Capture Response**

POST https://dev.ptranz.com/Api/capture HTTP/1.1
Host: staging.ptranz.com
Accept: text/plain
PowerTranz-PowerTranzId: #PowerTranzPasswordId#
PowerTranz-PowerTranzPassword: #PowerTranzPassword#
Request-Id: |8f9d8e1f-482cd3595d7a08db.
Content-Type: application/json-patch+json
Content-Length: TBD
 	
{
"TransactionIdentifier": "4f3fae73-b43a-4016-ae93-24f88d98e079",
"TotalAmount": 1,
"ExternalIdentifier": "#MerchantGenerated#"
}
	{
	"OriginalTrxnIdentifier": "4f3fae73-b43a-4016-ae93-24f88d98e079",
	"TransactionType": 3,
	"Approved": true,
	"TransactionIdentifier": "4f3fae73-b43a-4016-ae93-24f88d98e079",
	"TotalAmount": 1,
	"CurrencyCode": "978",
	"RRN": "127011162582",
	"IsoResponseCode": "00",
	"ResponseMessage": "Transaction is approved",
	"ExternalIdentifier": "#MerchantGenerated#"
	}

7.5 Refund Request

Refund Request

```
POST https://dev.ptranz.com/Api/refund HTTP/1.1
Host: dev.ptranz.com
Accept: text/plain
PowerTranz-PowerTranzId: #PowerTranzPasswordId#
PowerTranz-PowerTranzPassword: #PowerTranzPassword#
Request-Id: |8f9d8e1f-482cd3595d7a08db.
Content-Type: application/json-patch+json
Content-Length: TBD

{
  "Refund": true,
  "TransactionIdentifier": "cab173a7-f75e-444b-ac42-
cc6a367b8b6b",
  "TotalAmount": 1,
  "CurrencyCode": "978",
  "Source": {
    "CardPresent": false,
    "CardEmvFallback": false,
    "ManualEntry": false,
    "Debit": false,
    "Contactless": false,
    "CardPan": "",
    "MaskedPan": ""
  },
  "TerminalCode": "",
  "TerminalSerialNumber": "",
  "ExternalIdentifier": "#MerchantGenerated#",
  "AddressMatch": false
}
```

Refund Response

```
{
  "OriginalTrxnIdentifier": "cab173a7-f75e-444b-
ac42-cc6a367b8b6b",
  "TransactionType": 5,
  "Approved": true,
  "TransactionIdentifier": "0446a902-311d-4868-8247-
e9dfbd8ea0a6",
  "TotalAmount": 1,
  "CurrencyCode": "978",
  "RRN": "127013162598",
  "IsoResponseCode": "00",
  "ResponseMessage": "Transaction is approved",
  "ExternalIdentifier": "#MerchantGenerated#"}
```

7.6 Void Request

Void Request

```
POST https://dev.ptranz.com/Api/void HTTP/1.1
Host: dev.ptranz.com
Accept: text/plain
PowerTranz-PowerTranzId: #PowerTranzPasswordId#
PowerTranz-PowerTranzPassword: #PowerTranzPassword#
Request-Id: |8f9d8e1f-482cd3595d7a08db.
Content-Type: application/json-patch+json
Content-Length: TBD

{
  "TransactionIdentifier": "67a7689d-efe0-4a21-a3c3-
cd8b55d7825f",
  "ExternalIdentifier": "#MerchantGenerated#",
  "TerminalCode": "",
  "TerminalSerialNumber": "",
  "AutoReversal": false
}
```

Void Response

```
{
  "OriginalTrxnIdentifier": "67a7689d-efe0-4a21-
a3c3-cd8b55d7825f",
  "TransactionType": 4,
  "Approved": true,
  "TransactionIdentifier": "67a7689d-efe0-4a21-a3c3-
cd8b55d7825f",
  "TotalAmount": 1,
  "CurrencyCode": "978",
  "RRN": "127011162583",
  "IsoResponseCode": "00",
  "ResponseMessage": "Transaction is approved",
  "ExternalIdentifier": "#MerchantGenerated#"
}
```

8. PowerTranz Response Parameters

As shown in the previous code samples, there are two distinct set of response codes that the merchant needs to analyze and determine next steps.

The initial response to the Auth, Sale or RiskMgmt request will return the 3DS authentication result.

8.1 3DS Authentication Response Code

The 3DS ResponseCode is generated by PowerTranz that shows the status of the 3DS authentication.

Note 3D0 means the process completed successfully but the detailed results still need to be interpreted and a decision made before determining whether or not to send a payment completion. There are also rules that may be set on a per merchant basis that determines if a payment completion will be allowed depending on the 3DS Authentication result.

ResponseCode	3DS Response	Description	Notes
3D0	Authentication Complete	3DS Complete	3DS1 and 3DS2 process complete
3D1	Authentication not available	3DS not supported for card type	Pre-authentication process complete
3D3	Authentication Error	3DS Error	Either 3DS1 or 3DS2 error

Sample in authentication response:

```
"IsoResponseCode": "3D0",
  "ResponseMessage": "3D-Secure complete",
```

8.2 3DS Authentication Result

The nested object ThreeDSecure in the authentication response shows the 3DS authentication result. Merchants should be able to interpret important field values and decide to proceed or not proceed with payment completion based on the result.

8.3 3DS Authentication Status

The table below shows possible authentication status values and their meanings. If the authentication status is N (not authenticated) the payment completion will not be permitted.

Value	Description
Y	Authentication/account verification successful

A	Attempts processing performed
N	Not authenticated/account not verified; transaction denied
U	Authentication/account verification could not be performed due to a technical or other problem
R	Authentication/account verification rejected. Issuer is rejecting and requests that authorization not be attempted.

**Note that a challenge response will only return a result of Y or N

8.4 ECI value

The Electronic Commerce Indicator (ECI) is a value returned the card associations indicating the outcome of authentication attempted on transactions enforced by 3DS.

A) Possible values returned by **Visa and American Express** are:

- ECI 05: 3DS authentication was successful.
- ECI 06: 3DS authentication was attempted.
- ECI 07: 3DS authentication failed or not available. Considered non-3DS.

B) Possible value returned by **MasterCard** and its interpretation:

- ECI 02: 3DS authentication is successful.
- ECI 01: 3DS authentication was attempted.
- ECI 00: 3DS authentication failed or not available. Considered non-3DS.

Note that an ECI value will not be returned in all cases depending on the authentication result.

8.5 Transaction Status Reason Results (StatusReason)

In the case of a failed 3DS authentication (status N) you may also get additional information from StatusReason.

Value	Description	Value	Description
01	Card authentication failed	**12**	Transaction not permitted to cardholder
02	Unknown Device	**13**	Cardholder not enrolled in service
03	Unsupported Device	**17**	High confidence
04	Exceeds authentication frequency limit	**18**	Very high confidence
05	Expired card	**19**	Exceeds ACS maximum challenges
06	Invalid card number	**17**	High confidence
07	Invalid transaction	**18**	Very high confidence
08	No Card record	**19**	Exceeds ACS maximum challenges
09	Security failure	**20**	Non-Payment transaction not supported
10	Stolen card	**21**	3RI transaction not supported
11	Suspected fraud	**22-79**	Reserved for EMVCo future use (values invalid until defined by EMVCo)

9. Special Considerations

9.1 Unsupported card Types – non3DS

Cards that do not currently support 3DS (JCB, Discover, Diners) can still be sent in the same way 3DS enabled cards are sent via the Simplified Integration Method with or without HPP. Instead of receiving a 3DS result, you will receive a 3D1 response which means 3DS is not supported and you can choose whether or not to continue with the payment completion.

9.2 Transaction and Order Identifiers

PowerTranz requires a unique ***TransactionIdentifier*** **and** ***OrderIdentifier*** for all transactions that should be generated by the merchant.

The TransactonIdentifier is a GUID format and is the unique ID within PowerTranz.

The OrderIdentifier is one of the values used in the Merchant Portal and reports and must be unique for each approved transaction.

9.3 3DS 2 and Cardholder Information

While only cardholder name is mandatory for 3DS2 transactions it is recommended to include as many of the Billing Address fields as possible. The ACS server (issuing bank's authentication server) will decide on the frictionless versus challenge flow based on a number of factors and any information provided up front can assist in a smooth authentication flow. Note that for 3DS 2 the merchant name used in the authentication must match exactly the merchant name being used in the authorization. If a 3DS authentication only transaction is being performed and the authorization is being done separately, it is the merchant's responsibility to ensure these values are being submitted correctly.

9.4 Data Validation

The EMV 3DS protocol uses the ISO 8859 common character set for allowed values. If a 3DS authentication request parameters (such as cardholder name) sent in an unsupported character set, authentication will fail.

Annex B Common Character Set

Table 36 shows the character set common to all parts of ISO/IEC 8859:

				b8	0	0	0	0	0	0	0	0
				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1		00	01	02	03	04	05	06	07
0	0	0	0	00			SP	0	@	P	`	p
0	0	0	1	01			!	1	A	Q	a	q
0	0	1	0	02			"	2	B	R	b	r
0	0	1	1	03			#	3	C	S	c	s
0	1	0	0	04			$	4	D	T	d	t
0	1	0	1	05			%	5	E	U	e	u
0	1	1	0	06			&	6	F	V	f	v
0	1	1	1	07			'	7	G	W	g	w
1	0	0	0	08			(	8	H	X	h	x
1	0	0	1	09			)	9	I	Y	i	y
1	0	1	0	10			*	:	J	Z	j	z
1	0	1	1	11			+	;	K	[	k	{
1	1	0	0	12			,	<	L	\	l	|
1	1	0	1	13			-	=	M	]	m	}
1	1	1	0	14			.	>	N	^	n	~
1	1	1	1	15			/	?	O	_	o	

Table 36: Common Character Set

10. Test Cards and Cases

There are two main process flows for 3DS - frictionless and challenge. Frictionless occurs when no cardholder interaction is required during the authentication process. Challenge flow involves a redirection of the cardholder browser to the issuer bank ACS server to complete one or more 'challenges' before the authentication result is returned. Support for fingerprinting is determined by the issuer bank ACS server and this can be included in both frictionless and challenge flows. The test cards will determine the 3DS authentication and authorization results.

Test Case	Card Number	3DS Version	PW	Notes
Authorizations will approve for the following test cases				
M1-01-YA	5115010000000018	1.0.2	3ds1	3DS 1 Fallback, Status=A
V2-01-YA	4012000000020071	2.1.0		Frictionless, Status=Y
V2-02-AA	4012000000020089	2.1.0		Frictionless, Status=A
M2-01-YA	5100270000000023	2.1.0		Frictionless, Status=Y
M2-02-RA	5100270000000072	2.1.0		Frictionless, Status=R
V2-03-YA	4012000000020006	2.1.0	3ds2	Challenge, Status=Y
M2-03-YA	5100270000000031	2.1.0	3ds2	Challenge, Status=Y
V2-04-YA	4012010000020070	2.1.0		Frictionless, Fingerprinting, Status=Y
V2-05-AA	4012010000020088	2.1.0		Frictionless, Fingerprinting, Status=A
M2-04-YA	5100271000000120	2.1.0		Frictionless, Fingerprinting, Status=Y
V2-06-YA	4012010000020005	2.1.0	3ds2	Challenge, Fingerprinting, Status=Y
V2-07-YA	4012000000020071	2.1.0	3ds2	Challenge, include ChallengeIndicator = 03
A2-01-YA *	341111000000009	2.1.0		Frictionless, Status=Y
DS-01-0A	6011111111111111	n/a		Discover
JC-01-0A	3528111111111108	n/a		JCB
Authorizations will decline or not be available for the following test cases				
V2-01-ND	4012000000020121	2.1.0		Frictionless, Status=N, Payment Completion not permitted (response code 12)
M2-01-ND	5100270000000098	2.1.0		Frictionless, Status=N, Payment Completion not permitted (response code 12)
M2-02-ND	5100270000000056	2.1.0		Challenge, Status=N, Payment Completion not permitted (response code 12)
V2-02-AD	4666666666662222	2.1.0		Frictionless, Status = A, ISO Response Code = 05, CVV Response = N
M2-03-UD	5555666666662222	2.1.0		Frictionless, Status=U, ISO Response Code = 05
V2-03-AD	4111111111119999	2.1.0		Frictionless, Status = A, ISO Response Code = 98
M2-04-AD	5111111111113333	2.1.0		Frictionless, Status = A, ISO Response Code = 05
V2-04-YD	4111111111110000	2.1.0	3ds2	Challenge, Status =Y, ISO Response Code = 91
M2-05-YD	5111111111110000	2.10	3ds2	Challenge, Status=Y, ISO Response Code = 91
DS-01-0D	6011111111111152	n/a		Discover
JC-01-0D	3528111111111157	n/a		JCB

* Validate with the Powertranz team if AMEX 3DS is supported for your account at this time

Appendix 1 – Response Codes

PowerTranz Response Code and Error Information

ISO Response Code	Response Code	Response Message	Error Detail
00		Transaction is approved.	
03	310	Invalid merchant	
05	22	Transaction is declined	Default decline
12	315	Invalid card/currency	Invalid card/currency
12	321	Processing errors	Processing errors
12	326	Invalid transaction	Host plugin field invalid: {field name}
12	330	Invalid transaction	Not permitted {field name}
12	343	Invalid transaction	Invalid merchant
12	386	Invalid transaction	Trxn is closed
12	384	Invalid transaction	Invalid refund
12	387	Duplicate transaction	Duplicate TransactionIdentifier
12	354	Invalid transaction	Crypto error
12	380	Invalid transaction	Original auth invalid
12	381	Invalid transaction	Original auth not found
12	382	Invalid transaction	Original auth invalid
12	383	Invalid transaction	Invalid amount
12	344	Invalid transaction	Merchant closed
12	345	Invalid transaction	Payment setting disabled
12	370	Transaction mismatch	Simulator transaction mismatch
12	320	Invalid transaction	Invalid test transaction
12	426	Invalid transaction	Host plugin field invalid: {field name}
12	76	Invalid transaction	Invalid SPI transaction
12	757	Invalid transaction	Hosted page not found
12	546	3DS1 error	3DS1 fallback not allowed
12	362	Invalid transaction	Invalid transaction
12	361	Invalid transaction	Invalid transaction
12	75	SPI error	SPI error
12	758	HPP error	Invalid HPP page
3D0		3D-Secure complete	
3D1		3DS not supported	3DS not supported for this card type
3D3	519	3DS1 error	3DS1 verify result error: {field name}
3D3	611	3DS system error	Preauthentication failed
3D3	618	3DS1 system error	3DS1 verify enrollment error
3D3	619	3DS1 system error	3DS1 verify result error
3D3	540	3DS2 error	3DS2 authenticate error
3D3	640	3DS2 system error	3DS2 authenticate error
3D3	518	3DS1 error	3DS1 verify enrollment error: {field name}
3D3	520	3DS1 error	Cannot build PAReq
3D3	511	3DS error	Preauthentication failed
3D3	532	3DS error	Authentication failed
3D3	444	3DS2 system error	General 3DS error
3D3	541	3DS2 error	3DS2 challenge error
3D3	641	3DS2 system error	3DS2 challenge error
3D3	542	3DS2 error	3DS2 result error
3D3	642	3DS2 system error	3DS2 result error
3D3	543	3DS2 error	3DS2 notify error
3D3	643	3DS2 system error	3DS2 notify error
3D3	544	3DS2 system error	3DS2 fingerprint error

3D3	550	3DS2 error	DS error
3D3	548	3DS error	DS comms error
3D3	551	3DS2 error	3DS Server unreachable
3D3	549	3DS error	Cache error
3D3	649	3DS2 system error	Cache error
3D3	510	3DS error	3DS invalid parameter: {field name}
57	316	Invalid card type	Invalid card type
89	312	Failed authentication	Invalid credentials
91	391	Host timeout	Host timeout
91	392	Host comms error	Host comms error
91	329	Host comms error	Host not available
96	424	System error	Internal communication error
96	44	System error	General GateApi error
96	432	System error	Missing action: {field name}
96	459	System error	Persistence error
96	460	System error	Card mapping error
96	85	System error	SPI system error
96	850	System error	HPP system error
96	325	Host processing error	Host processing error
96	332	System error	Missing route
96	317	System error	Internal timeout
96	353	System error	TLV parse failure
96	332	System error	Missing route.
96	49	System error	Indeterminate: {field name}
96	610	3DS system error	Missing 3DS parameter: {field name}
96	456	System error	RiskMgmt not operational
96	457	System error	General RiskMgmt error
96	458	System error	Invalid route
96	45	System error	General api error
96	450	System error	General gate error
96	451	System error	General processor error
96	452	System error	General processor error
96	453	System error	TLV parse failure
96	455	System error	Api not operational
96	417	System error	Internal timeout
96	42	System error	Gate not available
96	421	System errors	Multiple errors detected
96	422	Host processing error	Host plugin error
96	425	Host processing error	Host processing error
96	43	System error	Internal routing error
96	431	System error	Rule error
96	433	System error	Invalid route
97	36	Request failed validation	Invalid request
97	37	Request failed validation	Missing field(s): {field name}
97	38	Request failed validation	Field is invalid: {field name}
97	57	Request failed validation	Missing 3DS field: {field name}
97	58	Request failed validation	Invalid 3DS field: {field name}
98	428	System error	Host plugin error
99	441	System error	Response code error
99	490	General error	General error
99	390	General error	General error
99	327	Host comms error	PL error
HP0		HPP preprocessing complete	
SP4		SPI Preprocessing complete	
TK0		Tokenize complete	

Payment ISO Response Codes

Response Code & Description		Response Code & Description	
00	Approved	53	No savings account
01	Refer to issuer	54	Expired card
02	Refer to issuer (special)	55	Incorrect PIN
03	Invalid merchant	56	No card record
04	Pick-up card	57	Transaction not permitted to card
05	Do not honor	58	Transaction not permitted to card
06	Error	59	Suspected fraud
07	Pick-up card (special)	60	Card acceptor contact acquirer
08	Honor with identification	61	Exceeds withdrawal limit
09	Request in progress	62	Restricted card
10	Approved for partial amount	63	Security violation
11	VIP Approval	64	Original amount incorrect
12	Invalid transaction	65	Activity count exceeded
13	Invalid amount	66	Card acceptor call acquirer
14	Card number does not exist	67	Card pick up at ATM
15	No such issuer	68	Response received too late
16	Approved, update track 3	75	Too many wrong PIN tries
17	Customer cancellation	76	Previous message not found
18	Customer dispute	77	Data does not match original message
19	Re-enter transaction	80	Invalid date
20	Invalid response	81	Cryptographic error in PIN
21	No action taken (no match)	82	Incorrect CVV
22	Suspected malfunction	83	Unable to verify PIN
23	Unacceptable transaction fee	84	Invalid authorization life cycle
24	File update not supported by receiver	85	No reason to decline
25	Unable to locate record	86	PIN validation not possible
26	Duplicate file update record	88	Cryptographic failure
27	File update field edit error	89	Authentication failure
28	File temporarily unavailable	90	Cutoff is in process
29	File update not successful	91	Issuer or switch inoperative
30	Format error	92	No routing path
31	Issuer sign-off	93	Violation of law
32	Completed partially	94	Duplicate transmission
33	Expired card	95	Reconcile error
34	Suspected fraud	96	System malfunction
35	Card acceptor contact acquirer	97	Format Error
36	Restricted card	98	Host Unreachable
37	Card acceptor call acquirer	99	Errored Transaction
38	Allowable PIN tries exceeded	N0	Force STIP
39	No credit account	N3	Cash Service Not Available
40	Function not supported	N4	Cash request exceeds issuer limit
41	Pick-up card (lost card)	N7	Decline for CVV2 failure
42	No universal account	P2	Invalid biller information

43	Pick-up card (stolen card)	P5	PIN Change Unblock Declined
44	No investment account	P6	Unsafe PIN
51	Not sufficient funds	XA	Forward to issuer
52	No checking account	XD	Forward to issuer

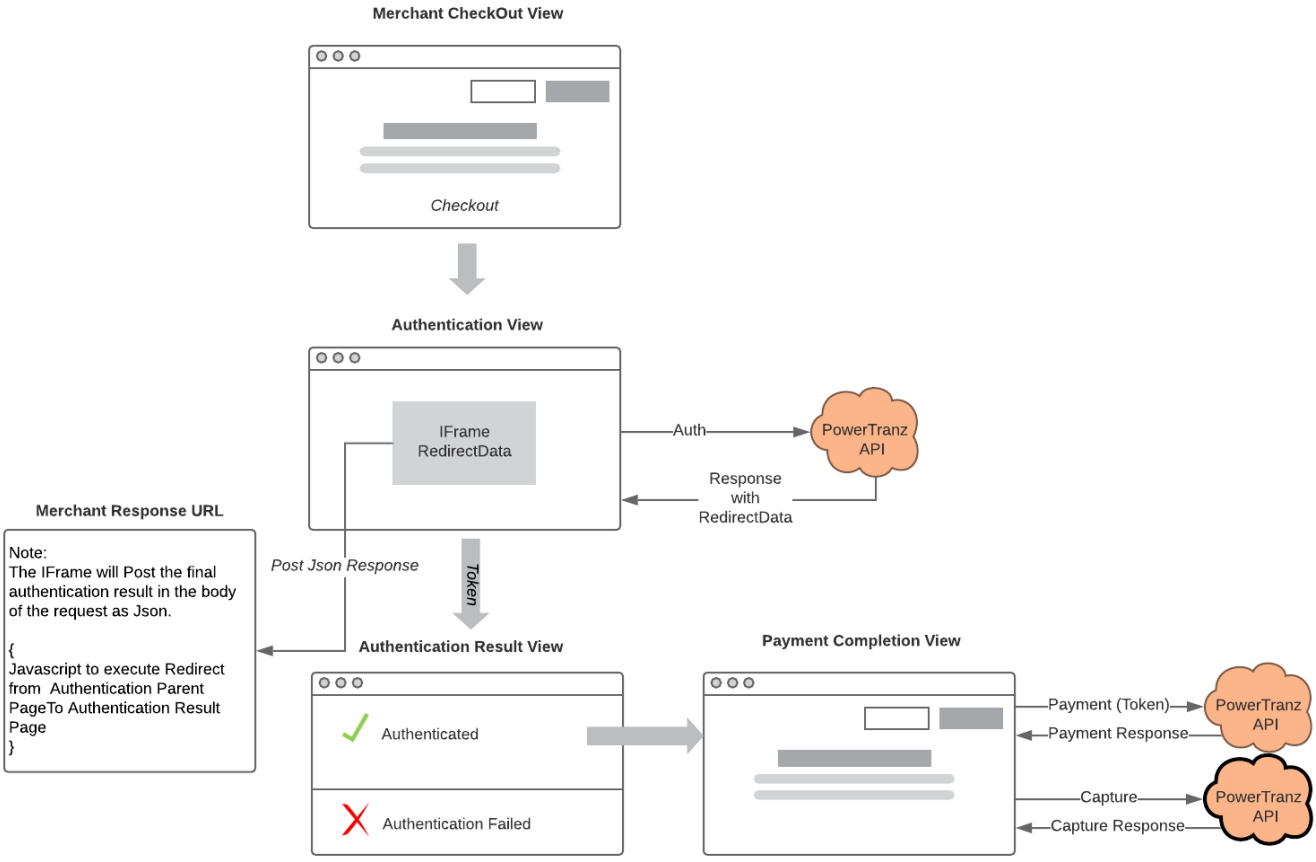
CVV2 Response Codes

Code	Definition
M	Match
N	No match.
P	Not Processed
S	Should be on card but was not provided. (Visa only)
U	Issuer not participating or certified.

Appendix 2 – Code Samples

Merchant Sample Implementation

Given the variety of possible implementations (e.g. SPA Web App, MVC Application, etc.) it's not possible in this document to capture every possible implementation. Below is a sample integration of the PowerTranz API into a merchant web application using a simplified MVC (Model, View, Controller) architecture using OpenAPI to generate a HTTP Client and Model.



1) Merchant Check Out View
   Merchant application gathers Card Holder data and posts data to Authentication View.
2) Authentication View with iFrame
   The Merchant application submits Auth Request to the Auth Endpoint and returns Auth Response to the Authentication View. This view will contain an Iframe to which the RedirectData will be bound.
   - PowerTranz End Point: {PowerTranz Root URL}/api/spi/auth
   - Request Body: Auth Request
   - The AuthRequest.ExtendedData.MerchantResponseUrl attribute must contain a URI in the Merchant Application domain to which the Iframe will Post the final Authentication Response.
   - Response: Auth Response containing IsoResponseCode and RedirectData – an HTML form that will execute within the context of the Iframe.
   - AuthResponse.RedirectData is injected or bound to the Iframe. For example:

```
<div class="text-center">
    <h4 class="display-4">IFrame</h4>
    <iframe id="threedsIframe" ref="threedsIframe" srcdoc="@Model.RedirectData">
    </iframe>
</div>
```

   -

3) The Iframe
   Once the RedirectData has been bound to the Iframe, the process will continue in the context of the Iframe.
   - The Card Holder may then be challenged (Challenge) to add further authentication at which point a form will appear in the iFrame and the Card Holder will enter additional information. Once the Card Holder enters the required information the Iframe context will post the Authentication result directly to the Merchant Response URL.
   - Alternatively, if no additional Card Holder input is required (Frictionless), the iFrame context will post the Authentication result directly to the Merchant Response URL.
   - In both examples (Challenge and Frictionless) the Authentication Result is posted to the Merchant Response URL.

4) Merchant Response URL and iFrame Removal
   - The Merchant Response URL is a page that exists within the Merchant Application's domain.
   - It is the iFrame context that will post the final Authentication result to this page and its lifespan is intended to be very short lived and transparent to the cardholder browser.
   - This page will contain JavaScript that will redirect the iFrame's parent container to the Authentication Result View effectively removing the Iframe and returning control to the Merchant Application. For example:

```
<script>

    window.onload = redirectParent;

    function redirectParent() {
        window.parent.location = './AuthenticationResult';
    }

</script>
```

5) Authentication Result View
   This view will process the final authentication result. If successful, the Merchant App will continue through to Payment Completion.
6) Payment Completion View
   The Merchant App can now call subsequent end points such as Payment, Capture and/or Void.