PROPAY® APPLICATION PROGRAMMING INTERFACE

Instruction to interface with ProPay®

Version 6.18.8
September 2020

PROPAY
A TSYS® Company

CONTENTS

1.0 ProPay® Application Programming Interface

ProPay is a robust payments network that utilizes ProPay merchant accounts to process major card brands, ACH payments, and supported alternative payment methods. A ProPay user account is not needed to make a purchase from a ProPay merchant using their credit card, ACH account information or supported alternative payment method type.

How to use this manual:

This manual is designed to enable developers to integrate the ProPay API into their own software solutions. The ProPay API is not written specific to a single platform but its users should have an understanding of basic principles.

- Developers should have an understanding of Hyper Text Transfer Protocol (HTTP) communication
- The consumption of external Web services and creation of a Secure Sockets Layer (SSL) connection from their own development platform.
- Certain methods require an understanding of SSL communication using X.509 certificates and Advanced Encryption Standard (AES) encryption on the developer's own platform.
- Extensible Markup Language

While ProPay offers resources and materials to assist developers in creating solutions and software, it is the responsibility of the developer to develop his or her own solution and software on the intended development platform to make use of and consume the services offered by ProPay. For additional resources please visit our new site: www.propay.com/developer.

Disclaimer

ProPay provides the following documentation on an "AS IS" basis without warranty. ProPay does not represent or warrant that ProPay's website or the API will operate securely or without interruption. ProPay further disclaims any representation or warranty as to the performance or any results that may be obtained through use of the API.

Regardless of its cause, ProPay will not be liable to client for any direct, indirect, special, incidental, or consequential damages or lost profits arising out of or in connection with client's use of this documentation, even if ProPay is advised of the possibility of such damages. Please be advised that this limitation applies whether the damage is caused by the system client uses to connect to the ProPay services or by the ProPay services themselves

Pay Card Industry (PCI) Data Security Standard (DSS) Compliance

All merchants must be compliant with the Payment Card Industry Data Security Standard (PCI DSS). For merchants that are integrating into the ProPay API, which includes the handling and transmission of card data directly, merchants are required to validate that they have completed the appropriate PCI DSS requirements.
Validating compliance may include an on on-site assessment (for very large merchants) by a Qualified Security Assessor (QSA), or the completion of a Self-Assessment Questionnaire for smaller merchants, as applicable to their processing volumes and circumstances. Depending upon the architecture of the cardholder data environment, compliance with the PCI DSS may also include quarterly vulnerability scans and a penetration test. It is the responsibility of the merchant to validate compliance with the PCI DSS and to provide validation documentation to ProPay. Failure to validate compliance may result in fines, fees, and penalties for non-compliance or data breaches.

Service Providers must also comply with the PCI DSS. A Service Provider is defined as any entity that stores, processes, or transmits cardholder data on behalf of a merchant or acquiring bank. Currently, service providers processing more than 300,000 transactions annually must undergo an on-site assessment by a QSA. Smaller Service Providers must validate compliance by completing the "SAQ – D Service Provider." Compliance may also include quarterly vulnerability scans and a penetration test. In addition to the requirement to validate compliance with the PCI DSS, Service

Providers have an additional obligation to register with the Card Brands. This allows the Card Brands additional insight into entities that may be storing, processing, or transmitting cardholder data. Registration involves some due diligence on the part of the acquiring bank and a listing on the **Visa Global Service Provider Registry** and the **MasterCard PCI-Compliant Service Provider List**. If a Service Provider has undergone the registration process with another acquirer, it must still register through ProPay, but needs only to provide its registration number, as opposed to undergoing the underwriting process again.

For current information about the defined Merchant and Service Provider processing levels and their corresponding PCI DSS requirements, please see www.pcisecuritystandards.org

Merchants and service providers may be able to limit the scope of their PCI Compliance requirements by using tokenization solutions, such as ProPay's ProtectPay solution to remove card data from traversing their environments. For more details on those options, please discuss with a ProPay relationship manager.

2.0 Interface Testing and Certification

Integrating a developed software solution to the ProPay web integration requires the following steps:

1. Request API credentials from a ProPay sales representative and/or account manager. Then integrate those methods specific for your project scope. A ProPay sales representative and/or account manager will help determine which methods are required for the specific project scope
2. Design, develop, build and test the software solution using the ProPay Integration Server. The ProPay Integration mobile API URI: https://mobileapitest.propay.com/merchant.svc
3. Request Production (Live) Credentials from a ProPay sales representative and/or account manager Live Credentials MUST be kept confidential The ProPay Production mobile API URI: https://mobileapi.propay.com/merchant.svc

All methods described in this documentation require the verb POST.

To improve the customer experience, ProPay requires that new developers test their software solutions before receiving credentials to process live transactions. This integration process is designed to assist the developer in building a robust solution that can handle and process all the various responses that come from real time credit card and ACH processing. This process ultimately improves the end-user experience. Please plan accordingly when developing timelines and schedules to accommodate for testing against the ProPay Integration environment. Negotiated fees are not refunded in the production environment.

Test account numbers for Credit Card and ACH processing are listed in the Appendix to this document

3.0 Technical Implementation

3.1 Secure Transmission

ProPay recognizes the importance of handling financial transactions in a secure manner, and ensures that our solutions offer high levels of transmission security. ProPay ensures that request information is transmitted using the latest Secure Sockets Layer (SSL) encryption practices. SSL creates a secure connection between client and server over which encrypted information is sent. ProPay hosts the SSL certificate for this connection type. Most method requests will negotiate an SSL connection automatically over port 443.

3.2 ProPay Mobile API Authentication

Most methods described by this document require a session key to authenticate. There are multiple ways to obtain this session key:

1. For ProPay account holders who use the ProPay website, and as a result, know their own username and password; you may use the method [Authenticate with username and password] described in this document.
2. Once the user has set up their pin using [Set Pin], you may use the method [Authenticate using PIN]

In no cases are you permitted to save and store the password or PIN in your mobile application. Merchants must enter this value to form any of the above requests.

If your users (merchants) never log in their ProPay account, and if you wish to use your own authentication processes to establish access to your mobile application, you can obtain a working key for the merchant.

1. Refer to the ProPay Merchant Services API (XML) to learn how to obtain a session key using your own credentials and client-side x509 certificate validation.

You may not embed the certificate or credentials into your mobile application. You must insist on authentication to your own online platform from which you may obtain a session key.

Session Keys last for 15 minutes. If a session key is used to perform one of the methods described in this document, its time to live is extended for an additional 15 minutes.

3.3 Best Practices

- When transitioning from the ProPay testing environments to the ProPay live servers, new service endpoints will be needed. These should be defined and referenced throughout the developed software solution as to only have to update a single reference. (Typically this is best done using a configuration file.)
- Payment processing over the internet can take up to 1 minute before an API response is received. Shorter system timeout values should not be configured.
- When building a solution, it is helpful to provide a basic means by which the business can locate the actual API request and response detail (with sensitive data redacted). Having this data available enables faster troubleshooting and issue resolution with the ProPay Technical Support team. Any such logs should include a UTC timestamp to a resolution no less than hh:mm:ss.
- Credit card transactions can take several seconds to process, due to the various parties involved in completing a transaction request. While ProPay has duplicate transaction prevention logic, it is recommended that developers take measures to discourage the clicking of a browser back button, or clicking 'submit' a second time to prevent duplicate transaction submission. ProPay also recommends that developers generate a visual control that indicates the transaction is processing during the waiting period.

4.0 Authentication Methods

4.1 Authenticate with Username and Password

By providing this method with a ProPay account holder's username and password, you can obtain a PIN key that will be used to log in. Feel free to save the PIN token on the user's device for future use in "logging in" (to obtain a working key with which you may make additional API requests.)
Note: While you are allowed to save the PIN token returned by this method on the user's device, the user must be prompted to provide their PIN each time he or she logs in. You may not save the PIN itself.

How to call this method

Resource URI
/json/AuthenticateUsernamePassword

Request Data - Required

Element	Nested Element	Type	Max	Required	Notes
Username		String	55	Required	Username of the merchant's ProPay Account
Password		String	19	Required	Password of the merchant's ProPay Account

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
ChangeKey		Guid	
PinToken		Guid	Value saved on device for future login along with PIN.

Sample JSON Request

```
{
 "Username": "test@email.com"
 "Password": "password"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "ChangeKey": "00000000-0000-0000-0000-000000000000",
   "PinToken": "c79b0cf8-c13c-40ea-8093-61940bb00571"
}
```

4.2 Login with PIN

Use this method to obtain a working "Session Key" with which you make additional calls. Please see section 3.2 for additional information about possible/allowed methods for establishing a login that consumes the ProPay mobile API.

This method requires a PIN token that may be obtained by using the method [Authenticate with Username and Password] While you are allowed to save the PIN token required for this method on the user's device, the user must be prompted to provide their PIN each time he or she logs in. You may not save the PIN itself.

How to call this method

Resource URI
/json/AuthenticateTokenPin

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
PinToken		Guid	36	Required	Returned by Authenticate Username and Password. May be stored on device.
Pin		String	4	Required	Customer's PIN. Must be

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AffiliateId		Integer	Additional useful information about merchant. Describes merchant as member of a specific partner program.
DefaultCountryCode		Integer	Additional useful information about merchant. USA=840, CAN=124
DefaultCurrencyCode		Integer	Additional useful information about merchant.
IdentityDetail		Container	
	GrantedRights (IdentityDetail)	Container	
	AccountId	String	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.
	RightName	String	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.
	SystemId	Integer	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.
	IdeneityId	String	

SessionToken		Guid	Primary response value for this method. Needed for most calls documented for this API.
TierId		Integer	Additional useful information about merchant.
TipRule		Container	
	CanAcceptTips	Boolean	Member of TipRule. Important if your solution needs to support tips at Capture.
	MaximumDifference	Integer	Member of TipRule. Important if your solution needs to support tips at Capture.
	MaximumDifferenceMultiplier	Decimal	Member of TipRule. Important if your solution needs to support tips at Capture.

Sample JSON Request


```
{
  "PinToken": "c79b0cf8-c13c-40ea-8093-61940bb00571",
  "Pin": "1111"
}
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AffiliateId": 15087,
   "DefaultCountryCode": 840,
   "DefaultCurrencyCode": 840,
   "IdentityDetail": {
      "GrantedRights": [
         {
            "AccountId": "8633026826591234",
            "RightName": "EDITPAYERS",
            "SystemId": 2
         },
         {
            "AccountId": "8633026826591234",
            "RightName": "PROCESSPAYMENTS",
            "SystemId": 2
         },
         {
            "AccountId": "1662952",
            "RightName": "USESPC",
            "SystemId": 3
         }
      ],
      "IdentityId": 9999999
   },
   "SessionToken": "df31b7fa-fffa-4d91-8ef4-53ad454f5d45",
   "TierId": 2119,
   "TipRule": {
      "CanAcceptTips": false,
      "MaximumDifference": 0,
      "MaximumDifferentialMultiplier": 0
   }
}
```

4.3 Set PIN

This method should be used the first time the user logs in to your mobile application. By providing both a username/password and a new PIN, you can establish first login and set up an easier-to-use PIN for subsequent visits.

This method returns both the PIN Token required to log in during subsequent visits, AND the Session Token so that you can proceed with the use of additional methods. Feel free to save the PIN Token on the user's device, but do not save the PIN itself.

How to call this method

Resource URI
/json/SetPin

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
Username		String	55	Required	Username of the merchant's ProPay Account
Password		String	19	Required	Password of the merchant's ProPay Account
Pin		String	4	Required	New PIN

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AffiliateId		Integer	Additional useful information about merchant. Describes merchant as member of a specific partner program.
CurrencyCode		Integer	Additional useful information about merchant.
IdentityDetail		Container	
	AccountId	String	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.
	SystemId	Integer	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.
	IdentityId	Integer	
PinToken		Guid	Primary response value for this method. Can be saved on user's device for future logins.
SessionToken		Guid	Primary response value for this method. Needed for most calls documented for this API.
TierId		Integer	Additional useful information about merchant.
RightName		String	Member of GrantedRights, which defines a list of available services on specific ProPay platform and for an account.

TipRule		Container	
	CanAcceptTips	Boolean	Member of TipRule. Important if your solution needs to support tips at Capture.
	MaximumDifference	Integer	Member of TipRule. Important if your solution needs to support tips at Capture.
	MaximumDifference Multiplier	Decimal	Member of TipRule. Important if your solution needs to support tips at Capture.

Sample JSON Request


```
{
 "Username": "c36c789c561d@propaytest.com",
 "Password": "AB897897",
 "Pin": "1234"
}
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": "Success"
   },
   "AffiliateId": 1027406,
   "CurrencyCode": 840,
   "IdentityDetail": {
      "GrantedRights": [
         {
            "AccountId": "1959916873458777",
            "RightName": "EDITPAYERS",
            "SystemId": 2
         },
         {
            "AccountId": "7979233331737920",
            "RightName": "EDITPAYERS",
            "SystemId": 2
         },
      ],
      "IdentityId": 4623773
   },
   "PinToken": "fda368a5-3e6a-47f9-a343-42a5b11c17d6",
   "SessionToken": "a2cd00e7-0941-4bcd-a39c-8ec58cd99ad0",
   "TierId": 1048075,
   "TipRule": {
      "CanAcceptTips": false,
      "MaximumDifference": 0,
      "MaximumDifferentialMultiplier": 0
   }
}
```

4.4 Logout

Use this method to manually expire a session token currently in use. Note: if you do not perform this action, session tokens expire after fifteen minutes unless they are used. (The act of using a session token extends its duration by an additional fifteen minutes.)

How to call this method

Resource URI
/json/Logout

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "6cd9203c-25d6-42e6-8c69-dbcdf3f1bef5"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

4.5 Get Challenge Questions

This method supports the first step needed to create a 'Forgot Password' process, or to register a device. (Which is also required for 'Forgot Password' and should be performed during initial login to your mobile application.) This method will return the text of challenge questions originally set up by the customer along with a unique id that, in a subsequent step, must be passed along with the question's answer.

How to call this method

Resource URI
/json/GetChallengeQuestions

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
Username		Int(32)	55	Required	The account whose password is to be reset

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
ChallengeQuestions		Container	
	Id	Integer	Identifies the question when passed along with answer using method [ForgotPassword]
	Question	String	Text of the question as displayed to merchant.

Sample JSON Request

```
{
 "Username": "testuser@propay.com"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "ChallengeQuestions": [
      {
         "Id": 829102,
         "Question": "1. What is your father's middle name?"
      },
      {
         "Id": 829103,
         "Question": "2. What is the first name of a favorite grandparent?"
      },
      {
         "Id": 829104,
         "Question": "3. What is the name of your favorite place?"
```

```
        }
      ]
    }
```

4.6 Register Device

Device Registration is another prerequisite for building a 'Forgot Password' solution because ProPay will only allow users to perform those actions on a registered device. You should consider building device registration into the first login event. (After the user establishes his or her password, PIN, and challenge questions.)

How to call this method

Resource URI
/json/RegisterDevice

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
Answers		Container			
	Answer	String		Required	User must answer challenge questions for device to be registered or for user to initiate Forgot Password steps.
	Id	Integer	10	Required	Identifies question being answered.
Username		String	55	Required	Username of the merchant's ProPay Account

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
DeviceRegistrationId		String	Used in Forgot Password. Should be saved on device.

Sample JSON Request

```
{
 "Answers": [{
  "Answer": "car",
  "Id": 4479389
 }],
 "Username": "e6da339d851640079@propaytest.com"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "DeviceRegistrationId": "bb5de8ebc63b4763ad4b726409a5b137"
}
```

4.7 Forgot Password

This is the second step of a 'Forgot Password' process and requires that you provide the answer to a challenge question. ProPay, after validating the answer to a question against that questions unique id, will send out an email to the merchant with insctructions for changing his or her password.

How to call this method

Resource URI
/json/ForgotPassword

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
Answers		Container			
	Answer	String			Answer to challenge question
	Id	Integer	10		Identifier for question answered
DeviceRegistrationId		String			Returned when device is registered. Should be stored on device.
Username		String	55		

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "Answers": [{
  "Answer": "car",
  "Id": 4479387
 }],
 "DeviceRegistrationId": "96fcc4b85b0843de821823650e462c80",
 "Username": "653b7f2a827c4ed29@propaytest.com"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

4.8 Change Password

This method will upate the login password for a user.

How to call this method

Resource URI
/json/ChangePassword

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
Username		String	55	Required	Username of the merchant's ProPay Account
CurrentPassword		String	19	Required	Password of the merchant's ProPay Account
NewPassword		String	19	Required	New Password to set for merchant's ProPay account

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "CurrentPassword": "rX63R!Fg3y",
 "NewPassword": "password123",
 "Username": "newpassword123"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

4.9 Set Challenge Questions

This question will allow you to set up challenge questions for a user. Challenge questions will be needed if a user forgets his or her login password.

How to call this method

Resource URI
/json/SetChallengeQuestions

Account Identification - Required

Element	Neted Element	Type	Max	Required	Notes
ChallengeQuestions		Container			
	Answer	String		Required	
	Question	String		Required	
Username		String	55	Required	It is required to 'Log In' when setting up challenge questions.
Password		String	19	Required	It is required to 'Log In' when setting up challenge questions.

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "ChallengeQuestions": [{
  "Answer": "answer 1",
  "Question": "Question 1"
 }, {
  "Answer": "answer 2",
  "Question": "Question 2"
 }, {
  "Answer": "answer 3",
  "Question": "Question 3"
 }],
 "Password": "A444444456",
 "Username": "06b34f9d81dd40cb8a31d9145a92a2f2"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

4.10 Issue Multi Factor Authentication Token

Issue a Multi Factor Authentication token

How to call this method

Resource URI
/json/CreateMultiFactorAuthToken

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
IgnorePreviousTokenFailures		Boolean		No	Token attempts earlier needs to be ignored and fresh token issued. This value should be set to true only if mobile user has failed token validation multiple attempts and user authentication has been reset.

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
TokenRequestId		Guid	The Request ID for the token. Required for validating the token.
EmailRecipient		String	Email where token was sent
IsFinalAttempt		Boolean	Next Attempt to validate the token will be the final , before account will be locked

Sample JSON Request

```
{
"SessionGuid":"7a26734b-3aa9-465e-967c-43cc32c471cb",
"IgnorePreviousTokenFailures":false
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
"EmailRecipient": "3d26f7633fulEqigcJx@propaytest.com",
  "IsFinalAttempt": false,
  "TokenRequestId": " 39EE956B-0098-4BD9-AFF2-17E9156C15DC"
}
```

4.11 Validate Multi Factor Authentication Token

Validate a Multi Factor Authentication token

How to call this method

Resource URI
/json/ValidateMultiFactorAuthToken

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
Token		String	6	Required	OTP Token received by email
TokenRequestId		Guid		Required	The Request ID for the token. This will be returned while issuing the token.

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
IsFinalAttempt		Boolean	Next Attempt to validate the token will be the final , before account will be locked

Sample JSON Request

```
{
"SessionGuid":"7a26734b-3aa9-465e-967c-43cc32c471cb",
"Token":111111
"TokenRequestId: 7a26734b-3aa9-465e-967c-43cc32c1234S"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
  "IsFinalAttempt": false,
}
```

5.0 Account Management Methods

5.1 Get Account Summary Details

This method can be used to create a landing page where multiple useful pieces of information about an account can be displayed.

How to call this method

Resource URI
/json/GetAccountSummary

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AccountExpiration		String	
AccountNumber		Integer	
AccountBalance		Integer	Whole number representing number of pennies.
LimitProcessed		Integer	Monthly limit
LimitRemaining		Integer	Monthly limit
TransactionsInProcess		Integer	

Sample JSON Request

```
{
 "SessionGuid": "d2e84548-42ea-42a1-8d36-7b08a8905a75"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AccountExpiration": "/Date(1577430000000-0700)/",
   "AccountNumber": 30995876,
   "AvailableBalance": 151520.04,
   "LimitProcessed": 75462.35,
   "LimitRemaining": 99924536.65,
```

"TransactionsInProcess": 1
}

5.2 Get Recent Transactions List

This method can be used to display a list of recent transactions. Be advised: this solution was designed to support merchants with relatively low transaction volumes. If you attempt to build a reporting interface using this feature, larger merchants may experience timeouts. Partners who wish to build a solution for active merchants are advised to track their own transactions based on API responses returned by ProPay during processing.

How to call this method

Resource URI
/json/GetRecentTransactions

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
Count		Integer	10	Required	
Start		Integer	10	Required	

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
ContainsLastTransaction		Boolean	Tells caller that returned list contains the last of the merchant's transactions. (Don't need to make another call to get the rest of the list.)
Transactions		Container	
	AccountNumber	Integer	Member of Transactions.
	AuthorizationCode	String	Member of Transactions.
	AvsCode	String	Member of Transactions.
	CurrencyCode	Integer	Member of Transactions.

	Date	String	Member of Transactions.
	Email	String	Member of Transactions.
	Fee	Integer	Member of Transactions.
	GrossAmount	Integer	Member of Transactions.
	InvoiceNumber	String	Member of Transactions.
	Name	String	Member of Transactions.
	NetAmount	Integer	Member of Transactions.
	Number	Integer	Member of Transactions.
	Status	Integer	Member of Transactions. 0=Unknown, 1=Pending, 2=Complated
	TipAmount	Integer	Member of Transactions.
	Type	String	Member of Transactions.

Sample JSON Request


```
{
 "SessionGuid": "6cd9203c-25d6-42e6-8c69-dbcdf3f1bef5",
 "Count": 1,
 "Start": 1
}
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "ContainsLastTransaction": false,
   "Transactions": [
      {
         "AccountNumber": "5100",
         "AuthorizationCode": "A11111",
         "AvsCode": "T",
         "CurrencyCode": 0,
         "Date": "/Date(1550784611883-0700)/",
         "Email": "",
         "Fee": 0,
         "GrossAmount": 100,
         "InvoiceNumber": "",
         "Name": "Manual Entry",
         "NetAmount": 100,
         "Number": 16324,
         "Status": 2,
         "TipAmount": 0,
         "Type": "+CC"
      }
   ]
}
```

5.3 Register for Push Notifications

This method can be used to register a device for various push notifications sent by the ProPay system. (Things like notice that a transaction has occurred, or that money has left an account by way of a bank transfer.)

How to call this method

Resource URI
/json/RegisterForPush

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
DeviceType		Integer	1	Required	1=iOS, 2=Android
NotificationToken		String		Required	
PreferredLanguage		String		Optional	

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
RegistrationId		Integer	Save on device in case user wants to un-register

Sample JSON Request

```
{
 "SessionGuid": "f8f734d4-8c15-4243-a0f6-99585c1a7d14",
  "DeviceType": 1,
  "NotificationToken": "e6da339d851640079",
 "PreferredLanguage": null
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 307,
      "ResultMessage": ""
   },
   "RegistrationId": "f8c2512ab2-4d9c9b8e3ae88927ec05sdfae45sd6f"
}
```

5.4 Unregister for Push Notifications

This method causes that a device will no longer receive push notifications.

How to call this method

Resource URI
/json/UnRegisterForPush

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
RegistrationId		Integer		Required	

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "53dfe76f-24fc-46e6-90f7-563306f927a9",
 "RegistrationId": "53dfe76f563306f927a9"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

6.0 Access Funds Methods

6.1 Get Account Data for Bank Transfer

This method is used to obtain information with which you can create a useful and user friendly bank transfer solution. (So that a merchant can send the proceeds of his or her transactions to an on-file bank account.) It provides details that will help you explain, to the merchant, how much money he or she can transfer, and how much of a fee he or she will be expected to pay for the transfer.

How to call this method

Resource URI
/json/GetAccountDataForTransfer

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AccountNumber		String	
AvailableToTransfer		Integer	
CanAchOut		Boolean	
CurrentAccountBalance		Integer	
MinimumTransferAmount		Integer	
TransferFee		Integer	
TransferFeeAdjusted		Boolean	

Sample JSON Request

```
{
 "SessionGuid": "d2e84548-42ea-42a1-8d36-7b08a8905a75"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AccountNumber": "6789",
   "AvailableToTransfer": 15152,
   "CanAchOut": true,
```

```
    "CurrentAccountBalance": 15152,
    "MinimumTransferAmount": 1,
    "TransferFee": 0,
    "TransferFeeAdjusted": true
}
```

6.2 Transfer Funds to Bank

This method will cause money to be sent from a merchant account to the owner's on-file bank account. Use after getting details from [Get Account Data for Bank Transfer] so that you can create a good user experience.

How to call this method

Resource URI
/json/TransferFundsToBank

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
Amount		Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals to transfer

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "458f21ee-4d2e-4d22-a261-acf539871534",
 "Amount": 100
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   }
}
```

6.3 Get Merchant Transfer Option Details

This method is used to obtain merchant flash funds details which you can create a useful and user-friendly flash funds bank transfer option. It provides details that will help you to explain, to the merchant, how much money they can transfer and how much of a fee they will be expected to pay to perform the transfer. The result will provide additional information to understand flash fund availability and transfer details.

How to call this method

Resource URI
json/GetMerchantTransferOptionDetails

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login

Response Elements

Element	Nested Element	Type	Notes
Result			
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode.
FlashFundsAccountData	AccountNumber	String	
	CardholderName	String	
	ExpirationDate	String	
	AvailableToTransfer	Integer	
	FlashFundsTransferFee	Decimal	
	FlashFundsEnabled	Boolean	
	CurrentAccountBalance	Integer	
	MinimumTransferAmount	Integer	
	RemainingTransferLimit	Integer	

Sample JSON Request

```
{
"SessionGuid": "d2e84548-42ea-42a1-8d36-7b08a8905a75"
}
```

Response

```
{
 "Result":{
  "ArgumentId": 0,
  "ResultCode": 0,
  "ResultMessage": ""
  },
```

```
 "FlashFundsAccountData":{
  "AccountNumber": "0006",
  "AvailableToTransfer": 150,
  "CardholderName": "Clint Eastwood",
  "CurrentAccountBalance": 25000,
  "ExpirationDate": "2020-10",
  "FlashFundsEnabled": true,
  "FlashFundsTransferFee": 2.00,
  "MinimumTransferAmount": 50,
  "RemainingTransferLimit": 1000000
  }
}
```

6.4 TransferFundsToFlashFundsCard

This method is used to transfer funds to flash funds card. Account holder can transfer funds to flash fund card from their bank balance. This method will return success response if amount was transferred to flash fund card. If amount is not successfully transferred to flash fund card, the reason will be in the response with result code & result message.

Resource URI
/json/TransferFundsToFlashFundsCard

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
Amount		Long		Required	The value representing the number of pennies in USD, or the number of [currency] without decimals to transfer

Element	Nested Element	Type	Notes
Result			
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode.

Sample JSON Request


```
{
"SessionGuid": "d2e84548-42ea-42a1-8d36-7b08a8905a75",
"Amount": "1245",
}
```


Response


```
{
 "Result":{
  "ArgumentId": 0,
  "ResultCode": 0,
  "ResultMessage": "Success"
  },
}
```

7.0 Managing Stored Cards

7.1 Fetch Stored Cards

This method will return details of cards saved for the merchant's ProtectPay collection of on-file payment methods. Note: this method is suitable only for ProtectPay 'Lite' customers whose number of on-file cards is limited to a few dozen. An 'Enterprise' ProtectPay biller making this request will likely encounter a timeout.

How to call this method

Resource URI
/json/FetchStoredCards

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
Stored Cards		Container	
	Description	String	
	Email	String	
	ExpirationDate	String	
	Name	String	
	ObfuscatedCardNumber	String	
	PayerId	Integer	
	PaymentMethodId	String	
	PaymentType	String	0=Bank account ACH transfers 1=CreditCard transaction 2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)
	PostalCode	String	

Sample JSON Request

Response

```
{
 "SessionGuid": "d2e84548-42ea-42a1-8d36-7b08a8905a75"
}
```



```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": "Success"
   },
   "StoredCards": [
      {
         "Description": "SPC",
         "Email": null,
         "ExpirationDate": "1230",
         "Name": "DEMO",
         "ObfuscatedCardNumber": "222123******2222",
         "PayerId": 2519603953371686,
         "PaymentMethodId": "e93e4de8-bfa6-49a9-a4b8-68f187862aba",
         "PaymentType": "CreditCard",
         "PostalCode": null
      },
      {
         "Description": "",
         "Email": null,
         "ExpirationDate": "0919",
         "Name": "ProtectPay Test",
         "ObfuscatedCardNumber": "411111******1111",
         "PayerId": 1869554709860866,
         "PaymentMethodId": "922f82a8-357c-4094-9595-80d118cafe52",
         "PaymentType": "CreditCard",
         "PostalCode": null
      }
   ]
}
```

7.2 Edit Stored Card

This method will allow you to edit the details of an existing stored card.

How to call this method

Resource URI
/json/EditStoredCard

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
Email		String	100	Optional	The email address for the payment method
ExpirationDate		String	4	Required	The expiration date for a payment method. For a credit card these are submitted as 4 digit numeric values MMYY
Name		String	50	Optional	The name on the account for a payment method
PayerId		Integer	19	Required	Identifies payment method that must be changed.
PaymentMethodId		Integer	36	Required	Identifies payment method that must be changed.
PostalCode		String	10*	Optional	The zipcode on the account for a payment method. *9 digits for US cardholders

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "a6430946-8fd3-4326-8017-28591a2801b7",
 "PayerId": "4721234391089922",
 "PaymentMethodId": "a020498d-55ab-4fc0-8cff-725c9e07e79e",
 "Name":  "abc123",
 "ExpirationDate": "0222",
 "PostalCode": "",
 "Email": "abc@test.com"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": null
   }
}
```

7.3 Delete Stored Card

This method will remove a stored card from a user's on-file collection.

How to call this method

Resource URI
/json/DeleteStoredCard

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
PayerId		Integer	16	Required	Both payer and payment method must be specified to ensure the correct payment method is deleted
PaymentMethodId		Integer	36	Required	Valid value is a GUID. Both payer and payment method must be specified to ensure the correct payment method is deleted

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Interger	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "a6430946-8fd3-4326-8017-28591a2801b7",
 "PayerId": "4721234391089922",
 "PaymentMethodId": "a020498d-55ab-4fc0-8cff-725c9e07e79e"
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": null
   }
}
```

7.4 Process and store a Swiped Card

This method will process a card and then store it for future use. Note: the only way to store cards using the Mobile API is after a transaction has been successfully processed. The mobile solution is not for use by 'Enterprise' ProtectPay customers. This version of Process and store works only with a ProPay approved/encrypted swipe device.

How to call this method

Resource URI
/json/ProcessAndStoreCardSwipe

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
BillingData		Container		Container	
	CountryCode	Integer	3	Required	USA=840, CAN=124
	Email	String	100	Optional	If sent, will cause an email receipt to be sent to cardholder.
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
FullTransactionData				Container	
	SignatureBlock	Byte[]	3	Optional	ProPay will save this value if provided and can use it to help represent a chargeback if one occurs.
ProcessCardData				Container	
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	TipAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tip amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true

SwipeCardData				Container	
	Name	String		Optional	Cardholder
EncryptedTrackData				Container	
	DeviceType	Integer	2	Required	Must be ProPay encrypted and approved swipe device. The following are ProPay specific reference names. 1=Magtek 20 2=Magtek Flash 3=IdTech Uni Mag 4=Manual card entry 5=Magtek ADynamo 6=Magtek Dynamag 7=Roam Data 8=Roam Data EMV 9=(Partner specific value) 10=Anywhere Commerce EMV 11=(Partner specific value) 12=(Partner specific value) 13=(Partner specific value) 14=Moby3000
	EncryptedTrack2Data	Byte[]		Required	
	EncryptedTrackData	Byte[]		Required	
	KeySerialNumber	Byte[]		Required	

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentID	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	0=Bank account ACH transfers 1=CreditCard transaction 2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)

Sample JSON Request


```
{
 "SessionGuid": "a6c155ce-3f26-48e4-90d7-b8ad541889a5",
 "BillingData": {
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
```

```
 "CountryCode": 0,
 "Email": "receipt@propay.com",
 "PostalCode": "84043"
},
"FullTransactionData": {
 "SignatureBlock": [42]
},
"ProcessCardData": {
 "Amount": 300,
 "Comment1": null,
 "Comment2": null,
 "CurrencyCode": 840,
 "InvoiceNumber": "cd0c6760",
"IsQuasiCash":false,
 "MerchantProfileId": 0,
 "TaxAmount": 30,
 "TipAmount": 60
},
"SwipeCardData": {
 "EncryptedTrackData": {
  "DeviceType": 7,
  "EncryptedTrack2Data": [82, 248, 50, 221, 242, 186, 240, 102, 161, 102, 192, 180, 64
  "EncryptedTrackData": [197, 245, 222, 249, 11, 253, 219, 255, 8, 186, 9, 152, 221, 17
  "KeySerialNumber": [144, 16, 36, 11, 0, 96, 29, 0, 0, 92]
 },
 "Name": "RoamData Payer"
}
}
```



```
    "ResultCode": 0,
    "ResultMessage": ""
  },
  "AuthorizationCode": "A11111",
  "PayerId": 4023675788442595,
  "PaymentMethodId": "62fb88f4-3da8-4639-9c6c-86b5c1652c4e",
  "PaymentType": 1
}
```

7.5 Process and Store a Manually Entered Card

This method will process a card and then store it for future use. Note: the only way to store cards using the Mobile API is after a transaction has been successfully processed. The mobile solution is not for use by 'Enterprise' ProtectPay customers.

How to call this method

Resource URI
/json/ProcessAndStoreManualCard

Account Identification - Required

Element	Nested Elemet	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
BillingData		Container			
	CountryCode	Integer	3	Required	USD=840, CAD=124
	Email	String	100	Optional	If sent, will cause an email receipt to be sent to cardholder.
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
FullTransactionData		Container			
	SignatureBlock	Byte[]	3	Optional	ProPay will save this value if provided and can use it to help represent a chargeback if one occurs.
ManualCardData		Container			
	CVV	String	4	Optional	CVV code
	CardNumber	String		Required	Standard cc numbers which must pass Luhn check
	ExpirationDate	String	4	Required	The expiration date in MMYY format
	Name	String	100	Optional	Cardholder Name
ProcessCardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	TipAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tip amount
	IsQuasiCash	Boolean	5	Optional	Valid values are:

					• true • false If the transaction is IsQuasiCash then this needs to be set to true

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentID	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	0=Bank account ACH transfers 1=CreditCard transaction 2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)

Sample JSON Request

```
{
 "SessionGuid": "8f173ab3-5a76-44ad-906b-ce821e546bab",
 "BillingData": {
  "CountryCode": 0,
  "Email": "receipt@propay.com",
  "PostalCode": "84043"
 },
 "FullTransactionData": {
  "SignatureBlock": [42]
 },
 "ManualCardData": {
  "CVV": "999",
  "CardNumber": "4111111111111111",
  "ExpirationDate": "0220",
  "Name": "My payer"
 },
 "ProcessCardData": {
  "Amount": 300,
  "Comment1": null,
  "Comment2": null,
  "CurrencyCode": 840,
  "InvoiceNumber": "3e424096",
 "IsQuasiCash":false,
  "MerchantProfileId": 0,
  "TaxAmount": 30,
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111",
   "PayerId": 4398227998040796,
   "PaymentMethodId": "caf6bc7b-8ab5-4d72-9dea-31001a1b6996",
   "PaymentType": 1
}
```

```
  "TipAmount": 60
 }
}
```

7.6 Process With Stored Card

This method will allow you to process a transaction with a previously-saved card.

How to call this method

Resource URI
/json/ProcessStoredCard

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String		Required	Primary authentication mechanism after initial login
BillingData		Container			
	CountryCode	Integer	3	Required	USD=840, CAD=124
	Email	String	100	Optional	If sent, will cause an email receipt to be sent to cardholder.
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
PayerId		Integer	16	Required	Payer Id the payment method was stored under
PaymentMethodId		Guid	36	Required	The payment method (token) you wish to process the transaction against
ProcessCardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer		Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	TipAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tip amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentID	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AuthorizationCode		String	Returned authorization code for the transaction

Sample JSON Request


```
{
 "SessionGuid": "b0b07a16-ccf9-4f14-96bd-e8c11f73670d",
 "BillingData": {
  "CountryCode": 0,
  "Email": "receipt@propay.com",
  "PostalCode": "84043"
 },
 "PayerId": 2076231298852445,
 "PaymentMethodId": "864c1011-51fb-428f-95cf-6ce1fc08fc03",
 "ProcessCardData": {
  "Amount": 300,
  "Comment1": null,
  "Comment2": null,
  "CurrencyCode": 840,
  "InvoiceNumber": "89d5893c129f",
 "IsQuasiCash":false,
  "MerchantProfileId": 0,
  "TaxAmount": 30,
  "TipAmount": 60
 }
}
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111"
}
```

8.0 Credit Card Processing Methods

8.1 Authorize a Swiped Card

Pre-Auth of swiped card.

How to call this method

Resource URI
/json/AuthorizeCardSwipe

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
ApplicationId		String	4	Optional	May be required if device supports EMV but transaction is swiped.
BillingData		Container			
	CountryCode	Integer	3	Required	USD=840, CAD=124
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
Card Data		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true
CardDataSource		Integer		Required	0=MagStripe (This value should be used with this method) 1=Keyed transaction with no reader 2=Keyed transaction with reader present 3=EMV Contact 4=EMV Contactless 5=EMV Fallback

ExternalTransactionIdentifier		String		Optional	
PointOfSaleData		Container			
	CardDataInputMode	Integer		Required	49=Input manually without use of terminal 50=MagStripe (This value should almost always be used for this API method) 51=barcode or payment code 54=Input manually on terminal 55=Merhant initiated transaction using on-file payment credential 65=Input using mag stripe including track 2 data 67=Chip: Online Mode 70=Chip: Offline Mode 77=Chip: Contactless 78=EMV fallback to mag stripe 80=EMV fallback to mag stripe. Card does not support applications that terminal can peform 82=Ecommerce 83=Ecommerce with no security channel encryption 86=manually input with keyed CID 87=mag stripe with keyed CID 90=Discover specific contactless interface
	CardDataOutputCapability	Integer		Required	49=Card has no output capability (most likely value for this method) 50=Card has mag stripe write output capability 51=Card has integrated chip output capability 83=Card has other output capability
	CardPresentData	Integer		Required	48=Card not present 49=Card present (most likely value for this method) 87=Transponder 88=Contactless chip transacition 90=Digital wallet
	CardholderAuthenticationEntity	Integer		Required	48=Cardholder not authenticated by any entity 49=Cardholder authenticated by integrated chip ard offline pin 50=Cardholder authenticated by card acceptance device 51=Cardholder authenticated by authorizing agent using online pin 52=Cardholder authenticated by merchant using signature 53=Other cardholder authenticating entity present
	CardholderAuthenticationMethod	Integer		Required	48=Cardholder not authenticated 49=PIN 50=electrongic signature analysis 53=manual signature verification 54=other manual verification 83=other systematic verification 84=using electronic ticket environment
	CardholderPresentData	Integer		Required	48=Cardholder present 49=Cardholder not present; unspecified reason 50=Cardholder not present; mail transaction 51=Cardholder not present; phone transaction

					52=Cardholder not present; recurring transaction such as subscription 53=Cardholder not present; eCommerce 56=Cardholder not present; recurring billing such as loan or installment payment 65=Reauthorization for full amount 80=Partial shipment purchase 82=Recurring purchase transaction 84=Pay Button
	PinCaptureCapability	Integer		Required	49=No PIN capture capability 52=PIN capture capability; 4 characters max 53=PIN capture capability; 5 characters max 54=PIN capture capability; 6 characters max 55=PIN capture capability; 7 characters max 56=PIN capture capability; 8 characters max 57=PIN capture capability; 9 characters max 65=PIN capture capability; 10 characters max 66=PIN capture capability; 11 characters max 67=PIN capture capability; 12 characters max
	TerminalCardCaptureCapability	Integer		Required	48=No capture capability 49=Card capture capable
	TerminalCardDataInputCapability	Integer		Required	49=manual; no terminal; voice auth/aru only 50=mag stripe reader capability only (most common value for this method) 51=bar code payment code capable 52=OCR capable 53=Integrated circuit card capable 54=key entry only capable 65=PAN auto entry via contactless mag stripe 66=mag stripe reader, and key entry capable 67=mag stripe, icc, and key entry capable 68=mag stripe and icc capable 69=icc and key entry capable 72=icc reader and contactless capable; mag stripe and manual entry implied 77=pan auto entry via contactless chip
	TerminalCardholderAuthenticationCapability	Integer		Required	48=no electronic authentication capability 49=PIN entry capable 50=electronic signature capable 53=Electronic authentication capability is inoperative
	TerminalDataOutputCapability	Integer		Required	49=No terminal data output capability 50=terminal is capable of output printing only 51=terminal is capable of output display only 52=terminal is capable of output printing and display
	TerminalOperatingEnvironment	Integer		Required	48=no terminal used; voice auth/aru only 49=on card acceptor premises; attended terminal 50=on card acceptor premises; unattended terminal 51=off card acceptor premises; attended

					52=off card acceptor premises; unattended 53=on cardholder premises; unattended 77=off card acceptor premises; merchant mobile POS environment including mpos 80=on card acceptor premises; merchant mobile POS environment including mpos 81=off card acceptor premises; cardholder mobile environment incl home pc, phone, pda 82=on card acceptor premises; cardholder mobile environment incl home pc, phone, pda 83=electronic delivery of product 84=physical delivery of product
ShouldStoreCardData		Boolean		Optional	
SwipeCardData		Container			
	Name	String		Optional	Cardholder
	EncryptedTrackData (SwipeCardData)	Container			
	DeviceType	Integer	2	Required	Must be ProPay encrypted and approved swipe device. The following are ProPay specific reference names. 1=Magtek 20 2=Magtek Flash 3=IdTech Uni Mag 4=Manual card entry 5=Magtek ADynamo 6=Magtek Dynamag 7=Roam Data 8=Roam Data EMV 9=(Partner specific value) 10=Anywhere Commerce EMV 11=(Partner specific value) 12=(Partner specific value) 13=(Partner specific value) 14=Moby3000
	EncryptedTrack2Data	Byte[]		Optional	
	EncryptedTrackData	Byte[]		Required	
	KeySerialNumber	Byte[]		Required	

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentID	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	0=Bank account ACH transfers 1=CreditCard transaction 2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)
AttemptNumber		Integer	Merchant recognized ProPay transaction identifier
InvoiceNumber			Invoice Number
TransactionInfo		Container	
	GrossAmt	Integer	Gross amount of transaction of pennies in USD, or the number of [currency] without decimals.
	GrossAmtLessNetAmt	Integer	Total amount of fees charged; *ProPay Gateway Only
	NetAmt	Integer	Net amount of transaction after fees charged; *ProPay Gateway Only
	PerTransFee	Integer	Per transaction fee; *ProPay Gateway Only
	Rate	Decimal	Percentage fee; *ProPay Gateway Only
	CreationDate	String	yyyy-mm-dd hh:mm:ss:ms AM
	TransactionHistoryId	Integer	Unique ProPay identifier for future reference/handling of transaction
	ObfuscatedAccountNumber	String	Obfuscated Account Number for the payment method
	AccountName	String	Cardholder repeated or taken from EMV or swipe information.

Sample JSON Request

```
{
 "SessionGuid": "0842a2fe-c9d1-417a-8612-dd33841eac4f",
 "ApplicationId": "B350",
 "BillingData": {
  "CountryCode": 840,
  "PostalCode": "85284"
 },
 "CardData": {
  "Amount": 52,
  "Comment1": "Auth Comment1",
  "Comment2": "Auth Comment2",
  "CurrencyCode": 840,
  "InvoiceNumber": "VisaSwipe",
  "IsQuasiCash":false,
  "MerchantProfileId": 0,
  "TaxAmount": 0
 },
 "CardDataSource": 0,
 "ExternalTransactionIdentifier": "AA6345",
 "PointOfSaleData": {
  "CardDataInputMode": 66,
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111",
   "PayerId": 3872325921204538,
   "PaymentMethodId": "b0ad4fba-63a1-4840-93a8-2e3942d624640",
   "PaymentType": null,
   "AttemptNumber": 16274,
   "InvoiceNumber": "VisaSwipe",
   "TransactionInfo": {
      "GrossAmt": 52,
      "GrossAmtLessNetAmt": 52,
      "NetAmt": 0,
      "PerTransFee": 0,
      "Rate": 0,
      "CreationDate": "2019-04-17 17:15:25.542 AM",
      "TransactionHistoryId":12345,
      "ObfuscatedAccountNumber": "123456******9876",
```

```
  "CardDataOutputCapability": 51,
  "CardPresentData": 49,
  "CardholderAuthenticationEntity": 52,
  "CardholderAuthenticationMethod": 53,
  "CardholderPresentData": 48,
  "PinCaptureCapability": 48,
  "TerminalCardCaptureCapability": 48,
  "TerminalCardDataInputCapability": 67,
  "TerminalCardholderAuthenticationCapability": 54,
  "TerminalDataOutputCapability": 51,
  "TerminalOperatingEnvironment": 77
 },
 "ShouldStoreCardData": true,
 "SwipeCardData": {
  "EncryptedTrackData": {
   "DeviceType": 12,
   "EncryptedTrack2Data": [59, 52, 48, 49, 50, 48, 48, 0, 0, 0, 0, 0, 0, 53, 52, 51, 57, 61, 50, 48],
   "EncryptedTrackData": [147, 2, 221, 239, 91, 240, 141, 19, 115, 220, 168, 154, 101, 40, 38],
   "KeySerialNumber": [255, 255, 255, 0, 6, 0, 0, 0, 0, 133]
  },
  "Name": "Visa Swipe"
 }
}
```



```
    "AccountName": "Test cardholder name"
  }
}
```

8.2 Authorize with EMV Data

Pre-Auth of EMV/Chip card.

How to call this method

Resource URI
/json/AuthorizeEmv

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
ApplicationId		String	4	Optional	May be required if device supports EMV but transaction is swiped.
BillingData		Container			
	CountryCode	Integer	3	Required	USD=840, CAD=124
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
CardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true
CardDataSource		Integer		Required	0=MagStripe 1=Keyed transaction with no reader 2=Keyed transaction with reader present 3=EMV Contact (commonly used for this method) 4=EMV Contactless 5=EMV Fallback
ExternalTransactionIdentifier		String		Optional	
PointOfSaleData		Container			
	CardDataInputMode	Integer		Required	49=Input manually without use of terminal

					50=MagStripe 51=barcode or payment code 54=Input manually on terminal 55=Merhant initiated transaction using on-file payment credential 65=Input using mag stripe including track 2 data 67=Chip: Online Mode (commonly used for this method) 70=Chip: Offline Mode 77=Chip: Contactless 78=EMV fallback to mag stripe 80=EMV fallback to mag stripe. Card does not support applications that terminal can peform 82=Ecommerce 83=Ecommerce with no security channel encryption 86=manually input with keyed CID 87=mag stripe with keyed CID 90=Discover specific contactless interface
	CardDataOutputCapability	Integer		Required	49=Card has no output capability (most likely value for this method) 50=Card has mag stripe write output capability 51=Card has integrated chip output capability 83=Card has other output capability
	CardPresentData	Integer		Required	48=Card not present 49=Card present (most likely value for this method) 87=Transponder 88=Contactless chip transacition 90=Digital wallet
	CardholderAuthenticationEntity	Integer		Required	48=Cardholder not authenticated by any entity 49=Cardholder authenticated by integrated chip ard offline pin 50=Cardholder authenticated by card acceptance device 51=Cardholder authenticated by authorizing agent using online pin 52=Cardholder authenticated by merchant using signature 53=Other cardholder authenticating entity present
	CardholderAuthenticationMethod	Integer		Required	48=Cardholder not authenticated 49=PIN 50=electrongic signature analysis 53=manual signature verification 54=other manual verification 83=other systematic verification 84=using electronic ticket environment
	CardholderPresentData	Integer		Required	48=Cardholder present 49=Cardholder not present; unspecified reason 50=Cardholder not present; mail transaction 51=Cardholder not present; phone transaction 52=Cardholder not present; recurring transaction such as subscription 53=Cardholder not present; eCommerce 56=Cardholder not present; recurring billing such as loan or installment payment 65=Reauthorization for full amount

					80=Partial shipment purchase 82=Recurring purchase transaction 84=Pay Button
	PinCaptureCapability	Integer		Required	48=No PIN capture capability 52=PIN capture capability; 4 characters max 53=PIN capture capability; 5 characters max 54=PIN capture capability; 6 characters max 55=PIN capture capability; 7 characters max 56=PIN capture capability; 8 characters max 57=PIN capture capability; 9 characters max 65=PIN capture capability; 10 characters max 66=PIN capture capability; 11 characters max 67=PIN capture capability; 12 characters max
	TerminalCardCaptureCapability	Integer		Required	48=No capture capability 49=Card capture capable
	TerminalCardDataInputCapability	Integer		Required	49=manual; no terminal; voice auth/aru only 50=mag stripe reader capability only (most common value for this method) 51=bar code payment code capable 52=OCR capable 53=Integrated circuit card capable 54=key entry only capable 65=PAN auto entry via contactless mag stripe 66=mag stripe reader, and key entry capable 67=mag stripe, icc, and key entry capable 68=mag stripe and icc capable 69=icc and key entry capable 72=icc reader and contactless capable; mag stripe and manual entry implied 77=pan auto entry via contactless chip
	TerminalCardholderAuthenticationCapability	Integer		Required	48=no electronic authentication capability 49=PIN entry capable 50=electronic signature capable 53=Electronic authentication capability is inoperative
	TerminalDataOutputCapability	Integer		Required	49=No terminal data output capability 50=terminal is capable of output printing only 51=terminal is capable of output display only 52=terminal is capable of output printing and display
	TerminalOperatingEnvironment	Integer		Required	48=no terminal used; voice auth/aru only 49=on card acceptor premises; attended terminal 50=on card acceptor premises; unattended terminal 51=off card acceptor premises; attended 52=off card acceptor premises; unattended 53=on cardholder premises; unattended 77=off card acceptor premises; merchant mobile POS environment including mpos

					80=on card acceptor premises; merchant mobile POS environment including mpos 81=off card acceptor premises; cardholder mobile environment incl home pc, phone, pda 82=on card acceptor premises; cardholder mobile environment incl home pc, phone, pda 83=electronic delivery of product 84=physical delivery of product
ShouldStoreCardData		Boolean		Optional	
EmvPayload		Container			
	EncryptingDeviceType	Integer	2	Required	Must be ProPay encrypted and approved swipe device. The following are ProPay specific reference names. 1=Magtek 20 2=Magtek Flash 3=IdTech Uni Mag 4=Manual card entry 5=Magtek ADynamo 6=Magtek Dynamag 7=Roam Data 8=Roam Data EMV 9=(Partner specific value) 10=Anywhere Commerce EMV 11=(Partner specific value) 12=(Partner specific value) 13=(Partner specific value) 14=Moby3000
	LastChipRead	Integer		Required	1=Successful read 2=Failed read 3=Not a chip card 4=unknown
	TlvData	String		Required	Encrypted

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	Argumentid	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	0=Bank account ACH transfers 1=CreditCard transaction

			2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)
AttemptNumber		Integer	Merchant recognized ProPay transaction identifier
InvoiceNumber			
TransactionInfo		Container	
	GrossAmt	Integer	Gross amount of transaction of pennies in USD, or the number of [currency] without decimals.
	GrossAmtLessNetAmt	Integer	Total amount of fees charged
	NetAmt	Integer	Net amount of transaction after fees charged
	PerTransFee	Integer	Per transaction fee
	Rate	Decimal	Percentage fee
	CreationDate	String	yyyy-mm-dd hh:mm:ss:ms AM
	TransactionHistoryId	Integer	Unique ProPay identifier for future reference/handling of transaction
	ObfuscatedAccountNumber	String	Obfuscated Account Number for the payment method
	AccountName	String	Cardholder repeated or taken from EMV or swipe information.
EmvResponsePayload		Container	
	EmvIssuerAuthenticationData	String	
	EmvIssuerScriptTemplate1	Integer	
	EmvIssuerScriptTemplate2	Integer	
	EmvResponseCode	String	
	TlvData	String	

Sample JSON Request


```
{
 "SessionGuid": "1abc59fa-54a8-419b-a2fc-bdc777d74711",
 "ApplicationId": "B350",
 "BillingData": {
  "CountryCode": 840,
  "PostalCode": "84043"
 },
 "CardData": {
  "Amount": 2500,
  "Comment1": "Auth Comment1",
  "Comment2": "Auth Comment2",
  "CurrencyCode": 840,
  "InvoiceNumber": "",
 "IsQuasiCash":false,
  "MerchantProfileId": 0,
  "TaxAmount": 0
 },
 "CardDataSource": 3,
 "ExternalTransactionIdentifier": "FFA6345",
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111",
   "PayerId": 328814584080 5793,
   "PaymentMethodId": "cb70945e-fc7e-49ec-9724-de27b2b3ff22",
   "PaymentType": 1,
   "AttemptNumber": 1,
   "InvoiceNumber": "",
   "TransactionInfo": {
      "GrossAmt": 2500,
      "GrossAmtLessNetAmt": 2500,
      "NetAmt": 0,
      "PerTransFee": 0,
      "Rate": 0,
      "CreationDate": "2019-04-17 17:15:25.542 AM",
```

```
"PointOfSaleData": {
 "CardDataInputMode": 67,
 "CardDataOutputCapability": 51,
 "CardPresentData": 49,
 "CardholderAuthenticationEntity": 52,
 "CardholderAuthenticationMethod": 53,
 "CardholderPresentData": 48,
 "PinCaptureCapability": 48,
 "TerminalCardCaptureCapability": 48,
 "TerminalCardDataInputCapability": 67,
 "TerminalCardholderAuthenticationCapability": 54,
 "TerminalDataOutputCapability": 51,
 "TerminalOperatingEnvironment": 77
},
"ShouldStoreCardData": true,
"EmvPayload": {
 "EncryptingDeviceType": 10,
 "LastChipRead": 1,
 "TlvData": "9F120A50726f50617920454d56DF790A393837363534",
}
```

```
    "TransactionHistoryId":12345,
    "ObfuscatedAccountNumber": "123456******9876",
    "AccountName": "Test cardholder name"
  },
  "EmvResponsePayload": {
    "EmvIssuerAuthenticationData": null,
    "EmvIssuerScriptTemplate1": null,
    "EmvIssuerScriptTemplate2": null,
    "EmvResponseCode": "8A023030",
    "TlvData": null
  }
}
```

8.3 Authorize an EMV Fallback Transaction

Used to perform an EMV 'Fallback' transaction which is done when an EMV capable card will not work via chip reader, and must be processed via swipe.

How to call this method

Resource URI
/json/AuthorizeEmvFallback

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
ApplicationId		String	4	Optional	May be required if device supports EMV but transaction is swiped.
BillingData		Container			
	CountryCode	Integer	3	Required	USD=840, CAD=124
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
CardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true
CardDataSource		Integer		Required	0=MagStripe 1=Keyed transaction with no reader 2=Keyed transaction with reader present 3=EMV Contact 4=EMV Contactless 5=EMV Fallback
ExternalTransactionIdentifier		String		Optional	
PointOfSaleData		Container			

	CardDataInputMode	Integer		Required	49=Input manually without use of terminal 50=MagStripe 51=barcode or payment code 54=Input manually on terminal 55=Merhant initiated transaction using on-file payment credential 65=Input using mag stripe including track 2 data 67=Chip: Online Mode (commonly used for this method) 70=Chip: Offline Mode 77=Chip: Contactless 78=EMV fallback to mag stripe 80=EMV fallback to mag stripe. Card does not support applications that terminal can peform 82=Ecommerce 83=Ecommerce with no security channel encryption 86=manually input with keyed CID 87=mag stripe with keyed CID 90=Discover specific contactless interface
	CardDataOutputCapability	Integer		Required	49=Card has no output capability (most likely value for this method) 50=Card has mag stripe write output capability 51=Card has integrated chip output capability 83=Card has other output capability
	CardPresentData	Integer		Required	48=Card not present 49=Card present (most likely value for this method) 87=Transponder 88=Contactless chip transacition 90=Digital wallet
	CardholderAuthenticationEntity	Integer		Required	48=Cardholder not authenticated by any entity 49=Cardholder authenticated by integrated chip ard offline pin 50=Cardholder authenticated by card acceptance device 51=Cardholder authenticated by authorizing agent using online pin 52=Cardholder authenticated by merchant using signature 53=Other cardholder authenticating entity present
	CardholderAuthenticationMethod	Integer		Required	48=Cardholder not authenticated 49=PIN 50=electrongic signature analysis 53=manual signature verification 54=other manual verification 83=other systematic verification 84=using electronic ticket environment
	CardholderPresentData	Integer		Required	48=Cardholder present 49=Cardholder not present; unspecified reason 50=Cardholder not present; mail transaction 51=Cardholder not present; phone transaction 52=Cardholder not present; recurring transaction such as subscription 53=Cardholder not present; eCommerce 56=Cardholder not present; recurring billing such as loan or installment payment

					65=Reauthorization for full amount 80=Partial shipment purchase 82=Recurring purchase transaction 84=Pay Button
	PinCaptureCapability	Integer		Required	48=No PIN capture capability 52=PIN capture capability; 4 characters max 53=PIN capture capability; 5 characters max 54=PIN capture capability; 6 characters max 55=PIN capture capability; 7 characters max 56=PIN capture capability; 8 characters max 57=PIN capture capability; 9 characters max 65=PIN capture capability; 10 characters max 66=PIN capture capability; 11 characters max 67=PIN capture capability; 12 characters max
	TerminalCardCapture Capability	Integer		Required	48=No capture capability 49=Card capture capable
	TerminalCardDataInp utCapability	Integer		Required	49=manual; no terminal; voice auth/aru only 50=mag stripe reader capability only (most common value for this method) 51=bar code payment code capable 52=OCR capable 53=Integrated circuit card capable 54=key entry only capable 65=PAN auto entry via contactless mag stripe 66=mag stripe reader, and key entry capable 67=mag stripe, icc, and key entry capable 68=mag stripe and icc capable 69=icc and key entry capable 72=icc reader and contactless capable; mag stripe and manual entry implied 77=pan auto entry via contactless chip
	TerminalCardholderA uthenticationCapabili ty	Integer		Required	48=no electronic authentication capability 49=PIN entry capable 50=electronic signature capable 53=Electronic authentication capability is inoperative
	TerminalDataOutputC apability	Integer		Required	49=No terminal data output capability 50=terminal is capable of output printing only 51=terminal is capable of output display only 52=terminal is capable of output printing and display
	TerminalOperatingEnv ironment	Integer		Required	48=no terminal used; voice auth/aru only 49=on card acceptor premises; attended terminal 50=on card acceptor premises; unattended terminal 51=off card acceptor premises; attended 52=off card acceptor premises; unattended 53=on cardholder premises; unattended 77=off card acceptor premises; merchant mobile POS environment including mpos

					80=on card acceptor premises; merchant mobile POS environment including mpos 81=off card acceptor premises; cardholder mobile environment incl home pc, phone, pda 82=on card acceptor premises; cardholder mobile environment incl home pc, phone, pda 83=electronic delivery of product 84=physical delivery of product
ShouldStoreCardData		Boolean		Optional	
SwipeCardData		Container			
	Name	String		Optional	Cardholder
	EncryptedTrackData (SwipeCardData)	Container			
	DeviceType	Integer	2	Required	Must be ProPay encrypted and approved swipe device. The following are ProPay specific reference names. 1=Magtek 20 2=Magtek Flash 3=IdTech Uni Mag 4=Manual card entry 5=Magtek ADynamo 6=Magtek Dynamag 7=Roam Data 8=Roam Data EMV 9=(Partner specific value) 10=Anywhere Commerce EMV 11=(Partner specific value) 12=(Partner specific value) 13=(Partner specific value) 14=Moby3000
	EncryptedTrack2Data	Byte[]		Required	
	EncryptedTrackData	Byte[]		Required	
	KeySerialNumber	Byte[]		Required	
LastChipRead		Integer		Required	1=Successful read 2=Failed read 3=Not a chip card 4=unknown

Response Elements

Element	Nested Element	Type	Notes
ResultCode		Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	Argumentid	Integer	
	ResultCode	Integer	
	ResultMessage	String	Further describes result provided via ResultCode

AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	
AttemptNumber		Integer	Merchant recognized ProPay transaction identifier
TransactionInfo		Container	
	GrossAmt	Integer	Gross amount of transaction of pennies in USD, or the number of [currency] without decimals.
	GrossAmtLessNetAmt	Integer	Total amount of fees charged
	NetAmt	Integer	Net amount of transaction after fees charged
	PerTransFee	Integer	Per transaction fee
	Rate	Decimal	Percentage fee
	CreationDate	String	yyyy-mm-dd hh:mm:ss:ms AM
	TransactionHistoryId	Integer	Unique ProPay identifier for future reference/handling of transaction
	ObfuscatedAccount Number	String	Obfuscated Account Number for the payment method
	AccountName	String	Cardholder repeated or taken from EMV or swipe information.

Sample JSON Request

```
{
 “SessionGuid”: “5acf5fd5-7941-44e9-82d9-fbe024d870f4”,
 “ApplicationId”: “B350”,
 “BillingData”: {

 },
 “CardData”: {
  “Amount”: 7700,
  “Comment1”: “Authorize Comment 1”,
  “Comment2”: “Authorize Comment 2”,
  “CurrencyCode”: 840,
  “InvoiceNumber”: null,
  “IsQuasiCash”:false,
  “MerchantProfileId”: 0,
  “TaxAmount”: 0
 },
 “CardDataSource”: 5,
 “ExternalTransactionIdentifier”: “BBCV8899”,
 “PointOfSaleData”: {
  “CardDataInputMode”: 78,
  “CardDataOutputCapability”: 51,
  “CardPresentData”: 49,
  “CardholderAuthenticationEntity”: 48,
  “CardholderAuthenticationMethod”: 48,
  “CardholderPresentData”: 48,
  “PinCaptureCapability”: 48,
  “TerminalCardCaptureCapability”: 48,
```

Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111",
   "PayerId": 4087998504689416,
   "PaymentMethodId": "00000000-0000-0000-0000-000000000000",
   "PaymentType": null,
   "AttemptNumber": 16300,
   "InvoiceNumber": null,
   "TransactionInfo": {
      "GrossAmt": 7700,
      "GrossAmtLessNetAmt": 7700,
      "NetAmt": 0,
      "PerTransFee": 0,
      "Rate": 0,
      "CreationDate": "2019-04-17 17:15:25.542 AM",
      "TransactionHistoryId":12345,
      "ObfuscatedAccountNumber": "123456******9876",
      "AccountName": "Test cardholder name"
   }
}
```

```
 "TerminalCardDataInputCapability": 68,
 "TerminalCardholderAuthenticationCapability": 48,
 "TerminalDataOutputCapability": 51,
 "TerminalOperatingEnvironment": 80
},
"ShouldStoreCardData": false,
"SwipeCardData": {
 "EncryptedTrackData": {
  "DeviceType": 12,
  "EncryptedTrack2Data": [39, 28, 194, 0, 62, 156, 33, 172, 108, 208, 85, 224, 240, 243] ,
  "EncryptedTrackData": [39, 28, 194, 0, 62, 156, 33, 172, 108, 208, 85, 128, 121, 240, 243],
  "KeySerialNumber": [255, 255, 255, 0, 6, 0, 62, 32, 2, 177],
  "Name": "Fallback Swipe"
 }
},
"LastChipRead": 3
}
```

8.4 Authorize a Manually Keyed Card

Pre-Auth of manually keyed card.

How to call this method

Resource URI
/json/AuthorizeManualEntry

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
ApplicationId		String	4	Optional	May be required if device supports EMV but transaction is swiped.
BillingData		Container			
	CountryCode	Integer	3	Required	
	PostalCode	String	10	Optional	Note: while PostalCode is optional, to receive the best credit card processing rates you will need to provide it.
CardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Required	USD=840, CAD=124
	InvoiceNumber	String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoice #, card # and amount in 1 minute. If SEPA then value is required.
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	TaxAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount
	IsQuasiCash	Boolean	5	Optional	Valid values are: • true • false If the transaction is IsQuasiCash then this needs to be set to true
CardDataSource		Integer		Required	0=MagStripe 1=Keyed transaction with no reader 2=Keyed transaction with reader present 3=EMV Contact 4=EMV Contactless 5=EMV Fallback
ExternalTransactionIdentifier		String		Optional	
PointOfSaleData		Container			
	CardDataInputMode	Integer		Required	49=Input manually without use of terminal

					50=MagStripe 51=barcode or payment code 54=Input manually on terminal 55=Merhant initiated transaction using on-file payment credential 65=Input using mag stripe including track 2 data 67=Chip: Online Mode (commonly used for this method) 70=Chip: Offline Mode 77=Chip: Contactless 78=EMV fallback to mag stripe 80=EMV fallback to mag stripe. Card does not support applications that terminal can peform 82=Ecommerce 83=Ecommerce with no security channel encryption 86=manually input with keyed CID 87=mag stripe with keyed CID 90=Discover specific contactless interface
	CardDataOutputCapability	Integer		Required	49=Card has no output capability (most likely value for this method) 50=Card has mag stripe write output capability 51=Card has integrated chip output capability 83=Card has other output capability
	CardPresentData	Integer		Required	48=Card not present 49=Card present (most likely value for this method) 87=Transponder 88=Contactless chip transacition 90=Digital wallet
	CardholderAuthenticationEntity	Integer		Required	48=Cardholder not authenticated by any entity 49=Cardholder authenticated by integrated chip ard offline pin 50=Cardholder authenticated by card acceptance device 51=Cardholder authenticated by authorizing agent using online pin 52=Cardholder authenticated by merchant using signature 53=Other cardholder authenticating entity present
	CardholderAuthenticationMethod	Integer		Required	48=Cardholder not authenticated 49=PIN 50=electrongic signature analysis 53=manual signature verification 54=other manual verification 83=other systematic verification 84=using electronic ticket environment
	CardholderPresentData	Integer		Required	48=Cardholder present 49=Cardholder not present; unspecified reason 50=Cardholder not present; mail transaction 51=Cardholder not present; phone transaction 52=Cardholder not present; recurring transaction such as subscription 53=Cardholder not present; eCommerce 56=Cardholder not present; recurring billing such as loan or installment payment 65=Reauthorization for full amount

					80=Partial shipment purchase 82=Recurring purchase transaction 84=Pay Button
	PinCaptureCapability	Integer		Required	48=No PIN capture capability 52=PIN capture capability; 4 characters max 53=PIN capture capability; 5 characters max 54=PIN capture capability; 6 characters max 55=PIN capture capability; 7 characters max 56=PIN capture capability; 8 characters max 57=PIN capture capability; 9 characters max 65=PIN capture capability; 10 characters max 66=PIN capture capability; 11 characters max 67=PIN capture capability; 12 characters max
	TerminalCardCaptureCapability	Integer		Required	48=No capture capability 49=Card capture capable
	TerminalCardDataInputCapability	Integer		Required	49=manual; no terminal; voice auth/aru only 50=mag stripe reader capability only (most common value for this method) 51=bar code payment code capable 52=OCR capable 53=Integrated circuit card capable 54=key entry only capable 65=PAN auto entry via contactless mag stripe 66=mag stripe reader, and key entry capable 67=mag stripe, icc, and key entry capable 68=mag stripe and icc capable 69=icc and key entry capable 72=icc reader and contactless capable; mag stripe and manual entry implied 77=pan auto entry via contactless chip
	TerminalCardholderAuthenticationCapability	Integer		Required	48=no electronic authentication capability 49=PIN entry capable 50=electronic signature capable 53=Electronic authentication capability is inoperative
	TerminalDataOutputCapability	Integer		Required	49=No terminal data output capability 50=terminal is capable of output printing only 51=terminal is capable of output display only 52=terminal is capable of output printing and display
	TerminalOperatingEnvironment	Integer		Required	48=no terminal used; voice auth/aru only 49=on card acceptor premises; attended terminal 50=on card acceptor premises; unattended terminal 51=off card acceptor premises; attended 52=off card acceptor premises; unattended 53=on cardholder premises; unattended 77=off card acceptor premises; merchant mobile POS environment including mpos

					80=on card acceptor premises; merchant mobile POS environment including mpos 81=off card acceptor premises; cardholder mobile environment incl home pc, phone, pda 82=on card acceptor premises; cardholder mobile environment incl home pc, phone, pda 83=electronic delivery of product 84=physical delivery of product
ShouldStoreCardData		Boolean		Optional	
ManualCardData		Container			
	CVV	String		Optional	CVV code
	CardNumber	String		Required	Standard cc numbers which must pass Luhn check
	ExpirationDate	String		Required	The expiration date in MMYY format
	Name	String		Optional	Cardholder Name

Response Elements

Element	Nested Element	Type	Notes
ResultCode		Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	Argumentid	Integer	
	ResultCode	Integer	
	ResultMessage	String	Further describes result provided via ResultCode
AuthorizationCode		String	Returned authorization code for the transaction
PayerId		Integer	The PayerId the payment method was stored under
PaymentMethodId		Guid	The ID of the stored payment method, also known as token
PaymentType		Integer	0=Bank account ACH transfers 1=CreditCard transaction 2=ProPay to ProPay transactions 3=ProPay to ProPay transfer (Push)
AttemptNumber		Integer	Merchant recognized ProPay transaction identifier
InvoiceNumber			
TransactionInfo		Container	
	GrossAmt	Integer	Gross amount of transaction of pennies in USD, or the number of [currency] without decimals.
	GrossAmtLessNetAmt	Integer	Total amount of fees charged
	NetAmt	Integer	Net amount of transaction after fees charged
	PerTransFee	Integer	Per transaction fee
	Rate	Decimal	Percentage fee
	CreationDate	String	yyyy-mm-dd hh:mm:ss:ms AM

	TransactionHistoryId	Integer	Unique ProPay identifier for future reference/handling of transaction
	ObfuscatedAccountNumber	String	Obfuscated Account Number for the payment method
	AccountName	String	Cardholder repeated or taken from EMV or swipe information.

Sample JSON Request


```
{
 "SessionGuid": "a83c6703-bdd9-45ef-838d-11ca0bf62e47",
 "ApplicationId": "B350",
 "BillingData": {
  "CountryCode": 840,
  "PostalCode": "30308"
 },
 "CardData": {
  "Amount": 7700,
  "Comment1": "Authorize Comment 1",
  "Comment2": "Authorize Comment 2",
  "CurrencyCode": 840,
  "InvoiceNumber": null,
  "IsQuasiCash:false,
  "MerchantProfileId": 0,
  "TaxAmount": 0
 },
 "CardDataSource": 1,
 "ExternalTransactionIdentifier": "B4CCC48899",
 "PointOfSaleData": {
  "CardDataInputMode": 54,
  "CardDataOutputCapability": 51,
  "CardPresentData": 49,
  "CardholderAuthenticationEntity": 48,
  "CardholderAuthenticationMethod": 48,
  "CardholderPresentData": 48,
  "PinCaptureCapability": 48,
  "TerminalCardCaptureCapability": 48,
  "TerminalCardDataInputCapability": 68,
  "TerminalCardholderAuthenticationCapability": 48,
  "TerminalDataOutputCapability": 51,
  "TerminalOperatingEnvironment": 80
 },
 "ShouldStoreCardData": true,
 "ManualCardData": {
  "CVV": "888",
  "CardNumber": "5105105105105100",
  "ExpirationDate": "0820",
  "Name": "Manual Entry"
 }
}
```


Response


```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AuthorizationCode": "A11111",
   "PayerId": 4944450652213798,
   "PaymentMethodId": "8708a686-6282-4530-b56b-9c459799959b",
   "PaymentType": 1,
   "AttemptNumber": 16317,
   "InvoiceNumber": null,
   "TransactionInfo": {
      "GrossAmt": 7700,
      "GrossAmtLessNetAmt": 7700,
      "NetAmt": 0,
      "PerTransFee": 0,
      "Rate": 0,
      "CreationDate": "2019-04-17 17:15:25.542 AM",
      "TransactionHistoryId":12345,
      "ObfuscatedAccountNumber": "123456******9876",
      "AccountName": "Test cardholder name"
   }
}
```

8.5 Capture a Transaction

Used to capture a transaction making it eligible for funding by ProPay into the Merchant's account. Following capture, ProPay will fund based on a schedule that is published but configurable by merchant or partner program.

How to call this method

Resource URI
/json/CaptureAuthorizedTransaction

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
AttemptNumber		Integer	10	Required	From authorization response
BillingData		Container			
	Email	String	100	Optional	Email for the receipt. Only pass if ProPay will send the receipt. Recommended not to enter an email here.
CardData		Container			
	Amount	Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
	Comment1	String	128	Optional	Transaction descriptor.
	Comment2	String	128	Optional	Transaction descriptor.
	CurrencyCode	Integer	3	Optional	USD=840, CAD=124
	MerchantProfileId	Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
	SignatureBlock	Byte[]	3	Optional	ProPay will save this value if provided and can use it to help represent a chargeback if one occurs.
	TipAmount	Integer	10	Optional	The value representing the number of pennies in USD, or the number of [currency] without decimals for the tax amount

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	Argumentid	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode
AttemptNumber		Integer	Transaction ID of the result of an authorization request that was captured
AuthorizationCode		String	Returned authorization code for the transaction
TransactionInfo		Container	
	GrossAmt	Integer	Gross amount of transaction of pennies in USD, or the number of [currency] without decimals.
	GrossAmtLessNetAmt	Integer	Total amount of fees charged

	NetAmt	Integer	Net amount of transaction after fees charged
	PerTransFee	Integer	Per transaction fee
	Rate	Decimal	Percentage fee
	TransactionHistoryId	Integer	Unique ProPay identifier for future reference/handling of transaction

Sample JSON Request

```
{
  "SessionGuid": "a83c6703-bdd9-45ef-838d-11ca0bf62e47",
  "AttemptNumber": 1,
  "BillingData": {
    "Email": "test@qamail.com"
  },
  "CardData": {
    "Amount": 100,
    "Comment1": "Auth Comment 1",
    "Comment2": "Auth Comment 2",
    "CurrencyCode": 840,
    "MerchantProfileId": 0,
    "SignatureBlock": [1],
    "TipAmount": 0
  }
}
```

Response

```
{
   "Result": {
      "ArgumentId": 0,
      "ResultCode": 0,
      "ResultMessage": ""
   },
   "AttemptNumber": 1
   "AuthorizationCode": null,
   "TransactionInfo": {
      "GrossAmt": 100,
      "GrossAmtLessNetAmt": 0,
      "NetAmt": 100,
      "PerTransFee": 25,
      "Rate": 2.1,
      "TransactionHistoryId":12345
   }
}
```

8.6 Void or Refund a Transaction

Used to attempt a void/refund of a card. As with other ProPay solutions, the appropriate action will take place based on the current state of a transaction.

How to call this method

Resource URI
/json/VoidOrRefund

Account Identification - Required

Element	Nested Element	Type	Max	Required	Notes
SessionGuid		String	36	Required	Primary authentication mechanism after initial login
AttemptNumber		Integer	10	Required	From authorization response
Amount		Integer	10	Required	The value representing the number of pennies in USD, or the number of [currency] without decimals.
Comment1		String	128	Optional	Transaction descriptor.
Comment2		String	128	Optional	Transaction descriptor.
CurrencyCode		Integer	3	Optional	USD=840, CAD=124
MerchantProfileId		Integer	19	Optional	Merchant Profile Id created for use with a particular gateway
EmvPayloadData		Container			
	EncryptingDeviceType	Integer		Optional	Must be ProPay encrypted and approved swipe device. The following are ProPay specific reference names. 1=Magtek 20 2=Magtek Flash 3=IdTech Uni Mag 4=Manual card entry 5=Magtek ADynamo 6=Magtek Dynamag 7=Roam Data 8=Roam Data EMV 9=(Partner specific value) 10=Anywhere Commerce EMV 11=(Partner specific value) 12=(Partner specific value) 13=(Partner specific value) 14=Moby3000
	LastChipRead	Integer		Optional	1=Successful read 2=Failed read 3=Not a chip card 4=unknown
	TlvData	String		Optional	
	VoidReason	Integer		Optional	Required for EMV voids. Do not send for refunds or non EMV voids.

					2501=voided by customer 2502=pos device timeout 2503=no confirmation from pos 2504=partial reversal 2516=premature chip card removal 2517=chip declined transaction
ExternalTransactionIdentifier		String		Optional	
InvoiceNumber		String	50	Optional	Recommended. Transaction descriptor. *ProPay gateway rejects duplicates for same invoic #, card # and amount in 1 minute. If SEPA then value is required.
MerchantProfileId		Integer	19	Optional	Merchant Profile Id created for use with a particular gateway

Response Elements

Element	Nested Element	Type	Notes
Result		Container	
	ArgumentId	Integer	
	ResultCode	Integer	Result of the transaction request. Uses Standard ProPay ProtectPay response values.
	ResultMessage	String	Further describes result provided via ResultCode

Sample JSON Request

```
{
 "SessionGuid": "e69f411c-b4cc-460f-bdeb-05c9b546669c",
 "Amount": 3100,
 "AttemptNumber": "1",
 "Comment1": null,
 "Comment2": null,
 "CurrencyCode": 840,
 "EmvPayloadData": {
  "EncryptingDeviceType": 12,
  "LastChipRead": 1,
  "TlvData": “,
  "VoidReason": 2501
 },
 "ExternalTransactionIdentifier": "AW56”,
 “InvoiceNumber”: “INV4345”,
 “MerchantProfileId": 0
}
```

Response

```
{
   "Result": {
      “ArgumentId”: 0,
      “ResultCode”: 0,
      “ResultMessage”: “”
   }
}
```