Powershop API

Developer Documentation

API version 1.0

Documentation version 0.7

Last updated 6 December 2010

Table of Contents

Getting Access to the API

Before you can use the Powershop API, you must sign up for an API account for your product. Send an e-mail to api@powershop.co.nz asking for access. Please include your product name and the e-mail address you used to sign up to Powershop.

All API accounts have a controlled number of requests per hour that they may submit. If you feel that your application will be submitting many requests, please mention this to the Powershop team.

Technical Overview

The Powershop API is available over HTTPS. It returns responses in JSON or XML format.

This API uses the OAuth protocol. This allows users to authenticate themselves without entering their password into your application. OAuth libraries are available for all major development platforms, including PHP, Ruby and the iPhone.

For information about OAuth, visit:
http://oauth.net/

It is a condition of using the Powershop API that you are never able to directly capture the user's password. This includes, for example, using a web control on the iPhone inside your application. Please search for “iphone oauth” to see how others have implemented OAuth for iPhone apps.

OAuth Configuration

Your OAuth library will require several URLs to be configured. For Powershop, these are:

Request Token URL:
https://secure.powershop.co.nz/external_api/oauth/request_token

Authorize URL:
https://secure.powershop.co.nz/external_api/oauth/authorize

Access Token URL:
https://secure.powershop.co.nz/external_api/oauth/access_token

For the test environment, replace “https://secure.powershop.co.nz” with “http://suppliertest.youdo.co.nz”.

A callback URL passed to the request token URL is required, as per the OAuth 1.0a specification. Note that your OAuth library may not require this parameter; you must manually specify it in this case or update the library to the latest version.

API Documentation

URL Format

Once you have received an access token using OAuth, use your OAuth library to call the API you are interested in. The format of API URLs is:

https://secure.powershop.co.nz/external_api/v1/***apiname*.*format***

where ***apiname*** is one of **customer**, **meter_readings**, **products** or **top_up**
where **format** is one of **js** or **xml**.

If you use format **js**, the response will be a JSON string. If you use format **xml**, the reponse will be in XML.

Parameters supplied to the calls should be passed URL-encoded. For example, a request to find meter readings for ICP 0123456789 between the 1st and 30th of September 2009, with a JSON response requested, would look like:

https://secure.powershop.co.nz/external_api/v1/meter_readings.js?icp_number=0123456789&
start_date=2009-09-01&end_date=2009-09-30

Your OAuth library will add OAuth token information and a signature before passing it to the Powershop server.

See below for details about each of the different API calls. Note that calls that get information require you use an HTTP GET, and calls that update information require you use an HTTP POST.

Response Codes

A successful call to the API will return an HTTP 200 status code, and either JSON or XML-encoded data.

An unsuccessful call will return HTTP status codes 400, 401 or 503, depending on the error. A single line of text describing the error will also be returned. Error messages common to all API calls are:

[E900] Invalid OAuth signature or consumer key
[E901] Unknown OAuth signature method
[E902] The application has exceeded the request rate limit for this hour
[E903] Invalid OAuth token
[E904] OAuth consumer is disabled
[E905] Expired request token
[E906] Invalid verifier
[E907] oauth_token not found
[E908] oauth_callback must be supplied

As the text of error messages may change, please use the [Exxx] error number to detect a particular error message.

customer API call

HTTP method:
GET

API URL:
https://secure.powershop.co.nz/external_api/v1/customer.**format**

Parameters:
none

Description:
The customer API call returns information about the currently authenticated customer. It does not take any parameters.

Returns:
version = 1.0
result =

properties **[Array]** =	
icp_number	*the property's ICP number*
status	*see "notes" section below*
status_detail	*see "notes" section below*
address =	*the address of the property*
property_name	
flat_number	
street_number	
street_name	
suburb	
district	
region	
start_date	*yyyy-mm-dd; date property went online*
end_date	*yyyy-mm-dd or nil; date property offline*
unit_balance	*customer's unit balance; may be negative*
daily_consumption	*number of kWh currently used per day*
last_account_review_at	*yyyy-mm-dd hh:mm:ss*
registers **[Array]** =	
register_number	
description	*customer-supplied description*
dials	*number of dials that the register has*
hidden	*1 if hidden; 0 otherwise*
last_reading_at	*yyyy-mm-dd hh:mm:ss*
last_reading_value	
last_reading_type	*actual, customer, estimated, derived*
estimated_reading_value	*estimated reading as of now*

Errors:
No specific errors returned by this method.

(continues next page)

Notes:
The status field is one of the following values:

active	*the property is currently being supplied by Powershop*
inbound	*the customer is moving in to switching to Powershop and the property is not ready for supply yet*
outbound	*the customer is moving out or switching away from Powershop*
archived	*the property is no longer active with Powershop*

A property's lifecycle is inbound → active → outbound → archived.

The status_detail field contains a text string when the property's status is either inbound or outbound. This text details what stage the inbound or outbound process is currently in, for example, “waiting for switch from other retailer.” This text string can be displayed directly to the user.

Note that you can only use the methods below on ICPs that have a status of active. Trying to use it on other ICPs will return error message E009.

The unit_balance field shows the current kWh balance of the account. A negative value means that the customer is that many units in arrears and will have to purchase power to make up the deficit at the next account review. A positive value means the customer has purchased more power than they have currently used.

A customer may hide a register if it is a “low use” register. This means you should generally not display it on your user interface.

Note that a customer may have several properties on the same account. Your user interface should allow for this possibility by letting the user select which property they wish to work with, if they have more than one.

meter_readings GET API call

HTTP method:

GET

API URL:

https://secure.powershop.co.nz/external_api/v1/meter_readings.**format**

Parameters:

icp_number	*the ICP number of the property*
start_date	*yyyy-mm-dd; retrieve readings from this date*
end_date	*yyyy-mm-dd; retrieve readings up until this date*

Description:

The GET meter_readings API call returns readings for the given ICP.

They may be "actual" - retrieved from a meter reading company, "customer" - supplied by the customer, "estimated" - estimated by Powershop, normally for account review purposes, or "derived" - calculated using smart meter data.

Returns:

version = 1.0

result **[Array]** =

register_number	*reading was taken on this register number*
read_at	*yyyy-mm-dd hh:mm:ss*
reading_type	*actual, customer, estimated, derived*
reading_value	*the reading*

Errors:

[E000] The icp_number parameter must be specified for this call
[E000] The start_date parameter must be specified for this call
[E000] The end_date parameter must be specified for this call
[E001] Dates must be in YYYY-MM-DD format
[E008] icp_number specified not found for this customer
[E009] This property is not active

meter_readings POST API call

HTTP method:

POST

API URL:

https://secure.powershop.co.nz/external_api/v1/meter_readings.**format**

Parameters:

icp_number	*the ICP number of the property*
readings[**register_number**]	*the reading for **register_number***

Description:

The POST meter_readings API call allows users to enter a reading for the specified ICP.

Note that a full set of readings must be specified, otherwise all readings will be rejected. A full set means that all non-hidden registers must have a valid reading. The reading may not be less than the previous actual reading, and must not be exceed the daily usage thresholds that Powershop calculates for the ICP. If it does, an failure message will be returned.

Returns:

version = 1.0
result =

result	*"success" or "failure"*
message	*if result is "failure", a message describing the failure*

Errors:

[E000] The icp_number parameter must be specified for this call
[E002] Readings must be in an array format
[E003] Register number not found
[E004] Only numbers may be present in the reading value
[E005] A reading for each active register must be supplied
[E008] icp_number specified not found for this customer
[E009] This property is not active

Notes:

It is recommended that you use the "dials" number, found in the results of the customer API call against each register, to show how the user many dials the register has and allow entry of exactly that many digits. Note also that under some cases the "dials" number may not be specified, in which case a UI handling up to 8 digits should be provided.

products API call

HTTP method:
GET

API URL:
https://secure.powershop.co.nz/external_api/v1/products.**format**

Parameters:

icp_number	*the ICP number of the property*

Description:
The products API call returns the products available for purchase from the specified property, and the per unit price of the products.

Returns:
version = 1.0
result **[Array]** =

name	*the name of the product*
description	*a long HTML-formatted description of the product*
type	*nominated, special, powerpack*
price_per_unit	*unit price to 4d.p.*
image_url	*URL for 130x130 or 165x130 pixel product image*

Errors:
[E000] The icp_number parameter must be specified for this call
[E008] icp_number specified not found for this customer
[E009] This property is not active

Notes:
Under usual circumstances, a price returned by this call will be valid for purchase through the Powershop web site until midnight.

The image_url parameter will not be present if there is no image associated with the product. The image URL supports the HTTP If-Modified-Since header, so you may wish to speed up your user's experience by caching the image. Note that the image may change if the product is out of stock.

top_up GET API call

HTTP method:

GET

API URL:

https://secure.powershop.co.nz/external_api/v1/top_up.**format**

Parameters:

icp_number	*the ICP number of the property*

Description:

The top_up GET API call returns an offer of how much it will cost the user to top up their account with their default preferred product (or the cheapest product currently on offer if they have not selected a preferred product.)

This call does not make the purchase; it just returns a price. The price must be displayed to the user and their consent granted before the purchase is made. See the top_up POST API call.

Returns:

version = 1.0

result =

unit_balance	*the current unit balance for this property*
product_name	*the name of the product that will be purchased*
price_per_unit	*the price per unit*
total_price	*the total price of the purchase*
offer_key	*a key that must be passed to the top_up POST API*

Errors:

[E000] The icp_number parameter must be specified for this call
[E008] icp_number specified not found for this customer
[E009] This property is not active

Notes:

If the customer's unit balance is zero or positive, only the “unit_balance” key will be returned.

top_up POST API call

HTTP method:

POST

API URL:

https://secure.powershop.co.nz/external_api/v1/top_up.**format**

Parameters:

icp_number	*the ICP number of the property*
offer_key	*the offer key returned by the top_up GET API call*

Description:

The top_up POST API call makes a purchase of power, topping up the customer's unit balance to 0. The top_up GET API must be called before calling this.

Returns:

version = 1.0
result =

result	*"success" or "failure"*
message	*if result is "failure", a description of the failure*

Errors:

[E000] The icp_number parameter must be specified for this call
[E008] icp_number specified not found for this customer
[E006] Invalid offer_key value
[E009] This property is not active

Notes:

If the result is "failure", the message field may be displayed to the user to inform them why the purchase failed.

contact_us POST API call

HTTP method:

POST

API URL:

https://secure.powershop.co.nz/external_api/v1/contact_us.**format**

Parameters:

comment	*text of the comment to be posted to Powershop*

Description:

The contact_us POST API call posts some feedback to Powershop. The currently authenticated user's name and e-mail address are automatically attached to the feedback so that Powershop can send them a response.

The comment field may be up to 64KB in length.

Returns:

version = 1.0
result =

result	*"success" or "failure"*
message	*if result is "failure", a description of the failure*

Errors:

[E000] The comment parameter must be specified for this call

usage_data GET API call

HTTP method:

GET

API URL:

https://secure.powershop.co.nz/external_api/v1/usage_data.**format**

Parameters:

icp_number	*the ICP number of the property*
start_date	*yyyy-mm-dd*
end_date	*yyyy-mm-dd; not more than a year after start_date*

Description:

The usage_data call returns pre-calculated data about the user's usage, ready to plot on a graph. It is the same data that is shown on the Powershop web site graphs. Powershop suggests you use this data when displaying graphs to give a seamless experience between the web site and your application.

If more than a year's data is requested, the end date will be truncated so that only a year's worth after the start date will be returned.

Returns:

version = 1.0
result =

start_date	*the start date of the data returned*
end_date	*the end date of the data returned*
graph_type	*optional – see below*
registers **[Array]**	*an array of register numbers*
data **[Array]** =	
day_data	*number of kWh for this day*

Errors:

[E000] The icp_number parameter must be specified for this call
[E008] icp_number specified not found for this customer
[E009] This property is not active
[E010] Invalid graph_type value

Notes:

The graph_type variable is optional. If specified, it must be either *smoothed* or *raw*. If not specified, it defaults to *smoothed*. Powershop will be changing these graph types in the future, and it is likely that this field will become unused.

An array of numbers is returned in the data variable. If you requested 60 days, the data array will have 60 elements, where the first element is the usage on the start date and the last element is the usage on the end date. Each element contains the number of kilowatt-hours (kWh) used on that day. This may be a floating point number, and under certain unusual circumstances, may be negative.

promotions GET API call

HTTP method:
GET

API URL:
https://secure.powershop.co.nz/external_api/v1/promotions.**format**

Parameters:
none

Description:
The promotions call returns all active referral promotions. The user can use one of these promotions to refer Powershop to a friend.

Returns:
version = 1.0
result =
promotions **[Array]** =

promotion_id	*a numeric identifier for this promotion*
name	*the promotion's name*
description	*a plain text description*
image_url	*a URL of an image related to this promotion*
start_date	*the start date of the data returned*
end_date	*the end date of the data returned*
t_and_c_accepted	*whether the user has accepted T&C*

Errors:
none

Notes:
An empty array of promotions may be returned if there are no active referral promotions. At present, Powershop only runs at most one referral promotion at a time, but this may change in the future.

The promotion description is UTF-8 text. It may contain linefeed characters.

referral POST API call

HTTP method:

POST

API URL:

https://secure.powershop.co.nz/external_api/v1/referral.**format**

Parameters:

promotion_id	*the numeric identifier of the promotion*
name	*the name of the referee*
email	*the e-mail address of the referee*
return_url	*the URL to return to once the user has accepted the T&C*

Description:

This call allows a user to refer a friend to Powershop. The promotion ID can be found from the results of the promotions GET call. A name and valid e-mail address must be specified.

Before a referral can be made, the logged in user must accept the terms and conditions of the promotion. They only need to accept the T&C once per promotion. If the user has not yet accepted the T&C, a result of "redirect" will be returned from the API, and the application must display the specified URL in an embedded browser, or redirect the user to the URL if in a web application. When the user has accepted the T&C, the referral POST request can be resubmitted.

Returns:

version = 1.0
result =

result	*"success" or "failure" or "redirect"*
message	*if result is "failure", a description of the failure*
url	*if result is "redirect", a URL to display to the user*

Errors:

[E000] The promotion_id parameter must be specified for this call
[E000] The name parameter must be specified for this call
[E000] The email parameter must be specified for this call
[E011] Invalid e-mail address
[E012] Invalid promotion id

Notes:

If the result is "failure", the message field may be displayed to the user to inform them why the referral failed.