

Package: octopusR (via r-universe)

October 1, 2026

Title Interact with the 'Octopus Energy' API

Version 1.0.1.9000

Description A simple wrapper for the 'Octopus Energy' REST API <<https://developer.octopus.energy/rest/>>. It handles authentication, by storing a provided API key and meter details. Implemented endpoints include 'products' for viewing tariff details and 'consumption' for viewing meter consumption data.

License MIT + file LICENSE

URL <https://github.com/Moohan/octopusR>,
<https://moohan.github.io/octopusR/>

BugReports <https://github.com/Moohan/octopusR/issues>

Depends R (>= 4.1.0)

Imports askpass, cli, glue, httr2, rlang, tibble

Suggests covr, data.table, jsonlite, lubridate (>= 0.2.1), mockery, spelling, testthat (>= 3.0.0), vctrs

Config/roxygen2/markdown true

Config/roxygen2/version 8.1.0

Config/testthat/edition 3

Config/testthat/parallel true

Config/testthat/start-first get_consumption

Encoding UTF-8

Language en-GB

LazyData true

Config/pak/sysreqs libssl-dev

Repository <https://moohan.r-universe.dev>

Date/Publication 2026-10-01 05:01:11 UTC

RemoteUrl <https://github.com/moohan/octopusr>

RemoteRef HEAD

RemoteSha 2a0fda013b4b023c4d36deed7f1d6d2770ca1afc

Contents

combine_consumption	2
get_consumption	3
get_meter_gsp	4
get_products	5
set_api_key	6
set_meter_details	6
Index	8

combine_consumption	<i>Combine import and export consumption data</i>
---------------------	---

Description

Combine consumption data from import and export meters into a single tibble with separate columns for import and export consumption. This is useful for users with solar panels or other export generation.

Usage

```
combine_consumption(  
  import_mpan = NULL,  
  import_serial = NULL,  
  export_mpan = NULL,  
  export_serial = NULL,  
  api_key = get_api_key(),  
  period_from = NULL,  
  period_to = NULL,  
  tz = NULL,  
  order_by = c("-period", "period"),  
  group_by = c("hour", "day", "week", "month", "quarter")  
)
```

Arguments

import_mpan	The import meter MPAN
import_serial	The import meter serial number
export_mpan	The export meter MPAN
export_serial	The export meter serial number
api_key	API key for authentication
period_from	Show consumption from the given datetime (inclusive)
period_to	Show consumption to the given datetime (exclusive)
tz	Time zone for date parsing (requires lubridate)
order_by	Ordering of results returned
group_by	Aggregates consumption over a specified time period

Value

a [tibble](#) with import_consumption, export_consumption, and net_consumption columns

get_consumption	<i>List consumption for a meter</i>
-----------------	-------------------------------------

Description

Return a list of consumption values for half-hour periods for a given meter-point and meter.

Unit of measurement:

- Electricity meters: kWh
- SMETS1 Secure gas meters: kWh
- SMETS2 gas meters: m³

Parsing dates:

To return dates properly parsed [lubridate](#) is required. Use the 'tz' parameter to specify a time zone e.g. tz = "UTC", the default (tz = NULL) will return the dates unparsed, as characters.

Usage

```
get_consumption(
  meter_type = c("electricity", "gas"),
  mpan_mprn = NULL,
  serial_number = NULL,
  direction = NULL,
  api_key = get_api_key(),
  period_from = NULL,
  period_to = NULL,
  tz = NULL,
  order_by = c("-period", "period"),
  page_size = NULL,
  group_by = c("hour", "day", "week", "month", "quarter")
)
```

Arguments

meter_type	Type of meter-point, electricity or gas
mpan_mprn	The electricity meter-point's MPAN or gas meter-point's MPRN.
serial_number	The meter's serial number.
direction	For electricity meters, specify "import", "export", or NULL (default). When NULL, uses the legacy single MPAN storage.
api_key	Your API key. If you are an Octopus Energy customer, you can generate an API key on the developer dashboard .

period_from	Show consumption from the given datetime (inclusive). This parameter can be provided on its own.
period_to	Show consumption to the given datetime (exclusive). This parameter also requires providing the 'period_from' parameter to create a range.
tz	a character string that specifies which time zone to parse the date with. The string must be a time zone that is recognized by the user's OS.
order_by	Ordering of results returned. Default is that results are returned in reverse order from latest available figure. Valid values: • 'period', to give results ordered forward. • '-period', (default), to give results ordered from most recent backwards.
page_size	The number of results to return per page. This is intended for internal testing and may be removed in a future release.
group_by	Aggregates consumption over a specified time period. A day is considered to start and end at midnight in the server's time zone. The default is that consumption is returned in half-hour periods. Accepted values are: • 'hour' • 'day' • 'week' • 'month' • 'quarter'

Value

a [tibble](#) of the requested consumption data.

Note

For the fastest data aggregation, it is recommended to have either the **data.table** or **vctrs** packages installed.

get_meter_gsp	<i>Get the GSP of a meter-point.</i>
---------------	--------------------------------------

Description

This endpoint can be used to get the GSP of a given meter-point.

Usage

```
get_meter_gsp(mpan = NULL)
```

Arguments

mpan	The electricity meter-point's MPAN
------	------------------------------------

Value

a character of the meter-points GSP.

get_products	<i>Return a list of energy products</i>
--------------	---

Description

By default, results will be public energy products but if authenticated organisations will also see products available to their organisation.

Usage

```
get_products(  
  is_variable = NULL,  
  is_green = NULL,  
  is_tracker = NULL,  
  is_prepay = NULL,  
  is_business = FALSE,  
  available_at = Sys.Date(),  
  authenticate = FALSE,  
  api_key = NULL  
)
```

Arguments

is_variable	(boolean, optional) Show only variable products.
is_green	(boolean, optional) Show only green products.
is_tracker	(boolean, optional) Show only tracker products.
is_prepay	(boolean, optional) Show only pre-pay products.
is_business	(boolean, default: FALSE) Show only business products.
available_at	Show products available for new agreements on the given datetime. Defaults to current datetime, effectively showing products that are currently available.
authenticate	(boolean, default: FALSE) Use an api_key to authenticate. Only useful for organisations.
api_key	Your API key. If you are an Octopus Energy customer, you can generate an API key on the developer dashboard .

Value

a [tibble](#)

Examples

```
get_products(is_green = TRUE)
```

set_api_key	<i>Set the Octopus API key</i>
-------------	--------------------------------

Description

Set the Octopus API key to use. This will be stored as an environment variable. You should add `OCTOPUSR_API_KEY = <api_key>` to your `.Renvi` otherwise you will have to call this function every session.

Usage

```
set_api_key(api_key = NULL)
```

Arguments

api_key	Your API key. If you are an Octopus Energy customer, you can generate an API key on the developer dashboard .
---------	---

Value

No return value, called for side effects.

set_meter_details	<i>Set meter details</i>
-------------------	--------------------------

Description

Store your meter details in environment variables for the session. To find your meter details, check your [Octopus Energy dashboard](#).

Usage

```
set_meter_details(
  meter_type = c("electricity", "gas"),
  mpan_mprn = NULL,
  serial_number = NULL,
  direction = NULL
)
```

Arguments

meter_type	Type of meter-point, electricity or gas
mpan_mprn	The electricity meter-point's MPAN or gas meter-point's MPRN.
serial_number	The meter's serial number.
direction	For electricity meters, specify "import", "export", or NULL (default). When NULL, uses the legacy single MPAN storage. When specified, stores separate import/export MPANs.

set_meter_details

7

Value

No return value, called for side effects.

Index

`combine_consumption`, [2](#)

`get_consumption`, [3](#)

`get_meter_gsp`, [4](#)

`get_products`, [5](#)

`lubridate`, [3](#)

`set_api_key`, [6](#)

`set_meter_details`, [6](#)

`tibble`, [3–5](#)