

Package ‘plug’

September 29, 2026

Type Package

Title Secure and Intuitive Access to 'Plug' Interface

Version 0.2.0

Date 2026-09-29

Description Provides a secure and user-friendly interface to interact with the 'Plug' <<https://plugbytpf.com.br>> 'API'. It enables developers to store and manage credentials and tokens securely using the 'keyring' package, and to retrieve data from 'API' endpoints with the 'httr2' package, using 'SQL' queries built safely from templates. Designed for simplicity and security, the package facilitates seamless integration with the 'Plug' ecosystem.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Imports glue, httr2 (>= 1.0.0), keyring, tibble

Suggests testthat (>= 3.0.0), withr

Config/testthat/edition 3

URL <https://github.com/StrategicProjects/plug>

BugReports <https://github.com/StrategicProjects/plug/issues>

Config/Needs/website tidyverse/tidytemplate

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Andre Leite [aut, cre],
Felipe Ferreira [aut],
Hugo Vasconcelos [aut],
Diogo Bezerra [aut],
Roger Azevedo [aut],
Marcos Wasiliew [aut] (ORCID: <<https://orcid.org/0009-0004-4694-3159>>),
Júlia Nascimento Barreto [aut] (ORCID:
<<https://orcid.org/0009-0004-2851-7770>>)

Maintainer Andre Leite <leite@castlab.org>

Repository CRAN
Date/Publication 2026-09-29 12:40:09 UTC

Contents

plug_clear_credentials	2
plug_download_base	3
plug_execute_query	4
plug_get_valid_token	5
plug_list_credentials	6
plug_list_tokens	6
plug_store_credentials	7
Index	8

plug_clear_credentials
<i>Remove stored credentials and tokens for Plug API</i>

Description

This function removes the username, password and cached token stored by this package from the keyring.

Usage

```
plug_clear_credentials(credentials = TRUE)
```

Arguments

credentials	If TRUE (the default), removes the stored username and password. If FALSE, only the cached token is removed.
-------------	--

Value

TRUE (invisibly) if something was removed, FALSE otherwise.

Examples

```
old <- options(keyring_backend = "env")

plug_store_credentials("myusername", "mypassword")
plug_clear_credentials()

options(old)
```

plug_download_base	<i>Download all data from a specific base</i>
--------------------	---

Description

This function downloads all data from a specified base using the query `SELECT * FROM base_name`.

Usage

```
plug_download_base(  
  base_name,  
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/executeQuery",  
  verbosity = 0,  
  auth_endpoint = "https://plug.der.pe.gov.br/MadrixApi/authenticate/"  
)
```

Arguments

base_name	The name of the base from which to download all data. It can include the schema, as in <code>"dbo.Contratos_VIEW"</code> .
endpoint	The endpoint URL for executing queries.
verbosity	The verbosity level of the API request (0 = none, 1 = minimal, 2 = detailed).
auth_endpoint	The endpoint URL for generating the token.

Value

A tibble containing all data from the specified base.

Examples

```
## Not run:  
data <- plug_download_base(  
  base_name = "Contratos_VIEW"  
)  
  
## End(Not run)
```

plug_execute_query	<i>Execute a custom SQL query on the Plug database</i>
--------------------	--

Description

This function executes a user-defined SQL query on the Plug database, with safe query construction using the same placeholder syntax as `glue::glue_sql()`.

Usage

```
plug_execute_query(
  sql_template,
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/executeQuery",
  verbosity = 0,
  ...,
  auth_endpoint = "https://plug.der.pe.gov.br/MadrixApi/authenticate/",
  .envir = parent.frame()
)
```

Arguments

<code>sql_template</code>	A SQL query template with placeholders for variables.
<code>endpoint</code>	The endpoint URL for executing queries.
<code>verbosity</code>	The verbosity level of the API request (0 = none, 1 = minimal, 2 = detailed).
<code>...</code>	Named arguments to replace placeholders in the SQL template.
<code>auth_endpoint</code>	The endpoint URL for generating the token.
<code>.envir</code>	The environment in which placeholders not supplied in <code>...</code> are evaluated.

Details

Values are inserted in the query as Microsoft SQL Server literals:

- `{x}` inserts a quoted value: strings are wrapped in single quotes (with embedded quotes escaped), dates and date-times are formatted, logical values become `1/0` and `NA` becomes `NULL`.
- `{x*}` inserts all the elements of `x` separated by commas, for use with `IN (...)`.
- `{`x`}` inserts a quoted identifier, such as a table or column name.
- `{I(x)}` inserts `x` as is, without any quoting.

Use double braces (`{{` and `}}`) to insert literal braces in the query.

If the API rejects the cached token, a new token is generated and the query is executed again.

Value

A tibble containing the query results.

Examples

```
## Not run:
data <- plug_execute_query(sql_template = "SELECT TOP 1 * FROM Contratos_VIEW")

# Using placeholders
data <- plug_execute_query(
  "SELECT TOP {n} * FROM {`base`} WHERE Ano IN ({years*})",
  n = 10,
  base = "Contratos_VIEW",
  years = c(2023, 2024)
)

## End(Not run)
```

plug_get_valid_token *Get a valid token for Plug API*

Description

This function checks if a valid global token exists for the Plug API. If no valid token is found, it generates a new token using the stored global credentials by sending a properly formatted request. If it fails to retrieve the token (for example, due to missing credentials or network issues), it will not throw an error but will display a message explaining the problem and return NULL.

Usage

```
plug_get_valid_token(
  validity_time = 3600,
  endpoint = "https://plug.der.pe.gov.br/MadrixApi/authenticate/",
  force = FALSE
)
```

Arguments

validity_time	The validity period of the token in seconds. Default is 3600 (1 hour).
endpoint	The endpoint URL for generating the token.
force	If TRUE, ignores the cached token and always requests a new one.

Value

The valid token as a string, or NULL if no valid credentials were found or an error occurred.

Examples

```
## Not run:
token <- plug_get_valid_token(validity_time = 3600)

## End(Not run)
```

plug_list_credentials *List registered credentials for Plug API*

Description

This function lists all globally stored credentials (username and password) for the Plug API. If none are found or an error occurs, it displays a message in English ("No credentials found for Plug API.") and returns an empty list.

Usage

```
plug_list_credentials()
```

Details

Note that the password is returned in plain text.

Value

A named list with username and password fields if credentials are found, or an empty list if no credentials are stored.

Examples

```
## Not run:
plug_list_credentials()

## End(Not run)
```

plug_list_tokens *List registered tokens for Plug API*

Description

This function lists the stored API token and its expiration time for the Plug API. If none are found or an error occurs, it displays a message in English ("No token found for Plug API.") and returns an empty list.

Usage

```
plug_list_tokens()
```

Value

A named list with token and expiration fields if a token is found, or an empty list if no token is stored.

Examples

```
## Not run:
plug_list_tokens()

## End(Not run)
```

plug_store_credentials

Store user credentials securely for Plug API

Description

This function securely stores the global username and password required to authenticate with the Plug application. Any cached token is discarded, so the next request authenticates with the new credentials.

Usage

```
plug_store_credentials(username, password)
```

Arguments

username	The username for the Plug.
password	The password for the Plug.

Value

No return value. The credentials are securely stored.

Examples

```
# Use a temporary, in-memory keyring so the example does not touch
# the credentials stored in the system keyring
old <- options(keyring_backend = "env")

plug_store_credentials("myusername", "mypassword")
plug_clear_credentials()

options(old)
```

Index

`glue::glue_sql()`, [4](#)

`plug_clear_credentials`, [2](#)

`plug_download_base`, [3](#)

`plug_execute_query`, [4](#)

`plug_get_valid_token`, [5](#)

`plug_list_credentials`, [6](#)

`plug_list_tokens`, [6](#)

`plug_store_credentials`, [7](#)