

Package: distilleR (via r-universe)

May 25, 2026

Type Package

Title A Wrap Around the 'DistillerSR' APIs

Version 1.0.0

Maintainer Luca Belmonte <luca.belmonte@efsa.europa.eu>

Description Interface to 'DistillerSR' APIs. See
<<https://apidocs.evidencepartners.com/>> for more details.

License EUPL-1.2

URL <https://openefsa.github.io/distilleR/>

BugReports <https://github.com/openefsa/distilleR/issues>

Depends R (>= 4.1.0)

Imports cli (>= 3.6.5), checkmate (>= 2.3.1), glue (>= 1.7.0), httr2 (>= 1.2.1), jsonlite (>= 1.8.7), readr (>= 2.1.5), readxl (>= 1.4.3), tibble (>= 3.3.0)

Suggests devtools (>= 2.4.5), knitr (>= 1.0), rmarkdown (>= 2.0), roxygen2 (>= 7.2.1), testthat (>= 3.0.0), usethis (>= 2.2.3), covr (>= 3.6.4), openxlsx (>= 4.2.8), rlang (>= 1.1.4)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Config/testthat/edition 3

VignetteBuilder knitr

Config/Needs/website pkgdown

Roxygen list(markdown = TRUE)

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

Date/Publication 2026-04-22 08:48:39 UTC

RemoteUrl <https://github.com/openefsa/distiller>

RemoteRef HEAD

RemoteSha 24adb0259379b43208a70537d06e7b907f595f5f

Contents

getAuthenticationToken	2
getProjects	3
getReport	4
getReports	5
Index	7

getAuthenticationToken

Authenticate to a DistillerSR session.

Description

Authenticates a user to a DistillerSR instance using a personal access key. The function returns a valid authentication token that can be used to access protected DistillerSR API endpoints.

Usage

```
getAuthenticationToken(
    distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),
    distillerKey = Sys.getenv("DISTILLER_API_KEY"),
    timeout = 1800
)
```

Arguments

<code>distillerInstanceUrl</code>	character (string). The base URL of the DistillerSR instance. By default: <code>Sys.getenv("DISTILLER_INSTANCE_URL")</code> .
<code>distillerKey</code>	character (string). The personal access key generated in DistillerSR. By default: <code>Sys.getenv("DISTILLER_API_KEY")</code> .
<code>timeout</code>	integer. The maximum number of seconds to wait for the authentication response. By default: 1800 seconds (30 minutes).

Details

By default, the personal access key and the instance URL are read from the environment variables `DISTILLER_API_KEY` and `DISTILLER_INSTANCE_URL`.

Value

A string containing a valid DistillerSR authentication token.

Examples

```
## Not run:
# If 'DISTILLER_INSTANCE_URL' and 'DISTILLER_API_KEY' are defined in your
# environment (e.g. .Renvirom).
distillerToken_ <- getAuthenticationToken()

# If 'distillerInstanceUrl' and 'distillerKey' are to be specified manually.
distillerToken_ <- getAuthenticationToken(
  distillerInstanceUrl = "https://url.to.distiller.instance",
  distillerKey = "YOUR_API_KEY")

## End(Not run)
```

getProjects	<i>Get the list of the Distiller projects associated to the authenticated user.</i>
-------------	---

Description

This function queries the DistillerSR API to retrieve the list of projects accessible to the authenticated user. It requires an authentication token and a valid API instance URL. The result is a dataframe listing available projects.

Usage

```
getProjects(
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),
  distillerToken,
  timeout = 1800
)
```

Arguments

distillerInstanceUrl	character (string). The distiller instance URL. By default: Sys.getenv("DISTILLER_INSTANCE_URL").
distillerToken	character (string). The token the user gets once authenticated.
timeout	integer. The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).

Value

A tibble with four columns:

- id: The project ID.
- name: The name of the project.
- de_project_id.
- is_hidden.

See Also

[getAuthenticationToken](#)

Examples

```
## Not run:
distillerToken_ <- getAuthenticationToken()

projects_ <- getProjects(distillerToken = distillerToken_)

## End(Not run)
```

getReport

Get a Distiller report associated to a project of the authenticated user.

Description

This function queries the DistillerSR API to retrieve a saved report associated with a given project ID. It requires user authentication and a valid API endpoint URL. The result is a dataframe containing metadata about the saved report.

Usage

```
getReport(
  projectId,
  reportId,
  format = c("excel", "csv"),
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),
  distillerToken,
  timeout = 1800,
  attempts = 1,
  retryEach = 600,
  verbose = TRUE
)
```

Arguments

projectId integer. The ID of the project as provided by DistillerSR.

reportId integer. The ID of the report as provided by DistillerSR.

format character (string). The desired format for the document. It can be either excel or csv.

distillerInstanceUrl character (string). The distiller instance URL.
By default: `Sys.getenv("DISTILLER_INSTANCE_URL")`.

distillerToken character (string). The token the user gets once authenticated.

timeout	integer. The maximum number of seconds to wait for the response. By default: 1800 seconds (30 minutes).
attempts	integer. The maximum number of attempts. By default: 1 attempt.
retryEach	integer. The delay between attempts. By default: 600 seconds (10 minutes).
verbose	logical. A flag to specify whether to make the function verbose or not. By default: TRUE.

Value

A data frame containing the Distiller report as designed within DistillerSR.

See Also

[getAuthenticationToken](#)
[getProjects](#)
[getReports](#)

Examples

```
## Not run:  
distillerToken_ <- getAuthenticationToken()  
  
projects_ <- getProjects(distillerToken = distillerToken_)  
  
reports_ <- getReports(  
  projectId = projects_$id[1],  
  distillerToken = distillerToken_)  
  
report_ <- getReport(  
  projectId = projects_$id[1],  
  reportID = reports_$id[7],  
  format = "csv",  
  distillerToken = distillerToken_)  
  
## End(Not run)
```

getReports	<i>Get the list of the Distiller reports associated to a project of the authenticated user.</i>
------------	---

Description

This function queries the DistillerSR API to retrieve the list of saved reports associated with a given project ID. It requires user authentication and a valid API endpoint URL. The result is a dataframe containing metadata about each saved report.

Usage

```
getReports(  
  projectId,  
  distillerInstanceUrl = Sys.getenv("DISTILLER_INSTANCE_URL"),  
  distillerToken,  
  timeout = 1800  
)
```

Arguments

`projectId` integer. The ID of the project as provided by DistillerSR.

`distillerInstanceUrl` character (string). The distiller instance URL.
By default: `Sys.getenv("DISTILLER_INSTANCE_URL")`.

`distillerToken` character (string). The token the user gets once authenticated.

`timeout` integer. The maximum number of seconds to wait for the response.
By default: 1800 seconds (30 minutes).

Value

A tibble with four columns:

- `id`: The ID of the saved report.
- `name`: The name of the report.
- `date`: The creation date of the report.
- `view`: The format of the report (e.g., html, csv, excel).

See Also

[getAuthenticationToken](#)

[getProjects](#)

Examples

```
## Not run:  
distillerToken_ <- getAuthenticationToken()  
  
projects_ <- getProjects(distillerToken = distillerToken_)  
  
reports_ <- getReports(  
  projectId = projects_$id[1],  
  distillerToken = distillerToken_)  
  
## End(Not run)
```

Index

getAuthenticationToken, [2](#), [4-6](#)
getProjects, [3](#), [5](#), [6](#)
getReport, [4](#)
getReports, [5](#), [5](#)