

Package ‘valdr’

July 24, 2025

Title Access and Analyse 'VALD' Data via Our External 'APIs'

Version 1.0.1

Description Provides helper functions and wrappers to simplify authentication, data retrieval, and result processing from the 'VALD' 'APIs'.

Designed to streamline integration for analysts and researchers working with 'VALD's external 'APIs'.

For further documentation on integrating with 'VALD' 'APIs', see:

<<https://support.vald.com/hc/en-au/articles/23415335574553-How-to-integrate-with-VALD-APIs>>.

For a step-by-step guide to using this package, see:

<<https://support.vald.com/hc/en-au/articles/48730811824281-A-guide-to-using-the-valdr-R-package>>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Imports httr, base64enc, keyring, jsonlite

NeedsCompilation no

Author Kieran Harrison [aut, cre],
VALD Support [ctb],
VALD [cph] (Copyright holder)

Maintainer Kieran Harrison <k.harrison@vald.com>

Repository CRAN

Date/Publication 2025-07-24 13:30:02 UTC

Contents

.retry_key_get	2
authenticate	3
get_access_token	3
get_config	4
get_forcedecks_data	4
get_forcedecks_result_definitions	5

get_forcedecks_result_definitions_only	5
get_forcedecks_tests	6
get_forcedecks_tests_only	6
get_forcedecks_tests_trials	7
get_forcedecks_trials	8
get_forcedecks_trials_only	8
get_profiles	9
get_profiles_only	9
get_start_date	10
load_credentials	10
set_credentials	11
set_start_date	11
Index	12

.retry_key_get	<i>Internal helper function to retry reading a key from the system keyring. This handles potential intermittent delays after writing to the keyring.</i>
----------------	--

Description

Internal helper function to retry reading a key from the system keyring. This handles potential intermittent delays after writing to the keyring.

Usage

```
.retry_key_get(service, username, retries = 5, delay = 0.5)
```

Arguments

- service Character. Keyring service name (e.g., "valdr_credentials").
- username Character. Keyring username (e.g., "client_id").
- retries Integer. Number of retry attempts before giving up (default 5).
- delay Numeric. Delay in seconds between retry attempts (default 0.5).

Value

Character scalar. The retrieved credential value. Will throw an error if unable to retrieve a valid, non-blank value after all retries.

authenticate	<i>Authenticate and retrieve a valid access token</i>
--------------	---

Description

Ensures a valid access token is available based on stored credentials. Also validates required tenant ID and token presence for subsequent function calls.

Usage

```
authenticate()
```

Value

A character vector of length 1 (a single string) representing a valid access token. Returned invisibly and used internally for authentication.

get_access_token	<i>Get or refresh VALD access token</i>
------------------	---

Description

Retrieves a cached access token from disk if valid, otherwise fetches a new one.

Usage

```
get_access_token(config = NULL, verbose = TRUE)
```

Arguments

config	Configuration object. If NULL, uses internal config set by <code>set_credentials()</code> .
verbose	Whether to print a message when refreshing the token.

Value

(Invisibly) a character vector of length 1 containing the bearer token used for authenticating API requests.

get_config	<i>Retrieve stored VALD configuration</i>
------------	---

Description

Returns the configuration list previously set by set_credentials(). If credentials have not been set, this function will raise an error.

Usage

```
get_config(safe = TRUE, quiet = FALSE)
```

Arguments

safe	Logical; if TRUE (default), sensitive values are redacted in printed output (only when not quiet).
quiet	Logical; if TRUE, suppresses printed output (default FALSE).

Value

A named list containing the stored VALD configuration values for the current user. Sensitive values are redacted when printed with safe = TRUE. Returned invisibly.

get_forcedecks_data	<i>Run full initial data fetch from the VALD ForceDecks API and External Profiles API</i>
---------------------	---

Description

This function is intended for first-time use or a full refresh of all datasets. It retrieves profiles, result definitions, tests (from a specified start date), and trials.

Usage

```
get_forcedecks_data(start_date = NULL)
```

Arguments

start_date	In ISO 8601 UTC format (e.g., "2025-06-25T00:00:00Z") indicating the start of the test retrieval window.
------------	--

Value

A named list with data frames: profiles, result_definitions, tests, and trials. Each element contains the respective dataset fetched from the ForceDecks API.

Examples

```
## Not run:  
# Fetch all data (profiles, results, tests, trials)  
data <- get_forcedecks_data()  
View(data$profiles)  
  
## End(Not run)
```

```
get_forcedecks_result_definitions  
  Get ForceDecks result definitions
```

Description

Retrieves all available result definitions from the ForceDecks API.

Usage

```
get_forcedecks_result_definitions()
```

Value

A data frame where each row corresponds to a ForceDecks result definition retrieved from the API. Returned invisibly.

```
get_forcedecks_result_definitions_only  
  Get only ForceDecks result definitions
```

Description

Wrapper around `get_forcedecks_result_definitions()` to refresh result definitions. Typically called infrequently unless definitions are known to have changed.

Usage

```
get_forcedecks_result_definitions_only()
```

Value

A data frame where each row corresponds to a ForceDecks result definition retrieved from the API. Returned invisibly.

Examples

```
## Not run:
# Fetch result definitions
results <- get_forcedecks_result_definitions_only()
View(results)

## End(Not run)
```

```
get_forcedecks_tests  Get ForceDecks tests
```

Description

Retrieves ForceDecks test data with optional filtering by start date and profile ID.

Usage

```
get_forcedecks_tests(start_date = NULL, profile_id = NULL)
```

Arguments

start_date	Optional ISO 8601 UTC date string (e.g., "2025-06-25T00:00:00Z").
profile_id	Optional Profile ID to filter results.

Value

A data frame containing ForceDecks test results matching the optional filters. If no tests are found, returns an empty data frame. Returned invisibly.

```
get_forcedecks_tests_only
Get only ForceDecks test data
```

Description

Wrapper around get_forcedecks_tests() if the user only wants test-level data without trials.

Usage

```
get_forcedecks_tests_only(start_date = NULL)
```

Arguments

start_date	In ISO 8601 UTC format (e.g., "2025-06-25T00:00:00Z") indicating the start of the test retrieval window.
------------	--

Value

A data frame containing ForceDecks test-level data.

Examples

```
## Not run:  
# Fetch only tests  
tests <- get_forcedecks_tests_only()  
View(tests)  
  
## End(Not run)
```

```
get_forcedecks_tests_trials
```

Run a standard session to get new ForceDecks tests and trials only

Description

Use this when profiles and result definitions have already been downloaded previously.

Usage

```
get_forcedecks_tests_trials(start_date = NULL)
```

Arguments

start_date	In ISO 8601 UTC format (e.g., "2025-06-25T00:00:00Z") indicating the start of the test retrieval window.
------------	--

Value

A named list with two data frames: `tests` and `trials`. These contain newly retrieved ForceDecks tests and their associated trial results.

Examples

```
## Not run:  
# Fetch tests and trials only  
session <- get_forcedecks_tests_trials()  
View(session$tests)  
View(session$trials)  
  
## End(Not run)
```

get_forcedecks_trials *Get trial results for a set of ForceDecks tests*

Description

Retrieves and flattens trial-level result data for each test provided.

Usage

```
get_forcedecks_trials(tests_df)
```

Arguments

tests_df A data frame of tests (as returned by get_forcedecks_tests())

Value

A data frame containing flattened trial-level results for the supplied tests. If no trial results are found, returns an empty data frame.

get_forcedecks_trials_only
 Get trials for an existing test data frame

Description

Useful for re-running or extending analysis without re-downloading tests.

Usage

```
get_forcedecks_trials_only(tests_df)
```

Arguments

tests_df A data frame of tests (from get_forcedecks_tests() or a previous session)

Value

A data frame containing trial-level results corresponding to the input tests.

Examples

```
## Not run:
# Fetch trials based on existing tests
tests <- get_forcedecks_tests_only()
trials <- get_forcedecks_trials_only(tests)
View(trials)

## End(Not run)
```

get_profiles	<i>Get VALD profiles</i>
--------------	--------------------------

Description

Queries the External Profiles API and returns a data frame of VALD profiles.

Usage

```
get_profiles()
```

Value

A data frame containing VALD profiles information for the stored tenant_id. If no profiles are found, the function raises an error.

get_profiles_only	<i>Get only VALD profiles</i>
-------------------	-------------------------------

Description

Wrapper around get_profiles() to retrieve VALD profiles. Useful if profile data needs to be refreshed independently.

Usage

```
get_profiles_only()
```

Value

A data frame containing VALD profiles information for the stored tenant_id.

Examples

```
## Not run:
# Fetch profiles
profiles <- get_profiles_only()
View(profiles)

## End(Not run)
```

get_start_date	<i>Retrieve the start date from config</i>
----------------	--

Description

Returns the saved start date string from the config file.

Usage

```
get_start_date()
```

Value

A character scalar containing the start date string in ISO 8601 format previously saved to configuration.

load_credentials	<i>Load Stored VALD API Credentials and Configuration (with retry logic)</i>
------------------	--

Description

Loads the saved VALD API configuration from the user's config file and retrieves sensitive credentials securely from the system keyring.

Usage

```
load_credentials(verbose = TRUE)
```

Arguments

verbose	Logical; if TRUE, prints messages on load status (default FALSE).
---------	---

Value

Invisibly returns TRUE if credentials and configuration were loaded successfully, FALSE otherwise.

set_credentials	<i>Set and Save VALD API Credentials</i>
-----------------	--

Description

Securely stores VALD API credentials in the system keyring and saves non-sensitive configuration to a JSON config file in the user's home directory for reuse across sessions.

Usage

```
set_credentials(client_id, client_secret, tenant_id, region)
```

Arguments

client_id	Your VALD API Client ID (stored securely in keyring)
client_secret	Your VALD API Client Secret (stored securely in keyring)
tenant_id	Your VALD Tenant ID
region	The VALD data region code (e.g., "aue", "use", "euw")

Details

Sensitive values (client ID and secret) are never written to disk and are retrieved securely from the keyring when needed.

Value

Invisibly returns TRUE if credentials and configuration were saved successfully.

set_start_date	<i>Set and persist the start date</i>
----------------	---------------------------------------

Description

Saves the provided ISO 8601 start date to the config file.

Usage

```
set_start_date(start_date)
```

Arguments

start_date	Date in ISO 8601 UTC format, e.g., "2025-06-25T00:00:00Z"
------------	---

Value

A logical scalar (TRUE), returned invisibly, indicating that the start date was saved and persisted successfully.

Index

`.retry_key_get`, [2](#)

`authenticate`, [3](#)

`get_access_token`, [3](#)

`get_config`, [4](#)

`get_forcedecks_data`, [4](#)

`get_forcedecks_result_definitions`, [5](#)

`get_forcedecks_result_definitions_only`,
[5](#)

`get_forcedecks_tests`, [6](#)

`get_forcedecks_tests_only`, [6](#)

`get_forcedecks_tests_trials`, [7](#)

`get_forcedecks_trials`, [8](#)

`get_forcedecks_trials_only`, [8](#)

`get_profiles`, [9](#)

`get_profiles_only`, [9](#)

`get_start_date`, [10](#)

`load_credentials`, [10](#)

`set_credentials`, [11](#)

`set_start_date`, [11](#)