Package 'ntfy'

November 6, 2024

Title Lightweight Wrapper to the 'ntfy.sh' Service

Version 0.0.6

Maintainer Jonathan Carroll <rpkg@jcarroll.com.au>

Description The 'ntfy' (pronounce: notify) service is a simple HTTP-based pub-sub
notification service. It allows you to send notifications to your phone or
desktop via scripts from any computer, entirely without signup, cost or
setup. It's also open source if you want to run your own.
Visit <https://ntfy.sh> for more details.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Imports httr2, jsonlite

URL https://github.com/jonocarroll/ntfy

BugReports https://github.com/jonocarroll/ntfy/issues

Depends R (>= 4.1)

LazyData true

Suggests ggplot2, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Jonathan Carroll [aut, cre] (<https://orcid.org/0000-0002-1404-5264>),
Andrew Heiss [ctb] (<https://orcid.org/0000-0002-3948-3914>)

Repository CRAN

Date/Publication 2024-11-06 15:20:02 UTC

Contents

1

---

emoji	*Emoji*

Description

Emoji symbols available in ntfy

Usage

```
emoji
```

Format

`emoji`:

A data frame with 7,240 rows and 60 columns:

emoji Symbol

aliases Name of the emoji that can be used in a `tag`

tags Alternative names

category Category

description Description

unicode_version Unicode Version ...

Source

https://github.com/binwiederhier/ntfy/blob/main/web/src/app/emojis.js

---

ntfy_auth   *Get the ntfy authorization indicator*

Description

Get the ntfy authorization indicator

Usage

```
ntfy_auth(var = "NTFY_AUTH")
```

Arguments

var   environment variable in which the ntfy authorization indicator is stored

Value

a logical that indicates if password authorization is used

---

ntfy_done   *Notify Completion of a Process*

Description

Notify Completion of a Process

Usage

```
ntfy_done(
  x,
  message = paste0("Process completed at ", Sys.time()),
  title = "ntfy_done()",
  tags = "white_check_mark",
  topic = ntfy_topic(),
  server = ntfy_server(),
  auth = ntfy_auth(),
  username = ntfy_username(),
  password = ntfy_password(),
  ...
)
```

Arguments

`x`	a result (ignored)
`message`	text to send as notification
`title`	title of notification. See https://docs.ntfy.sh/publish/#message-title
`tags`	text tags or emoji shortcodes from https://docs.ntfy.sh/emojis/, provided as a list
`topic`	subscribed topic to which to send notification
`server`	ntfy server
`auth`	logical indicating if the topic requires password authorization
`username`	username with access to a protected topic.
`password`	password with access to a protected topic.
`...`	other arguments passed to `ntfy_send()`

Value

the input x (for further piping) plus a notification will be sent

Examples

```
# report that a process has completed
Sys.sleep(3) |> ntfy_done("Woke up")
```

ntfy_done_with_timing   *Notify Completion of a Process with Timing*

Description

Notify Completion of a Process with Timing

Usage

```
ntfy_done_with_timing(
  x,
  message = paste0("Process completed in ", time_result, "s"),
  title = "ntfy_done_with_timing()",
  tags = "stopwatch",
  topic = ntfy_topic(),
  server = ntfy_server(),
  auth = ntfy_auth(),
  username = ntfy_username(),
  password = ntfy_password(),
  ...
)
```

Arguments

x	expression to be evaluated and timed
message	text to send as notification
title	title of notification. See https://docs.ntfy.sh/publish/#message-title
tags	text tags or emoji shortcodes from https://docs.ntfy.sh/emojis/, provided as a list
topic	subscribed topic to which to send notification
server	ntfy server
auth	logical indicating if the topic requires password authorization
username	username with access to a protected topic.
password	password with access to a protected topic.
...	other arguments passed to ntfy_send()

Value

the result of evaluating x (for further piping) plus a notification will be sent

Examples

```
# report that a process has completed, and how long it took
Sys.sleep(3) |> ntfy_done_with_timing()
```

---

ntfy_history   *Retrieve History of Notifications*

Description

Retrieve History of Notifications

Usage

```
ntfy_history(
  since = "all",
  topic = ntfy_topic(),
  server = ntfy_server(),
  auth = ntfy_auth(),
  username = ntfy_username(),
  password = ntfy_password(),
  ...
)
```

Arguments

since	duration (e.g. ″10m″ or ″30s″), a Unix timestamp (e.g. ″1635528757″), a message ID (e.g. ″nFS3knfcQ1xe″), or ″all″ (all cached messages)
topic	subscribed topic to which to send notification
server	ntfy server
auth	logical indicating if the topic requires password authorization
username	username with access to a protected topic.
password	password with access to a protected topic.
...	any other (named) query parameters to add to the request

Value

a data.frame() with one row per notification, with columns as described in the documentation

See Also

https://ntfy.sh/docs/subscribe/api/#json-message-format

Examples

```
# get the last hour of notifications
ntfy_history(since = ″1h″)
```

---

ntfy_password   *Get the ntfy password*

Description

Get the ntfy password

Usage

```
ntfy_password(var = ″NTFY_PASSWORD″)
```

Arguments

var	environment variable in which the password is stored

Value

the password for the username with access to the protected ntfy topic

---

`ntfy_send`   *Send a Notification*

Description

Send a Notification

Usage

```
ntfy_send(
  message = "test",
  title = NULL,
  tags = NULL,
  priority = 3,
  actions = NULL,
  click = NULL,
  image = NULL,
  attach = NULL,
  filename = NULL,
  delay = NULL,
  email = NULL,
  topic = ntfy_topic(),
  server = ntfy_server(),
  auth = ntfy_auth(),
  username = ntfy_username(),
  password = ntfy_password()
)
```

Arguments

`message`	text to send as notification
`title`	title of notification. See https://docs.ntfy.sh/publish/#message-title
`tags`	text tags or emoji shortcodes from https://docs.ntfy.sh/emojis/, provided as a list
`priority`	Message priority with 1=min, 3=default and 5=max. See https://docs.ntfy.sh/publish/#message-priority
`actions`	Custom user action buttons for notifications. See https://docs.ntfy.sh/publish/#action-buttons
`click`	Website opened when notification is clicked. See https://docs.ntfy.sh/publish/#click-action
`image`	Image to include in the body of the notification. Either a `ggplot` object or a filename.
`attach`	URL of an attachment, see attach via URL. See https://docs.ntfy.sh/publish/#attach-file-from-url
`filename`	File name of the attachment

delay	Timestamp or duration for delayed delivery
email	E-mail address for e-mail notifications??
topic	subscribed topic to which to send notification
server	ntfy server
auth	logical indicating if the topic requires password authorization
username	username with access to a protected topic.
password	password with access to a protected topic.

Value

a [httr2::response()](#) object

Examples

```
# send a message to the default topic ('mytopic')
ntfy_send("test from R!")

# can use tags (emoji)
ntfy_send(message = "sending with tags!",
          tags = c(tags$cat, tags$dog)
)
```

---

ntfy_server   *Get the ntfy server*

Description

Get the ntfy server

Usage

```
ntfy_server(var = "NTFY_SERVER")
```

Arguments

var	environment variable in which the server URL is stored

Value

the ntfy server URL

---

ntfy_topic *Get the ntfy topic*

Description

Get the ntfy topic

Usage

```
ntfy_topic(var = "NTFY_TOPIC")
```

Arguments

var   environment variable in which the topic is stored

Value

the ntfy topic to which the user should be subscribed

---

ntfy_username *Get the ntfy username*

Description

Get the ntfy username

Usage

```
ntfy_username(var = "NTFY_USERNAME")
```

Arguments

var   environment variable in which the username is stored

Value

the username with access to the protected ntfy topic

show_emoji	*Show an emoji symbol, or find one by name*

Description

Show an emoji symbol, or find one by name

Usage

```
show_emoji(name = NULL, search = FALSE)
```

Arguments

name	name of emoji to either print or find
search	search the `tags` column for this name?

Details

Emoji are loaded with `data("emoji")` and the `aliases` column contains the names compatible with ntfy. Alternative names are included in the `tags` column and these will be searched if the name is not found in `aliases`.

Value

nothing, just prints the emoji if one or more are found

Examples

```
show_emoji("dog")
show_emoji("party")
```

tags	*Emoji tags*

Description

Emoji symbols compatible with ntfy to be used as tags

Usage

```
tags
```

Format

An object of class `list` of length 1855.

Index