

Package: ntfy (via r-universe)

October 10, 2026

Title Lightweight Wrapper to the 'ntfy.sh' Service

Version 0.1.1

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

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports httr2, jsonlite

URL <https://github.com/jonocarroll/ntfy>,
<https://jonocarroll.github.io/ntfy/>

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

Depends R (>= 4.1)

LazyData true

Suggests ggplot2, testthat (>= 3.0.0), withr

Config/testthat/edition 3

Config/pak/sysreqs libssl-dev

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

Date/Publication 2025-12-13 05:55:07 UTC

RemoteUrl <https://github.com/jonocarroll/ntfy>

RemoteRef HEAD

RemoteSha dbb10bea304bdedd07313f76c87d3219bb65862a

Contents

emoji	2
ntfy_done	3
ntfy_history	4
ntfy_send	5
show_emoji	7
tags	8
Index	9

emoji	<i>Emoji</i>
-------	--------------

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_done

Notify on Completion of a Process

Description

ntfy_done() tells you when the code completed, and ntfy_done_with_timing() tells you how long it took.

Usage

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

ntfy_done_with_timing(
    x,
    message = paste0("Process completed in ", format(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	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 (when not using https://ntfy.sh)

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 <code>ntfy_send()</code>

Value

The input `x` (for further piping). A notification will be sent as a side-effect.

Examples

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

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

ntfy_history	<i>Retrieve History of Notifications</i>
--------------	--

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 (when not using https://ntfy.sh)
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_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,
  markdown = ntfy_md(),
  topic = ntfy_topic(),
  server = ntfy_server(),
  auth = ntfy_auth(),
  username = ntfy_username(),
  password = ntfy_password()
)
```

Arguments

<code>message</code>	Text to send as notification
<code>title</code>	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
markdown	(logical) use markdown formatting (TRUE or FALSE). May not be supported on all platforms
topic	Subscribed topic to which to send notification
server	ntfy server (when not using https://ntfy.sh)
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

Details

topic, server, auth, username, password, and markdown can be set via environment variables NTFY_TOPIC, NTFY_SERVER, NTFY_AUTH, NTFY_USERNAME, NTFY_PASSWORD, and NTFY_MD, respectively.

Value

a `httr2::response()` object, invisibly.

See Also

`ntfy_topic()`

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)
)
```

```
# if supported, can use markdown
ntfy_send(message = "**Important**: see _details_ [here](https://example.com)")
```

show_emoji	<i>Show an emoji symbol, or find one by name</i>
------------	--

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	<i>Emoji tags</i>
------	-------------------

Description

Emoji symbols compatible with ntfy to be used as tags

Usage

tags

Format

An object of class list of length 1855.

Index

* datasets

emoji, [2](#)

tags, [8](#)

data.frame(), [5](#)

emoji, [2](#)

httr2::response(), [6](#)

ntfy_done, [3](#)

ntfy_done_with_timing (ntfy_done), [3](#)

ntfy_history, [4](#)

ntfy_send, [5](#)

ntfy_send(), [4](#)

ntfy_topic(), [6](#)

show_emoji, [7](#)

tags, [8](#)