

Package ‘httping’

May 8, 2026

Title 'Ping' 'URLs' to Time 'Requests'

Description A suite of functions to ping 'URLs' and to time 'HTTP' 'requests'. Designed to work with 'httr'.

Version 0.3.0

License MIT + file LICENSE

URL <https://github.com/sckott/httping>

BugReports <https://github.com/sckott/httping/issues>

Encoding UTF-8

Language en-US

Imports httr (>= 1.3.1), methods, stats, jsonlite, magrittr, httpcode (>= 0.2.0), lobster

Suggests testthat, webfakes

RoxygenNote 7.3.2

NeedsCompilation no

Author Scott Chamberlain [aut, cre]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2025-12-18 23:30:02 UTC

Contents

ping	2
time	3
Index	4

ping

Ping a url, doing a single call, with any http verbs

Description

Ping a url, doing a single call, with any http verbs

Usage

```
ping(url, verb = GET, ...)
```

Arguments

url	A url
verb	(character) An http verb, default: <code>httr::GET()</code>
...	Any httr verb parameters passed on to those functions

Examples

```
## Not run:
"https://mockbin.com/request" %>% ping()
ping("https://mockbin.com/request")
"https://mockbin.com/request" %>% ping(config=verbose())
ping("https://mockbin.com/request", config=verbose())
"https://mockbin.com/request" %>% ping(config=c(verbose(), accept_json()))

"https://mockbin.com/request" %>% ping()
"https://mockbin.com/request" %>% ping(verb=HEAD)
"https://mockbin.com/request" %>% ping(verb=PUT)
"https://google.com" %>% ping()

# pass just a port number, tries to resolve, fails if not found
# "9200" %>% ping()
# 9200 %>% ping()
# 9200 %>% ping(verb=POST)
# 9200 %>% ping(verb=HEAD)
# ping(9200)
# ping("9200")

## End(Not run)
```

time *Ping a url to time the request*

Description

Ping a url to time the request

Usage

```
time(.request, count = 10, delay = 0.5, flood = FALSE, verbose = TRUE, ...)
```

Arguments

<code>.request</code>	A httr response object
<code>count</code>	integer, Number of requests to do.
<code>delay</code>	integer, Seconds to delay successive calls by. Default: 0.5 seconds.
<code>flood</code>	logical; If TRUE, no delay between requests. If FALSE, delay by 0.5 second.
<code>verbose</code>	logical; If TRUE, print progress.
<code>...</code>	Further args passed on to functions in httr

Examples

```
## Not run:  
GET("https://mockbin.com/request") %>% time()  
GET("https://api.github.com") %>% time()  
GET("http://google.com") %>% time()  
  
## End(Not run)
```

Index

`httr::GET()`, 2

`ping`, 2

`time`, 3