

Package: firestorm (via r-universe)

June 5, 2026

Title Reverse Proxy and Load Balancing for 'fiery'
Version 0.1.0.9000
Description Provides plugins for setting up 'fiery' apps as a reverse proxy. This allows you to use a 'fiery' server as a front for multiple services or even work as a load-balancer.
License MIT + file LICENSE
Encoding UTF-8
Roxygen list(markdown = TRUE)
RoxygenNote 7.3.2
Imports cli, R6, rlang, websocket
URL <https://github.com/thomasp85/firestorm>
BugReports <https://github.com/thomasp85/firestorm/issues>
Suggests fiery (>= 1.3.0), httpuv, later, reqres (>= 1.0.0), testthat (>= 3.0.0)
Config/testthat/edition 3
Config/pak/sysreqs make libssl-dev
Repository <https://thomasp85.r-universe.dev>
Date/Publication 2025-09-01 17:13:50 UTC
RemoteUrl <https://github.com/thomasp85/firestorm>
RemoteRef HEAD
RemoteSha 5175f204c874cdbc22f49b059a2641f96329130b

Contents

ReverseProxy	2
Index	4

ReverseProxy

Fiery plugin for a single reverse proxy

Description

Fiery plugin for a single reverse proxy

Fiery plugin for a single reverse proxy

Details

This plugin establishes a reverse proxy between the server and a target origin server. It supports both HTTP and WebSocket connections and sets all the relevant headers for tracking the request along the proxy chain.

Initialization

A new 'ReverseProxy'-object is initialized using the `new()` method on the generator:

Usage

```
route <- ReverseProxy$new(target, root = "/", continue = FALSE, trust = FALSE)
```

Active bindings

`name` The name used by default for the plugin

Methods**Public methods:**

- [ReverseProxy\\$new\(\)](#)
- [ReverseProxy\\$on_attach\(\)](#)
- [ReverseProxy\\$clone\(\)](#)

Method `new()`: Create a new reverse proxy

Usage:

```
ReverseProxy$new(target, root = "/", except = NULL, trust = FALSE)
```

Arguments:

`target` The URL to the origin server being proxied

`root` The root path the reverse proxy should respond to. Only requests to subsets of the root path will be proxied, and the root will be stripped from the URL path before being forwarded

`except` Subpaths to root that should be exempt from forwarding to the target.

`trust` Are requests coming from a trusted source

Method `on_attach()`: Hook for attaching the plugin to a fiery app. Should not be called directly

Usage:

```
ReverseProxy$on_attach(app, ...)
```

Arguments:

app The fiery app to attach to

... Ignored

Method clone(): The objects of this class are cloneable with this method.

Usage:

```
ReverseProxy$clone(deep = FALSE)
```

Arguments:

deep Whether to make a deep clone.

Examples

```
# Create a reverse proxy forwarding requests to http://example.com
rev_prox <- ReverseProxy$new(
  "http://example.com"
)

# Use root to only proxy requests to a specific subpath
rev_prox <- ReverseProxy$new(
  "http://example.com",
  root = "forward"
)

# Use except to exempt certain subpaths from proxying
rev_prox <- ReverseProxy$new(
  "http://example.com",
  except = c("no_proxy", "dont/proxy")
)

## Attach it to a fiery app
if (requireNamespace("fiery", quietly = TRUE)) {
  app <- fiery::Fire$new()
  app$attach(rev_prox)
}
```

Index

ReverseProxy, [2](#)