Package 'fidelius'

October 13, 2022

Title Browser-Side Password-Protected HTML Documents

Version 0.0.2

Description Create secure, encrypted, and password-protected static HTML
documents that include the machinery for secure in-browser authentication
and decryption.

License MIT + file LICENSE

URL https://mattwarkentin.github.io/fidelius/,

https://github.com/mattwarkentin/fidelius

BugReports https://github.com/mattwarkentin/fidelius/issues

Encoding UTF-8

RoxygenNote 7.1.2

Imports askpass, fs, htmltools, readr, rlang, rmarkdown, rstudioapi,
sodium, whisker

NeedsCompilation no

Author Matthew T. Warkentin [aut, cre, cph]
(<https://orcid.org/0000-0001-8730-3511>)

Maintainer Matthew T. Warkentin <warkentin@lunenfeld.ca>

Repository CRAN

Date/Publication 2021-11-16 08:10:12 UTC

R topics documented:

---

charm   *Password protect a document*

Description

Encrypt and password protect an HTML document.

Usage

```
charm(
  input,
  password,
  hint = NULL,
  output = NULL,
  style = stylize(),
  bundle = FALSE,
  minified = TRUE,
  ...
)
```

Arguments

input	Path to an R Markdown or HTML file.
password	Private password to unlock file. If not provided, you will be asked to supply a password by a prompt if you are working from an interactive session.
hint	Optional public password hint.
output	Override the name of the output file.
style	Object returned from `stylize()`.
bundle	Logical. Should all of the decryption machinery and dependencies be bundled into the HTML document? Default is `FALSE`.
minified	Logical. Should minified versions of JavaScript dependencies be included? Default is `TRUE`.
...	Arguments passed on to `rmarkdown::render()`.

Value

`input`, invisibly.

Note

Using `bundle = TRUE` only applies to bundling the decryption machinery and dependencies for the document generated by `charm()`. It is still the users responsibility to decide on whether `input` is self-contained by passing `self_contained = TRUE` to `rmarkdown::render()` using `...`, or by specifying `self_contained: true` in the YAML header of `input`.

Examples

```
## Not run:
charm("input.Rmd")

charm("input.Rmd", password = "pw1234!")

## End(Not run)
```

```
html_password_protected
```
Password protected HTML document

Description

An R Markdown output format to encrypt and password protect an HTML document using `charm()`.
`secret_keeper()` is an alias for `html_password_protected`.

Usage

```
html_password_protected(
  output_format = "rmarkdown::html_document",
  password,
  hint = NULL,
  style = list(),
  bundle = FALSE,
  minified = TRUE,
  preview = FALSE,
  ...
)

secret_keeper(
  output_format = "rmarkdown::html_document",
  password,
  hint = NULL,
  style = list(),
  bundle = FALSE,
  minified = TRUE,
  preview = FALSE,
  ...
)
```

Arguments

output_format   An R Markdown format that renders to HTML. By default, will render to the
standard `rmarkdown::html_document()` format. You may pass any arguments
to your `output_format` as shown below:

```
                      ---
                      output:
                        fidelius::html_password_protected:
                          output_format:
                            rmarkdown::html_document:
                              toc: true
                      ---
```

password   Private password to unlock file. If not provided, you will be asked to supply a
                  password by a prompt if you are working from an interactive session.

hint   Optional public password hint.

style   Any number of style options that are configurable via `stylize()`. For example:

```
                      output:
                        fidelius::html_password_protected:
                          style:
                            button_text: "Open Sesame!"
```

bundle   Logical. Should all of the decryption machinery and dependencies be bundled
                  into the HTML document? Default is `FALSE`.

minified   Logical. Should minified versions of JavaScript dependencies be included? De-
                  fault is `TRUE`.

preview   Logical. Set to `FALSE` to bypass the encryption and password protection to ren-
                  der the HTML document specified by `output_format`. See Notes for more
                  details.

...   Not currently used.

Value

R Markdown output format based on `output_format` to pass to `rmarkdown::render()`.

Note

If you hard-code your `password` into the YAML header of the document be careful not to check the
R Markdown file into a public version control repository (e.g. git or GitHub) where the password is
stored and visible in plain-text. Please use this format carefully.

Try using `xaringan::infinite_moon_reader()` with `preview = TRUE` to preview your document
with automatic reloading of slides on change. When you are done developing your document, set
`preview = FALSE` or remove it from the YAML header entirely to encrypt and password protect
your document.

See Also

[charm](#) for more information on the password protection and encryption.

Examples

```
## Not run:
library(rmarkdown)
render("input.Rmd", html_password_protected())

## End(Not run)
```

```
stylize                    Style password landing page
```

Description

Use this function to style the appearance of the landing page for the password-protected HTML file.

Usage

```
stylize(
  header_text = "PLEASE ENTER THE PASSWORD",
  placeholder_text = "",
  button_text = "SUBMIT",
  font_family = "-apple-system",
  font_size = "16px",
  title_color = "#2d3737",
  background_color = "#f2f2f2",
  box_color = "#ffffff",
  btn_font_color = "#ffffff",
  btn_bg_color = "#19A974",
  btn_hover_color = "#137752",
  modal_overlay = "rgba(0, 0, 0, 0.6)",
  modal_box_color = "#ffffff",
  modal_title_color = "#137752",
  modal_font_color = "rgba(0, 0, 0, 0.8)"
)
```

Arguments

header_text   Header text.
placeholder_text

                   Password input placeholder text.

button_text   Button text.

font_family   Font family.

font_size   Font size.

title_color   Title font color.
background_color

                   Background color.

box_color   Box color.

btn_font_color Button font color.

btn_bg_color   Button background color.

btn_hover_color

                Button hover color.

modal_overlay   Modal overlay color.

modal_box_color

                Modal box color.

modal_title_color

                Modal title font color.

modal_font_color

                Modal text font color.

Value

A list with class fidelius_styling.

Examples

```
stylize(font_family = "Times")
```

Index