

Package ‘cognitoR’

January 30, 2024

Type Package

Title Authentication for 'Shiny' Apps with 'Amazon Cognito'

Version 1.0.5

Description Provides authentication for Shiny applications using 'Amazon Cognito' (<<https://aws.amazon.com/es/cognito/>>).

Imports shiny, config, shinyjs, httr, dplyr, base64enc, jsonlite, paws

License GPL-3

BugReports <https://github.com/chi2labs/cognitoR/issues>

Encoding UTF-8

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Pablo Pagnone [aut, cre],
Aleksander Dietrichson [aut]

Maintainer Pablo Pagnone <pablodp84@gmail.com>

Repository CRAN

Date/Publication 2024-01-30 16:10:02 UTC

R topics documented:

cognito_add_account	2
cognito_server	3
cognito_ui	4
cookie_server	4
cookie_ui	5
logout_server	6
logout_ui	6
Index	8

cognito_add_account *Allow create an user in Cognito Pool*

Description

This method is working with amazing package "paws" , so you need to have the required AWS secret and access key (see example).

Usage

```
cognito_add_account(  
  cognito_config,  
  Username,  
  UserAttributes,  
  DesiredDeliveryMediums,  
  ...  
)
```

Arguments

cognito_config - Cognito Config list
Username - Username to use in Cognito - This can be a username, email ,phone depending configuration in Amazon Pool.
UserAttributes - User attributes: This can be multiples attributes, depends of configuration in Amazon Pool.
DesiredDeliveryMediums - Medium to delivert email when user is created, can be EMAIL or SMS
... extra params to pass to method paws::admin_create_user

Details

Also needs to have the config for cognitoR. Specially the pool Id (group_id in config), where the new user is created.

Value

boolean

Examples

```
## Not run:  
Sys.setenv(  
  AWS_ACCESS_KEY_ID = '',  
  AWS_SECRET_ACCESS_KEY = '',  
  AWS_REGION = ''  
)  
cognito_add_account("account@mail.com",
```

```
UserAttributes = list(list(Name = "email", Value = "account@mail.com"),
                      list(Name = "phone_number", Value = "+12123212312321")
                      ),
                      DesiredDeliveryMediums = "EMAIL"
                      )

## End(Not run)
```

cognito_server

Cognito Shiny Module Server

Description

A shiny server function to implement Cognito Authentication in your shiny app.

Usage

```
cognito_server(
  input,
  output,
  session,
  with_cookie = FALSE,
  cookiename = "cognitor",
  cookie_expire = 7
)
```

Arguments

input	- Shiny input
output	- Shiny Output
session	- Shiny Session
with_cookie	- Create a own cookie when is authenticated in Cognito.
cookiename	- name for cookie
cookie_expire	- Expiration time for cookie

Value

reactiveValues (isLogged and userdata) and a callback function to do logout in Cognito.

Author(s)

Pablo Pagnone

 cognito_ui

Cognito Shiny Module UI

Description

A Shiny UI function to be used to implement Amazon Cognito in your shiny app. This UI loads the required JS.

Usage

```
cognito_ui(id)
```

Arguments

id character - Namespace ID

Value

A Shiny UI

Author(s)

Pablo Pagnone

Examples

```
cognito_ui("demo")
```

 cookie_server

Cookie Shiny Module Server

Description

This server module returns a list with methods to interact with cookie created via JS. You have:

Usage

```
cookie_server(input, output, session, cookie_name = "user", cookie_expire = 7)
```

Arguments

input shiny input
 output shiny output
 session shiny session
 cookie_name character - Name of cookie to create
 cookie_expire numeric - Expiration timeof cookie

Details

- getCookie - Reactive function, returns the content of cookie if exist. Else return FALSE.
- setCookie - Sets the content for a cookie. (Required param: list())
- rmCookie - Removes a cookie

Value

list of reactive element and function to manage cookies

Author(s)

Pablo Pagnone

Examples

```
cookie_server(NULL, NULL, NULL)
```

cookie_ui

Cookie Shiny Module UI

Description

This UI module load the required js methods to create/remove/get cookie in browser.

Usage

```
cookie_ui(id)
```

Arguments

id character

Value

a Shiny UI

Examples

```
cookie_ui("cookie")
```

logout_server	<i>Logout Shiny Module Server</i>
---------------	-----------------------------------

Description

Receive a reactive element (isLoggedIn), when is TRUE the logout button and information about is shown.

Usage

```
logout_server(
  input,
  output,
  session,
  isLoggedIn = reactive(FALSE),
  textlogged = "You are logged in"
)
```

Arguments

input	Shiny input
output	Shiny output
session	Shiny session
isLoggedIn	reactive
textlogged	character

Value

reactive (the logout button)

Examples

```
logout_server(NULL, NULL, NULL)
```

logout_ui	<i>Logout Shiny Module UI</i>
-----------	-------------------------------

Description

Return a UI with a logout button and information about user logged. By default is hidden and is show with reactive element from logoutServer

Usage

```
logout_ui(id, textbutton = "Log out", classbutton = "btn-logout btn-danger")
```

logout_ui

7

Arguments

id	character
textbutton	character
classbutton	character

Value

Shiny UI

Examples

```
logout_ui("test")
```

Index

`cognito_add_account`, 2

`cognito_server`, 3

`cognito_ui`, 4

`cookie_server`, 4

`cookie_ui`, 5

`logout_server`, 6

`logout_ui`, 6