

Package ‘Rrdap’

July 14, 2022

Title RDAP Server Querying

Version 1.0.6

Date 2022-07-12

Author Brad Cable

Maintainer Brad Cable <brad@bcable.net>

Description Queries data from RDAP servers.

Imports curl, rjson, stringr, utils

License MIT + file LICENSE

NeedsCompilation no

Repository CRAN

Date/Publication 2022-07-14 11:00:12 UTC

R topics documented:

| | |
|-----------------|----------|
| Rrdap | 1 |
| Index | 4 |

| | |
|-------|-----------------------------|
| Rrdap | <i>RDAP server querying</i> |
|-------|-----------------------------|

Description

Queries RDAP servers

Usage

```
rdap_query(entities, rdap_uri="https://rdap-bootstrap.arin.net/bootstrap/",  
           query_entity=FALSE, debug=FALSE)  
  
rdap_keyextract(query_ret, key)
```

```
rdap_keynames(query_ret)

rdap_extract_df(query_ret, sub_name)

rdap_keyval_extract(query_ret, keys, blacklist_values=NULL, unlist.recursive=TRUE)
```

Arguments

| | |
|-------------------------------|---|
| <code>entities</code> | Entity (or entities as vector) to query RDAP for. Can IP addresses, domains, or a mix of the two. |
| <code>rdap_uri</code> | RDAP URI to use as the search base. This should not need to be anything other than the default bootstrap URI unless you really want to do something specific. |
| <code>query_entity</code> | Manually override what type of entity is being queried. Not recommended. |
| <code>query_ret</code> | Data structure that was returned from <code>rdap_query()</code> |
| <code>sub_name</code> | Name of sub-table to return as a flat list |
| <code>key</code> | Single string, case-sensitive, of what key to look to return back values from the RDAP return |
| <code>keys</code> | Single string or vector of strings, case-insensitive, of what keys to look for in-order to return back values of from the RDAP return |
| <code>debug</code> | Boolean if you want debug prints (good for large quantities of manual data) |
| <code>blacklist_values</code> | Character or vector of values (or starting values) to skip when searching |
| <code>unlist.recursive</code> | Boolean if you want the unlist at the end to pass recursively |

Value

`rdap_query()` returns a large `JSON data.frame()` `rdap_extract_df()` returns a `data.frame` `rdap_keyextract()` returns a vector `rdap_keynames()` returns a vector `rdap_keyval_extract()` returns a vector

Author(s)

Brad Cable

Examples

```
# Grab RDAP data for a domain
rdap_query("bcable.net")

# Grab RDAP data for an IP
rdap_query("1.1.1.1")

# Grab RDAP data for a domain from a specific RDAP provider
rdap_query("bcable.net", rdap_uri="https://rdap.verisign.com/net/v1/")

# Grab multiple mixed vectorized results
hosts_ips <- c("1.0.0.1", "bcable.net")
rdap_data <- rdap_query(hosts_ips)
```

```
rdap_data

# Extract Country Info About Domains
countries <- rdap_keyextract(rdap_data, "country")

# Get list of available attribute names for each entry
rdap_keynames(rdap_data)

# Get more traditional WHOIS style registration data
entity_data <- rdap_extract_df(rdap_data, "entities")

# Grab default role reported
rdap_keyval_extract(entity_data, "roles")
```

Index

- * **Rrdap**

 - Rrdap, [1](#)

- * **rdap_extract_df**

 - Rrdap, [1](#)

- * **rdap_keyextract**

 - Rrdap, [1](#)

- * **rdap_keynames**

 - Rrdap, [1](#)

- * **rdap_keyval_extract**

 - Rrdap, [1](#)

- * **rdap_query**

 - Rrdap, [1](#)

- * **rdap**

 - Rrdap, [1](#)

[rdap_extract_df \(Rrdap\), 1](#)

[rdap_keyextract \(Rrdap\), 1](#)

[rdap_keynames \(Rrdap\), 1](#)

[rdap_keyval_extract \(Rrdap\), 1](#)

[rdap_query \(Rrdap\), 1](#)

[Rrdap, 1](#)

[Rrdap::rdap_extract_df \(Rrdap\), 1](#)

[Rrdap::rdap_keyextract \(Rrdap\), 1](#)

[Rrdap::rdap_keynames \(Rrdap\), 1](#)

[Rrdap::rdap_keyval_extract \(Rrdap\), 1](#)

[Rrdap::rdap_query \(Rrdap\), 1](#)