cdrouter.py Documentation

Release 0.4.10

QA Cafe

Sep 12, 2017

Getting Started

1 Introduction **1**
- 1.1 Download & Install . . . 1
- 1.2 Usage . . . 1

2 Reference **3**
- 2.1 CDRouter . . . 3
- 2.2 Filters . . . 5
- 2.3 Configs . . . 7
- 2.4 Devices . . . 13
- 2.5 PowerCmd . . . 17
- 2.6 Connection . . . 17
- 2.7 Attachments . . . 17
- 2.8 Jobs . . . 19
- 2.9 Packages . . . 22
- 2.10 Results . . . 26
- 2.11 Update . . . 35
- 2.12 TestResults . . . 36
- 2.13 Annotations . . . 40
- 2.14 Captures . . . 41
- 2.15 Highlights . . . 46
- 2.16 Imports . . . 48
- 2.17 Exports . . . 50
- 2.18 History . . . 51
- 2.19 System . . . 52
- 2.20 Tags . . . 57
- 2.21 Testsuites . . . 58
- 2.22 Users . . . 64

Python Module Index **67**

CHAPTER 1

Introduction

cdrouter is a simple Python wrapper for the CDRouter Web API. https://support.qacafe.com/cdrouter-web-api/

For more information on CDRouter, please visit http://www.qacafe.com/.

Download & Install

cdrouter is available on PyPI.

```
$ pip install -U cdrouter
```

Usage

First create a *CDRouter* object, passing it the URL of your CDRouter system, for example `http://localhost`, `https://cdrouter.example.com` or `http://172.20.0.1:8015`. If your CDRouter system uses a self-signed certificate for HTTPS connections (the default), you will need to pass in `insecure=True` to disable certificate validation. If Automatic Login is not enabled on your CDRouter system, you will need to provide authentication credentials by passing in an API token via the `token` parameter or a username and password via the `username` and `password` parameters. If `token` is not specified, it will default to the value of the `CDROUTER_API_TOKEN` environment variable. *CDRouter* will prompt for a username and password as necessary if Automatic Login is not enabled and an API token is not provided.

```
import time

from cdrouter import CDRouter
from cdrouter.cdrouter import CDRouterError
from cdrouter.filters import Field as field
from cdrouter.jobs import Job

c = CDRouter('http://localhost:8015', token='deadbeef')
```

```
for p in c.packages.iter_list(filter=field('tags').contains('noretry')):
    print('Launching package ' + p.name)

    try:
        j = c.jobs.launch(Job(package_id=p.id, extra_cli_args='-testvar myvar=example
↪'))
    except CDRouterError as ce:
        print('Error launching job: {}'.format(ce))

    while j.result_id == None:
        time.sleep(1)
        j = c.jobs.get(j.id)

    print('    Result-ID: ' + j.result_id)

print('done.')
```

More examples of using cdrouter can be found here. Please see the *Reference* page for more information on available fields and methods for each class.

CHAPTER 2

Reference

Python client for the CDRouter Web API.

CDRouter

CDRouter

class cdrouter.**CDRouter**(*base*, *token=None*, *username=None*, *password=None*, *_getuser=<function _getuser_default>*, *_getpass=<function _getpass_default>*, *insecure=False*)

Service for accessing the CDRouter Web API.

Parameters

- **base** – Base HTTP or HTTPS URL for CDRouter system as a string, optionally including a port. For example *http://localhost*, *http://cdrouter.lan:8015* or *https://127.0.0.1*.
- **token** – (optional) CDRouter API token as a string. Not required if Automatic Login is enabled. If omitted, value will be taken from CDROUTER_API_TOKEN environment variable.
- **username** – (optional) Username as string. Can be omitted if `token` is specified or Automatic Login is enabled. If omitted, `_getuser` will be called when a username is required.
- **password** – (optional) Password as string. Can be omitted if `token` is specified or Automatic Login is enabled. If omitted, `_getpass` will be called when a password is required.
- **_getuser** – (optional) If username is *None*, function to be called as `_getuser(base)` which returns a username as a string. If `_getuser` is *None*, `cdrouter` will print a prompt to stdout and read the username from stdin.
- **_getpass** – (optional) If password is *None*, a function to be called as `_getpass(base, username)` which returns user's password as a string. If `_getpass` is *None*, `cdrouter` will print a password prompt to stdout and read the password from stdin.

- **insecure** – (optional) If bool *True* and *base* is an HTTPS URL, skip certificate verification and allow insecure connections to the CDRouter system.

annotations = None
annotations.AnnotationsService object

attachments = None
attachments.AttachmentsService object

authenticate(*retries=3*)
Set API token by authenticating via username/password.

Parameters **retries** – Number of authentication attempts to make before giving up as an int.

Returns Learned API token

Return type string

captures = None
captures.CapturesService object

configs = None
configs.ConfigsService object

devices = None
devices.DevicesService object

exports = None
exports.ExportsService object

highlights = None
highlights.HighlightsService object

history = None
history.HistoryService object

imports = None
imports.ImportsService object

jobs = None
jobs.JobsService object

packages = None
packages.PackagesService object

results = None
results.ResultsService object

system = None
system.SystemService object

tags = None
tags.TagsService object

tests = None
testresults.TestResultsService object

testsuites = None
testsuites.TestsuitesService object

users = None
users.UsersService object

CDRouterError

class cdrouter.cdrouter.**CDRouterError**(**args*, ***kwargs*)
Class for representing CDRouter Web API errors.

Links

class cdrouter.cdrouter.**Links**(***kwargs*)
Class representing paging information returned by `list` calls to the CDRouter Web API.

Parameters

- **first** – (optional) First page number as an int.
- **last** – (optional) Last page number as an int.
- **current** – (optional) Current page number as an int.
- **total** – (optional) Total element count across all pages as an int.
- **limit** – (optional) Resources per page limit as an int.
- **next** – (optional) Next page number as an int.
- **prev** – (optional) Previous page number as an int.

Share

class cdrouter.cdrouter.**Share**(***kwargs*)
Model for CDRouter Shares.

Parameters

- **user_id** – (optional) User ID as an int.
- **read** – (optional) Bool *True* if reading is allowed.
- **write** – (optional) Bool *True* if writing is allowed.
- **execute** – (optional) Bool *True* if executing is allowed.

Filters

Field

class cdrouter.filters.**Field**(**args*)
Class for constructing CDRouter Web API filters. Field objects can be passed to `filter` parameters to make constructing filters easier.

The `value` parameter passed to Field methods is converted to a string using the builtin `str` function. Bool values are additionally lowercased such that bool `True` becomes string `'true'` and `False` becomes string `'false'`. The Field methods `contains`, `contained_by` and `overlaps` accept a variable number of values which are converted into a list in accordance with the API's syntax. For example, the int values `1, 2, 3` is converted to the string `'{1,2,3}'` and the string values `'one', 'two', 'three'` is converted to the string `'{one,two,three}'`.

Usage:

```
from cdrouter.filters import Field as field
# each pair of filters below are equivalent
c.packages.list(filter=[
    field('id').eq(123),
    'id=123',
    #
    field('use_as_testlist').ne(True),
    'use_as_testlist!=true',
    #
    field('name').not_().match('^(foo|bar)$', ignorecase=True),
    'name!~*^(foo|bar)$',
    #
    field('tags').contains('foo', 'bar', 'baz'),
    'tags@>{foo,bar,baz}',
])
```

See this page for more details on CDRouter Web API filters.

contained_by(**args*)
Construct an array contained by (<@) filter.

Parameters **args** – Filter values

Returns *filters.Field* object

Return type *filters.Field*

contains(**args*)
Construct an array contains (@>) filter.

Parameters **args** – Filter values

Returns *filters.Field* object

Return type *filters.Field*

eq(*value*)
Construct an equal to (=) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

ge(*value*)
Construct a greater than or equal to (>=) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

gt(*value*)
Construct a greater than (>) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

le(*value*)
Construct a less than or equal to (<=) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

lt(*value*)
Construct a less than (<) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

match(*value*, *ignorecase=False*)
Construct a regexp match (~) filter. Combine with `not_` method to construct a negative regexp match (!~) filter.

Parameters

- **value** – Filter value
- **ignorecase** – If bool *True*, make match case insensitive (~*, !~*)

Returns *filters.Field* object

Return type *filters.Field*

ne(*value*)
Construct a not equal to (!=) filter.

Parameters **value** – Filter value

Returns *filters.Field* object

Return type *filters.Field*

not_()
Negate the filter. Not supported by `contains`, `contained_by` or `overlaps` methods.

Returns *filters.Field* object

Return type *filters.Field*

overlaps(**args*)
Construct an array overlaps (&&) filter.

Parameters **args** – Filter values

Returns *filters.Field* object

Return type *filters.Field*

Configs

ConfigsService

class cdrouter.configs.**ConfigsService**(*service*)
Service for accessing CDRouter Configs.

bulk_copy(*ids*)
Bulk copy a set of configs.

Parameters **ids** – Int list of config IDs.

Returns *configs.Config* list

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of configs.

Parameters

- **ids** – (optional) Int list of config IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_edit(*_fields*, *ids=None*, *filter=None*, *type=None*, *all=False*, *testvars=None*)
Bulk edit a set of configs.

Parameters

- **_fields** – *configs.Config* object
- **ids** – (optional) Int list of config IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.
- **testvars** – (optional) `configs.ConfigTestvars` list

bulk_edit_testvars(*id*, *testvars*)
Bulk edit a config's testvars.

Parameters

- **id** – Config ID as an int.
- **testvars** – *configs.Testvar* list

Returns *configs.Testvar* list

bulk_export(*ids*)
Bulk export a set of configs.

Parameters **ids** – Int list of config IDs.

Return type tuple *(io.BytesIO, 'filename')*

check_config(*contents*)
Process config contents with cdrouter-cli -check-config.

Parameters **contents** – Config contents as string.

Returns *configs.CheckConfig* object

Return type *configs.CheckConfig*

create(*resource*)
Create a new config.

Parameters **resource** – *configs.Config* object

Returns *configs.Config* object

Return type *configs.Config*

delete(*id*)
Delete a config.

Parameters **id** – Config ID as an int.

delete_testvar(*id*, *name*, *group=None*)
Delete a testvar in a config. Deleting a testvar unsets any explicitly configured value for it in the config.

Parameters

- **id** – Config ID as an int.
- **name** – Testvar name as string.
- **group** – (optional) Testvar group as string.

edit(*resource*)
Edit a config.

Parameters **resource** – *configs.Config* object

Returns *configs.Config* object

Return type *configs.Config*

edit_shares(*id*, *user_ids*)
Edit shares for a config.

Parameters

- **id** – Config ID as an int.
- **user_ids** – User IDs as int list.

Returns *cdrouter.Share* list

edit_testvar(*id*, *resource*)
Edit a testvar in a config.

Parameters

- **id** – Config ID as an int.
- **resource** – *configs.Testvar* object.

Returns *configs.Testvar* object

Return type *configs.Testvar*

export(*id*)
Export a config.

Parameters **id** – Config ID as an int.

Return type tuple *(io.BytesIO, 'filename')*

get(*id*)
Get a config.

Parameters **id** – Config ID as an int.

Returns *configs.Config* object

Return type *configs.Config*

get_by_name(*name*)
Get a config by name.

Parameters **name** – Config name as string.

Returns *configs.Config* object

Return type *configs.Config*

get_networks(*contents*)
Process config contents with cdrouter-cli -print-networks-json.

Parameters **contents** – Config contents as string.

Returns *configs.Networks* object

Return type *configs.Networks*

get_new()
Get output of cdrouter-cli -new-config.

Return type string

get_plaintext(*id*)
Get a config as plaintext.

Parameters **id** – Config ID as an int.

Return type string

get_shares(*id*)
Get shares for a config.

Parameters **id** – Config ID as an int.

Returns *cdrouter.Share* list

get_testvar(*id*, *name*, *group=None*)
Get a testvar from a config.

Parameters

- **id** – Config ID as an int.
- **name** – Testvar name as string.
- **group** – (optional) Testvar group as string.

Returns *configs.Testvar* object

Return type *configs.Testvar*

iter_list(**args*, ***kwargs*)
Get a list of configs. Whereas `list` fetches a single page of configs according to its `limit` and `page` arguments, `iter_list` returns all configs by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *configs.Config* list

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of configs.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *configs.Page* object

list_testvars(*id*)
Get a list of a config's testvars.

Parameters **id** – Config ID as an int.

Returns *configs.Testvar* list

upgrade_config(*contents*)
Process config contents with cdrouter-cli -upgrade-config.

Parameters **contents** – Config contents as string.

Returns *configs.UpgradeConfig* object

Return type *configs.UpgradeConfig*

Config

class cdrouter.configs.**Config**(***kwargs*)
Model for CDRouter Configs.

Parameters

- **id** – (optional) Config ID as an int.
- **name** – (optional) Config name as string.
- **description** – (optional) Config description as string.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **contents** – (optional) Config contents as string.
- **user_id** – (optional) User ID as an int.
- **result_id** – (optional) Result ID as an int (if a config snapshot).
- **tags** – (optional) Tags as string list.
- **note** – (optional) Note as string.

Page

class cdrouter.configs.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *configs.Config* list
- **links** – *cdrouter.Links* object

ConfigError

class cdrouter.configs.**ConfigError**(***kwargs*)
Model for CDRouter Check Config Error.

Parameters

- **lines** – (optional) Line numbers as string list.
- **error** – (optional) Error message as string.

CheckConfig

class cdrouter.configs.**CheckConfig**(***kwargs*)
Model for CDRouter Check Config.

Parameters **errors** – (optional) *configs.ConfigError* list

UpgradeConfig

class cdrouter.configs.**UpgradeConfig**(***kwargs*)
Model for CDRouter Config Upgrades.

Parameters

- **success** – (optional) Bool *True* if successful.
- **output** – (optional) Output as string.

Networks

class cdrouter.configs.**Networks**(***kwargs*)
Model for CDRouter Config Networks.

Parameters

- **name** – (optional) Network name as string.
- **type** – (optional) Network type as string.
- **side** – (optional) Network side as string.
- **title** – (optional) Network title as string.
- **children** – (optional) *configs.Networks* list.

Testvar

class cdrouter.configs.**Testvar**(***kwargs*)
Model for CDRouter Config Testvars.

Parameters

- **group** – (optional) Testvar group name as string.
- **name** – (optional) Testvar name as string.
- **value** – (optional) Testvar value as string.

- **default** – (optional) Testvar default value as string.
- **isdefault** – (optional) Bool *True* if testvar is set to default value.
- **line** – (optional) Config file line number as int.

Devices

DevicesService

class cdrouter.devices.**DevicesService**(*service*)
Service for accessing CDRouter Devices.

bulk_copy(*ids*)
Bulk copy a set of devices.

Parameters **ids** – Int list of device IDs.

Returns *devices.Device* list

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of devices.

Parameters

- **ids** – (optional) Int list of device IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_edit(*_fields*, *ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk edit a set of devices.

Parameters

- **_fields** – *devices.Device* object
- **ids** – (optional) Int list of device IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_export(*ids*)
Bulk export a set of devices.

Parameters **ids** – Int list of device IDs.

Return type tuple *(io.BytesIO, 'filename')*

connect(*id*)
Open proxy connection to a device's management interface.

Parameters **id** – Device ID as an int.

Returns *devices.Connection* object

Return type *devices.Connection*

create(*resource*)
Create a new device.

Parameters **resource** – *devices.Device* object

Returns *devices.Device* object

Return type *devices.Device*

delete(*id*)
Delete a device.

Parameters **id** – Device ID as an int.

disconnect(*id*)
Close proxy connection to a device's management interface.

Parameters **id** – Device ID as an int.

edit(*resource*)
Edit a device.

Parameters **resource** – *devices.Device* object

Returns *devices.Device* object

Return type *devices.Device*

edit_shares(*id*, *user_ids*)
Edit shares for a device.

Parameters

- **id** – Device ID as an int.
- **user_ids** – User IDs as int list.

Returns *cdrouter.Share* list

export(*id*)
Export a device.

Parameters **id** – Device ID as an int.

Return type tuple *(io.BytesIO, 'filename')*

get(*id*)
Get a device.

Parameters **id** – Device ID as an int.

Returns *devices.Device* object

Return type *devices.Device*

get_by_name(*name*)
Get a device by name.

Parameters **name** – Device name as string.

Returns *devices.Device* object

Return type *devices.Device*

get_connection(*id*)
Get information on proxy connection to a device's management interface.

Parameters **id** – Device ID as an int.

Returns *devices.Connection* object

Return type *devices.Connection*

get_shares(*id*)
Get shares for a device.

Parameters **id** – Device ID as an int.

Returns *cdrouter.Share* list

iter_list(**args*, ***kwargs*)
Get a list of devices. Whereas `list` fetches a single page of devices according to its `limit` and `page` arguments, `iter_list` returns all devices by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *devices.Device* list

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of devices.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *devices.Page* object

power_off(*id*)
Power off a device using it's power off command.

Parameters **id** – Device ID as an int.

Returns *devices.PowerCmd* object

Return type *devices.PowerCmd*

power_on(*id*)
Power on a device using it's power on command.

Parameters **id** – Device ID as an int.

Returns *devices.PowerCmd* object

Return type *devices.PowerCmd*

Device

class cdrouter.devices.**Device**(***kwargs*)
Model for CDRouter Devices.

Parameters

- **id** – (optional) Device ID as an int.

- **name** – (optional) Name as string.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **user_id** – (optional) User ID as an int.
- **result_id** – (optional) Result ID as an int (if a device snapshot).
- **attachments_dir** – (optional) Filepath for attachments as string.
- **picture_id** – (optional) Attachment ID for used for device picture as an int.
- **tags** – (optional) Tags as string list.
- **default_ip** – (optional) Default IP as a string
- **default_login** – (optional) Default login as a string
- **default_password** – (optional) Default password as a string
- **location** – (optional) Location as a string
- **device_category** – (optional) Device category as a string
- **manufacturer** – (optional) Manufacturer as a string
- **manufacturer_oui** – (optional) Manufacturer OUI as a string
- **model_name** – (optional) Model name as a string
- **model_number** – (optional) Model number as a string
- **description** – (optional) Description as a string
- **product_class** – (optional) Product class as a string
- **serial_number** – (optional) Serial number as a string
- **hardware_version** – (optional) Hardware version as a string
- **software_version** – (optional) Software version as a string
- **provisioning_code** – (optional) Provisioning code as a string
- **note** – (optional) Note as a string
- **insecure_mgmt_url** – (optional) *True* if insecure HTTPS management URLs are allowed
- **mgmt_url** – (optional) Management URL as a string
- **add_mgmt_addr** – (optional) *True* if address should be configured when opening proxy connection
- **mgmt_interface** – (optional) Interface on which to configure address as string
- **mgmt_addr** – (optional) Address to configure as string
- **power_on_cmd** – (optional) Command to run to power on device as string
- **power_off_cmd** – (optional) Command to run to power off device as string

Page

class cdrouter.devices.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *devices.Device* list
- **links** – *cdrouter.Links* object

PowerCmd

class cdrouter.devices.**PowerCmd**(***kwargs*)
Model for CDRouter Device Power command result.

Parameters **output** – (optional) Output from power on/off command as string.

Connection

class cdrouter.devices.**Connection**(***kwargs*)
Model for CDRouter Device Connections.

Parameters

- **proxy_port** – (optional) HTTP proxy port as an int.
- **proxy_https** – (optional) HTTPS proxy port as an int.

Attachments

AttachmentsService

class cdrouter.attachments.**AttachmentsService**(*service*)

create(*id*, *fd*, *filename='attachment-name'*)
Add an attachment to a device.

Parameters

- **id** – Device ID as an int.
- **fd** – File-like object to upload.
- **filename** – (optional) Name to use for new attachment as a string.

Returns *attachments.Attachment* object

Return type *attachments.Attachment*

delete(*id*, *attid*)
Delete a device's attachment.

Parameters

- **id** – Device ID as an int.

- **attid** – Attachment ID as an int.

download(*id*, *attid*)
Download a device's attachment.

Parameters

- **id** – Device ID as an int.
- **attid** – Attachment ID as an int.

Return type tuple *(io.BytesIO, 'filename')*

edit(*resource*)
Edit a device's attachment.

Parameters **resource** – `attachments.Attachment` object

Returns `attachments.Attachment` object

Return type *attachments.Attachment*

get(*id*, *attid*)
Get a device's attachment.

Parameters

- **id** – Device ID as an int.
- **attid** – Attachment ID as an int.

Returns `attachments.Attachment` object

Return type *attachments.Attachment*

iter_list(*id*, **args*, ***kwargs*)
Get a list of attachments. Whereas `list` fetches a single page of attachments according to its `limit` and `page` arguments, `iter_list` returns all attachments by internally making successive calls to `list`.

Parameters

- **id** – Device ID as an int.
- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns `attachments.Attachment` list

list(*id*, *filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of a device's attachments.

Parameters

- **id** – Device ID as an int.
- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns `attachments.Page` object

thumbnail(*id*, *attid*, *size=None*)
Download thumbnail of a device's attachment. Attachment must be a GIF, JPEG or PNG image.

Parameters

- **id** – Device ID as an int.
- **attid** – Attachment ID as an int.
- **size** – (optional) Height in pixels of generated thumbnail.

Return type tuple *(io.BytesIO, 'filename')*

Attachment

class cdrouter.attachments.**Attachment**(***kwargs*)
Model for CDRouter Attachments.

Parameters

- **id** – (optional) Attachment ID as an int.
- **name** – (optional) Name as string.
- **description** – (optional) Description as string.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **size** – (optional) Attachment size as an int.
- **path** – (optional) Filepath to attachment as string.
- **device_id** – (optional) Device ID as an int.

Page

class cdrouter.attachments.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *attachments.Attachment* list
- **links** – *cdrouter.Links* object

Jobs

JobsService

class cdrouter.jobs.**JobsService**(*service*)
Service for accessing CDRouter Jobs.

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of jobs.

Parameters

- **ids** – (optional) Int list of job IDs.

- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_launch(*jobs=None*, *filter=None*, *all=False*)
Bulk launch a set of jobs.

Parameters

- **jobs** – *jobs.Job* list
- **filter** – (optional) Filters to apply as a string list.
- **all** – (optional) Apply to all if bool *True*.

delete(*id*)
Delete a job.

Parameters **id** – Job ID as an int.

edit(*resource*)
Edit a job.

Parameters **resource** – *jobs.Job* object

Returns *jobs.Job* object

Return type *jobs.Job*

get(*id*)
Get a job.

Parameters **id** – Job ID as an int.

Returns *jobs.Job* object

Return type *jobs.Job*

iter_list(**args*, ***kwargs*)
Get a list of jobs. Whereas `list` fetches a single page of jobs according to its `limit` and `page` arguments, `iter_list` returns all jobs by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *jobs.Job* list

launch(*resource*)
Launch a new job.

Parameters **resource** – *jobs.Job* object

Returns *jobs.Job* object

Return type *jobs.Job*

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of jobs.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.

- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *jobs.Page* object

Job

class cdrouter.jobs.**Job**(***kwargs*)
Model for CDRouter Jobs.

Parameters

- **id** – (optional) Job ID as an int.
- **status** – (optional) Bool *True* if user is an administrator.
- **options** – (optional) *jobs.Options* object
- **package_id** – (optional) Package ID as an int.
- **package_name** – (optional) Package name as string.
- **device_id** – (optional) Device ID as an int.
- **device_name** – (optional) Device name as string.
- **result_id** – (optional) Result ID as an int.
- **user_id** – (optional) User ID as an int.
- **created** – (optional) Job creation time as *DateTime*.
- **updated** – (optional) Job last-updated time as *DateTime*.

Options

class cdrouter.jobs.**Options**(***kwargs*)
Model for CDRouter Job Options.

Parameters

- **tags** – (optional) Tags as string list.
- **skip_tests** – (optional) Tests to skip as string list.
- **begin_at** – (optional) Test name to begin testing at as string.
- **end_at** – (optional) Test name to end testing at as string.
- **extra_cli_args** – (optional) Extra *cdrouter-cli* arguments as string.

Page

class cdrouter.jobs.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *jobs.Job* list

- **links** – *cdrouter.Links* object

Packages

PackagesService

class cdrouter.packages.**PackagesService**(*service*)
Service for accessing CDRouter Packages.

analyze(*id*)
Get a list of tests that will be skipped for a package.

Parameters **id** – Package ID as an int.

Returns *packages.Analysis* object

Return type *packages.Analysis*

bulk_copy(*ids*)
Bulk copy a set of packages.

Parameters **ids** – Int list of package IDs.

Returns *packages.Package* list

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of packages.

Parameters

- **ids** – (optional) Int list of package IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_edit(*_fields*, *ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk edit a set of packages.

Parameters

- **_fields** – *packages.Package* object
- **ids** – (optional) Int list of package IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_export(*ids*)
Bulk export a set of packages.

Parameters **ids** – Int list of package IDs.

Return type tuple *(io.BytesIO, 'filename')*

create(*resource*)
Create a new package.

Parameters **resource** – *packages.Package* object

Returns *packages.Package* object

Return type *packages.Package*

delete(*id*)
Delete a package.

Parameters **id** – Package ID as an int.

edit(*resource*)
Edit a package.

Parameters **resource** – *packages.Package* object

Returns *packages.Package* object

Return type *packages.Package*

edit_shares(*id*, *user_ids*)
Edit shares for a package.

Parameters

- **id** – Package ID as an int.
- **user_ids** – User IDs as int list.

Returns *cdrouter.Share* list

export(*id*)
Export a package.

Parameters **id** – Package ID as an int.

Return type tuple *(io.BytesIO, 'filename')*

get(*id*)
Get a package.

Parameters **id** – Package ID as an int.

Returns *packages.Package* object

Return type *packages.Package*

get_by_name(*name*)
Get a package by name.

Parameters **name** – Package name as string.

Returns *packages.Package* object

Return type *packages.Package*

get_shares(*id*)
Get shares for a package.

Parameters **id** – Package ID as an int.

Returns *cdrouter.Share* list

iter_list(**args*, ***kwargs*)
Get a list of packages. Whereas `list` fetches a single page of packages according to its `limit` and `page` arguments, `iter_list` returns all packages by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.

- **kwargs** – Optional arguments that `list` takes.

Returns *packages.Package* list

list (*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of packages.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *packages.Page* object

Package

class cdrouter.packages.**Package** (***kwargs*)
Model for CDRouter Packages.

Parameters

- **id** – (optional) Package ID as an int.
- **name** – (optional) Name as a string.
- **description** – (optional) Description as a string.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **test_count** – (optional) Test count as an int.
- **testlist** – (optional) Testlist as a string list.
- **extra_cli_args** – (optional) Extra CLI args as a string.
- **user_id** – (optional) User ID as an int.
- **agent_id** – (optional) Agent ID as an int.
- **config_id** – (optional) Config ID as an int.
- **result_id** – (optional) Result ID as an int (if a package snapshot).
- **device_id** – (optional) Device ID as an int.
- **options** – (optional) *packages.Options* object
- **tags** – (optional) Tags as a string list.
- **use_as_testlist** – (optional) Bool *True* if package is used as a testlist.
- **note** – (optional) Note as a string.

Options

class cdrouter.packages.**Options**(***kwargs*)
Model for CDRouter Package Options.

Parameters

- **forever** – (optional) Bool *True* if package looped forver.
- **loop** – (optional) Loop count as an int.
- **repeat** – (optional) Repeat count as an int.
- **maxfail** – (optional) Max fail count as an int.
- **duration** – (optional) Max testing time duration as an int.
- **wait** – (optional) Wait between tests duration as an int.
- **pause** – (optional) Bool *True* is pausing between tests.
- **shuffle** – (optional) Bool *True* if testlist is shuffled.
- **seed** – (optional) Shuffle seed as an int.
- **retry** – (optional) Retry count as an int.
- **rdelay** – (optional) Retry delay as an int.

Page

class cdrouter.packages.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *packages.Package* list
- **links** – *cdrouter.Links* object

Analysis

class cdrouter.packages.**Analysis**(***kwargs*)
Model for CDRouter Package Analysis.

Parameters

- **total_count** – (optional) Total count as an int.
- **run_count** – (optional) Run count as an int.
- **skipped_count** – (optional) Skipped test count as an int.
- **skipped_tests** – (optional) Skipped tests as a string list.

Results

ResultsService

class cdrouter.results.**ResultsService**(*service*)
Service for accessing CDRouter Results.

all_stats()
Compute stats for all results.

Returns *results.AllStats* object

Return type *results.AllStats*

bulk_copy(*ids*)
Bulk copy a set of results.

Parameters **ids** – Int list of result IDs.

Returns *results.Result* list

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of results.

Parameters

- **ids** – (optional) Int list of result IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_edit(*_fields*, *ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk edit a set of results.

Parameters

- **_fields** – *results.Result* object
- **ids** – (optional) Int list of result IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_export(*ids*, *exclude_captures=False*)
Bulk export a set of results.

Parameters **ids** – Int list of result IDs.

Return type tuple *(io.BytesIO, 'filename')*

delete(*id*)
Delete a result.

Parameters **id** – Result ID as an int.

diff_stats(*ids*)
Compute diff stats for a set of results.

Parameters **id** – Result IDs as int list.

Returns `results.SetStats` object

Return type *results.SetStats*

download_logdir_archive(*id*, *format='zip'*, *exclude_captures=False*)
Download logdir archive in tgz or zip format.

Parameters

- **id** – Result ID as an int.
- **format** – (optional) Format to download, must be string *zip* or *tgz*.
- **exclude_captures** – If bool *True*, don't include capture files

Return type tuple *(io.BytesIO, 'filename')*

edit(*resource*)
Edit a result.

Parameters **resource** – `results.Result` object

Returns `results.Result` object

Return type *results.Result*

edit_shares(*id*, *user_ids*)
Edit shares for a result.

Parameters

- **id** – Result ID as an int.
- **user_ids** – User IDs as int list.

Returns `cdrouter.Share` list

export(*id*, *exclude_captures=False*)
Export a result.

Parameters

- **id** – Result ID as an int.
- **exclude_captures** – If bool *True*, don't export capture files

Return type tuple *(io.BytesIO, 'filename')*

get(*id*)
Get a result.

Parameters **id** – Result ID as an int.

Returns `results.Result` object

Return type *results.Result*

get_logdir_file(*id*, *filename*)
Download a logdir file.

Parameters

- **id** – Result ID as an int.
- **filename** – Logdir filename as string.

Return type tuple *(io.BytesIO, 'filename')*

get_shares(*id*)
Get shares for a result.

Parameters **id** – Result ID as an int.

Returns *cdrouter.Share* list

get_test_metric(*id*, *name*, *metric*)
Get a test metric.

Parameters

- **id** – Result ID as an int.
- **name** – Test name as string.
- **metric** – Metric name as string.

Returns *results.Metric* object

Return type *results.Metric*

get_test_metric_csv(*id*, *name*, *metric*)
Get a test metric as CSV.

Parameters

- **id** – Result ID as an int.
- **name** – Test name as string.
- **id** – Metric name as string.

Return type string

iter_list(**args*, ***kwargs*)
Get a list of results. Whereas `list` fetches a single page of results according to its `limit` and `page` arguments, `iter_list` returns all results by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *results.Result* list

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of results.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *results.Page* object

list_csv(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of results as CSV.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Return type string

list_logdir(*id*, *filter=None*, *sort=None*)
Get a list of logdir files.

Parameters

- **id** – Result ID as an int.
- **filter** – Filter to apply as string.
- **sort** – Sort field to apply as string.

Returns *results.LogDirFile* list

pause(*id*, *when=None*)
Pause a running result.

Parameters

- **id** – Result ID as an int.
- **when** – Must be string *end-of-test* or *end-of-loop*.

pause_end_of_loop(*id*)
Pause a running result at the end of the current loop.

Parameters **id** – Result ID as an int.

pause_end_of_test(*id*)
Pause a running result at the end of the current test.

Parameters **id** – Result ID as an int.

progress_stats(*id*)
Compute progress stats for a result.

Parameters **id** – Result ID as an int.

Returns *results.Progress* object

Return type *results.Progress*

set_stats(*ids*)
Compute stats for a set of results.

Parameters **id** – Result IDs as int list.

Returns *results.SetStats* object

Return type *results.SetStats*

single_stats(*id*)
Compute stats for a result.

Parameters **id** – Result ID as an int.

Returns *results.SingleStats* object

Return type *results.SingleStats*

stop(*id*, *when=None*)
Stop a running result.

Parameters

- **id** – Result ID as an int.
- **when** – Must be string *end-of-test* or *end-of-loop*.

stop_end_of_loop(*id*)
Stop a running result at the end of the current loop.

Parameters **id** – Result ID as an int.

stop_end_of_test(*id*)
Stop a running result at the end of the current test.

Parameters **id** – Result ID as an int.

summary_stats(*id*)
Compute summary stats for a result.

Parameters **id** – Result ID as an int.

Returns *results.SummaryStats* object

Return type *results.SummaryStats*

unpause(*id*)
Unpause a running result.

Parameters **id** – Result ID as an int.

updates(*id*, *update_id=None*)
Get updates of a running result via long-polling. If no updates are available, CDRouter waits up to 10 seconds before sending an empty response.

Parameters

- **id** – Result ID as an int.
- **update_id** – (optional) Update ID as an int.

Returns *results.Update* object

Return type *results.Update*

Result

class cdrouter.results.**Result**(***kwargs*)
Model for CDRouter Results.

Parameters

- **id** – (optional) Result ID as an int.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **result** – (optional) Result as a string.
- **status** – (optional) Status as a string.
- **loops** – (optional) Loop count as an int.

- **tests** – (optional) Test count as an int.
- **passed** – (optional) Passed count as an int.
- **fail** – (optional) Failed count as an int.
- **duration** – (optional) Duration in seconds as an int.
- **size_on_disk** – (optional) Size on disk in bytes as an int.
- **starred** – (optional) Bool *True* if result is starred.
- **archived** – (optional) Bool *True* if result is archived.
- **result_dir** – (optional) Filepath to result directory as a string.
- **agent_name** – (optional) Agent name as a string.
- **package_name** – (optional) Package name as a string.
- **device_name** – (optional) Device name as a string.
- **config_name** – (optional) Config name as a string.
- **package_id** – (optional) Package ID as an int.
- **device_id** – (optional) Device ID as an int.
- **config_id** – (optional) Config ID as an int.
- **user_id** – (optional) User ID as an int.
- **note** – (optional) Note as a string.
- **pause_message** – (optional) Pause message as a string (if currently paused).
- **build_info** – (optional) Build info as a string.
- **tags** – (optional) Tags as a string list.
- **testcases** – (optional) Testcases as a string list.
- **options** – (optional) *results.Options* object

Page

class cdrouter.results.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *results.Result* list
- **links** – *cdrouter.Links* object

TestCount

class cdrouter.results.**TestCount**(***kwargs*)
Model for CDRouter Test Counts.

Parameters

- **name** – (optional) Name as a string.
- **count** – (optional) Count as an int.

TestDuration

class cdrouter.results.**TestDuration**(***kwargs*)
Model for CDRouter Test Durations.

Parameters

- **name** – (optional) Name as a string.
- **duration** – (optional) Duration as an int.

ResultBreakdown

class cdrouter.results.**ResultBreakdown**(***kwargs*)
Model for CDRouter Result Breakdowns.

Parameters

- **passed** – (optional) Pass count as an int.
- **failed** – (optional) Fail count as an int.
- **skipped** – (optional) Skipped count as an int.

TimeBreakdown

class cdrouter.results.**TimeBreakdown**(***kwargs*)
Model for CDRouter Time Breakdowns.

Parameters

- **passed** – (optional) Pass duration as an int.
- **failed** – (optional) Fail duration as an int.

SetStats

class cdrouter.results.**SetStats**(***kwargs*)
Model for CDRouter Result Set Stats.

Parameters

- **frequent_failures** – (optional) *results.TestCount* list
- **longest_tests** – (optional) *results.TestDuration* list
- **result_breakdown** – (optional) *results.ResultBreakdown* object
- **time_breakdown** – (optional) *results.TimeBreakdown* object

TestResultBreakdown

class cdrouter.results.**TestResultBreakdown**(***kwargs*)
Model for CDRouter TestResult Breakdowns.

Parameters

- **failed_at_least_once** – (optional) *results.TestCount* list

- **passed_every_time** – (optional) *results.TestCount* list

Progress

class cdrouter.results.**Progress**(***kwargs*)
Model for CDRouter Result Progress.

Parameters

- **finished** – (optional) Finished count as an int.
- **total** – (optional) Total count as an int.
- **progress** – (optional) Progress as an int.
- **unit** – (optional) Unit as a string.

SingleStats

class cdrouter.results.**SingleStats**(***kwargs*)
Model for CDRouter Single Results Stats.

Parameters

- **result_breakdown** – (optional) *results.ResultBreakdown* object
- **progress** – (optional) *results.Progress* object

SummaryStats

class cdrouter.results.**SummaryStats**(***kwargs*)
Model for CDRouter Summary Results Stats.

Parameters

- **result_breakdown** – (optional) *results.ResultBreakdown* object
- **test_summaries** – (optional) *testresults.TestResult* list

PackageCount

class cdrouter.results.**PackageCount**(***kwargs*)
Model for CDRouter Package Counts.

Parameters

- **package_name** – (optional) Package name as a string.
- **count** – (optional) Count as an int.

DeviceCount

class cdrouter.results.**DeviceCount**(***kwargs*)
Model for CDRouter Device Counts.

Parameters

- **device_name** – (optional) Device name as a string.

- **count** – (optional) Count as an int.

AllStats

class cdrouter.results.**AllStats**(***kwargs*)
Model for CDRouter All Results Stats.

Parameters

- **frequent_packages** – (optional) *results.PackageCount* list
- **package_names** – (optional) *results.PackageCount* list
- **frequent_devices** – (optional) *results.DeviceCount* list
- **device_names** – (optional) *results.DeviceCount* list

Metric

class cdrouter.results.**Metric**(***kwargs*)
Model for CDRouter Metrics.

Parameters

- **log_file** – (optional) Filepath to logfile as a string.
- **timestamp** – (optional) Timestamp for metric as a *DateTime*.
- **metric** – (optional) Metric name as a string.
- **value** – (optional) Metric value as a float.
- **units** – (optional) Metric units as a string.
- **result** – (optional) Metric result as a string.
- **interface_1** – (optional) First interface as a string.
- **interface_2** – (optional) Second interface as a string.
- **streams** – (optional) Stream count as an int.

LogDirFile

class cdrouter.results.**LogDirFile**(***kwargs*)
Model for CDRouter Logdir Files.

Parameters

- **name** – (optional) Name as a string.
- **size** – (optional) Filesize as an int.
- **modified** – (optional) Last-updated time as a *DateTime*.

Options

class cdrouter.results.**Options**(***kwargs*)
Model for CDRouter Result Options.

Parameters

- **tags** – (optional) Tags as string list.
- **skip_tests** – (optional) Tests to skip as string list.
- **begin_at** – (optional) Test name to begin testing at as string.
- **end_at** – (optional) Test name to end testing at as string.
- **extra_cli_args** – (optional) Extra *cdrouter-cli* arguments as string.

Update

class cdrouter.results.**Update**(***kwargs*)
Model for CDRouter Result Update.

Parameters

- **id** – (optional) Update ID as an int.
- **timestamp** – (optional) System time as *DateTime*.
- **progress** – (optional) *results.Progress* object
- **running** – (optional) *testresults.TestResult* object
- **updates** – (optional) *results.Result* and *testresults.TestResult* list

Result

class cdrouter.results.**Result**(***kwargs*)
Model for CDRouter Results.

Parameters

- **id** – (optional) Result ID as an int.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **result** – (optional) Result as a string.
- **status** – (optional) Status as a string.
- **loops** – (optional) Loop count as an int.
- **tests** – (optional) Test count as an int.
- **passed** – (optional) Passed count as an int.
- **fail** – (optional) Failed count as an int.
- **duration** – (optional) Duration in seconds as an int.
- **size_on_disk** – (optional) Size on disk in bytes as an int.
- **starred** – (optional) Bool *True* if result is starred.

- **archived** – (optional) Bool *True* if result is archived.
- **result_dir** – (optional) Filepath to result directory as a string.
- **agent_name** – (optional) Agent name as a string.
- **package_name** – (optional) Package name as a string.
- **device_name** – (optional) Device name as a string.
- **config_name** – (optional) Config name as a string.
- **package_id** – (optional) Package ID as an int.
- **device_id** – (optional) Device ID as an int.
- **config_id** – (optional) Config ID as an int.
- **user_id** – (optional) User ID as an int.
- **note** – (optional) Note as a string.
- **pause_message** – (optional) Pause message as a string (if currently paused).
- **build_info** – (optional) Build info as a string.
- **tags** – (optional) Tags as a string list.
- **testcases** – (optional) Testcases as a string list.
- **options** – (optional) *results.Options* object

TestResults

TestResultsService

class cdrouter.testresults.**TestResultsService**(*service*)

Service for accessing CDRouter TestResults.

edit(*resource*)

Edit a test result.

Parameters **resource** – *testresults.TestResult* object

Returns *testresults.TestResult* object

Return type *testresults.TestResult*

get(*id*, *seq*)

Get a test result.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.

Returns *testresults.TestResult* object

Return type *testresults.TestResult*

get_log_plaintext(*id*, *seq*)

Get a test result's log as plaintext.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.

Return type string

iter_list(*id*, **args*, ***kwargs*)
Get a list of test results. Whereas `list` fetches a single page of test results according to its `limit` and `page` arguments, `iter_list` returns all test results by internally making successive calls to `list`.

Parameters

- **id** – Result ID as an int.
- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *testresults.TestResult* list

iter_list_log(*id*, *seq*, **args*, ***kwargs*)
Get a test result's log. Whereas `list_log` fetches a single range of log lines according to its `limit` and `offset` arguments, `iter_list_log` returns all log lines by internally making successive calls to `list_log`.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **args** – Arguments that `list_log` takes.
- **kwargs** – Optional arguments that `list_log` takes.

Returns *testresults.Line* list

list(*id*, *filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of test results.

Parameters

- **id** – Result ID as an int.
- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *testresults.Page* object

list_csv(*id*, *filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of test results as CSV.

Parameters

- **id** – Result ID as an int.
- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Return type string

list_log(*id*, *seq*, *offset=None*, *limit=None*, *filter=None*, *packets=None*, *timestamp_format=None*)
Get a test result's log.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **offset** – (optional) Offset within logfile to get.
- **limit** – (optional) Limit returned list length.
- **filter** – (optional) Filters to apply as a string list.
- **packets** – (optional) Set to bool *False* to omit packet logs.
- **timestamp_format** – (optional) Timestamp format to use, must be string *long* or *short*.

Returns *testresults.Log* object

Return type *testresults.Log*

TestResult

class cdrouter.testresults.**TestResult**(***kwargs*)
Model for CDRouter TestResults.

Parameters

- **id** – (optional) Result ID as an int.
- **seq** – (optional) TestResult sequence ID as an int.
- **loop** – (optional) Loop number as an int.
- **result** – (optional) Test result as string.
- **retries** – (optional) Retry count as int.
- **started** – (optional) Test start time as *DateTime*.
- **duration** – (optional) Test duration as int.
- **flagged** – (optional) *True* if test is flagged.
- **name** – (optional) Test name as string.
- **description** – (optional) Test description as string.
- **skip_name** – (optional) Skip name for TestResult as string.
- **skip_reason** – (optional) Skip reason for TestResult as string.
- **log** – (optional) Logfile path for TestResult as string.
- **note** – (optional) Note for TestResult as string.

Page

class cdrouter.testresults.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *testresults.TestResult* list
- **links** – *cdrouter.Links* object

Summary

class cdrouter.testresults.**Summary**(***kwargs*)
Model for CDRouter Log Section Summaries.

Parameters

- **errors** – (optional) Error log count as int.
- **fails** – (optional) Fail log count as int.
- **passes** – (optional) Pass log count as int.
- **warning** – (optional) Warning log count as int.

Line

class cdrouter.testresults.**Line**(***kwargs*)
Model for CDRouter Log Lines.

Parameters

- **raw** – (optional) Raw log text as string.
- **line** – (optional) Line number as int.
- **header** – (optional) *True* if line is a header.
- **section** – (optional) *True* if line is a section.
- **prefix** – (optional) Log prefix as string.
- **name** – (optional) Log stack name as string.
- **timestamp** – (optional) Log timestamp as string.
- **message** – (optional) Log message as string.
- **interface** – (optional) Log interface as string (if packet log).
- **packet** – (optional) Log frame number as an int (if packet log).
- **src** – (optional) Log source address as string (if packet log).
- **dst** – (optional) Log destination address as string (if packet log).
- **proto** – (optional) Log protocol name as string (if packet log).
- **info** – (optional) Log protocol summary as string (if packet log).
- **summary** – (optional) *testresults.Summary* object (if section log)

Log

class cdrouter.testresults.**Log**(***kwargs*)
Model for CDRouter Log.

Parameters

- **offset** – (optional) Zero-based offset in logfile as int.
- **limit** – (optional) Limit as int.
- **lines** – (optional) *testresults.Line* list
- **total** – (optional) Total line count as int.

Annotations

AnnotationsService

class cdrouter.annotations.**AnnotationsService**(*service*)
Service for accessing CDRouter Annotations.

create(*id*, *seq*, *resource*)
Create an annotation.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *annotations.Annotation* object

Returns *annotations.Annotation* object

Return type *annotations.Annotation*

create_or_edit(*id*, *seq*, *resource*)
Create or edit an annotation.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *annotations.Annotation* object

Returns *annotations.Annotation* object

Return type *annotations.Annotation*

delete(*id*, *seq*, *line*)
Delete an annotation.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **line** – Line number in TestResult's logfile as an int.

edit(*id*, *seq*, *resource*)
Edit an annotation.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *annotations.Annotation* object

Returns *annotations.Annotation* object

Return type *annotations.Annotation*

get(*id*, *seq*, *line*)
Get an annotation.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **line** – Line number in TestResult's logfile as an int.

Returns *annotations.Annotation* object

Return type *annotations.Annotation*

list(*id*, *seq*)
Get a list of annotations.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.

Returns *annotations.Annotation* list

Annotation

class cdrouter.annotations.**Annotation**(***kwargs*)
Model for CDRouter Annotations.

Parameters

- **id** – (optional) Result ID as an int.
- **seq** – (optional) TestResult sequence ID as an int.
- **line** – (optional) Line number in TestResult's logfile as an int.
- **comment** – (optional) Comment text as string.

Captures

CapturesService

class cdrouter.captures.**CapturesService**(*service*)
Service for accessing CDRouter Captures.

ascii(*id*, *seq*, *intf*, *filter=None*, *frame=None*, *inline=False*)
Get a capture's ASCII (hex dump).

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.
- **filter** – (optional) PCAP filter to apply as string.
- **frame** – (optional) Frame number to decode.
- **inline** – (optional) Use inline version of capture file.

Returns *captures.ASCII* object

Return type *captures.ASCII*

decode(*id*, *seq*, *intf*, *filter=None*, *frame=None*, *inline=False*)
Get a capture's decode.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.
- **filter** – (optional) PCAP filter to apply as string.
- **frame** – (optional) Frame number to decode.
- **inline** – (optional) Use inline version of capture file.

Returns *captures.Decode* object

Return type *captures.Decode*

download(*id*, *seq*, *intf*, *inline=False*)
Download a capture as a PCAP file.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.
- **inline** – (optional) Use inline version of capture file.

Return type tuple *(io.BytesIO, 'filename')*

get(*id*, *seq*, *intf*)
Get a capture.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.

Returns *captures.Capture* object

Return type *captures.Capture*

list(*id*, *seq*)
Get a list of captures.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.

Returns *captures.Capture* list

send_to_cloudshark(*id*, *seq*, *intf*, *inline=False*)
Send a capture to a CloudShark Appliance. Both cloudshark_appliance_url and cloudshark_appliance_token must be properly configured via system preferences.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.
- **inline** – (optional) Use inline version of capture file.

Returns *captures.CloudShark* object

Return type *captures.CloudShark*

summary(*id*, *seq*, *intf*, *filter=None*, *inline=False*)
Get a capture's summary.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **intf** – Interface name as string.
- **filter** – (optional) PCAP filter to apply as string.
- **inline** – (optional) Use inline version of capture file.

Returns *captures.Summary* object

Return type *captures.Summary*

Capture

class cdrouter.captures.**Capture**(***kwargs*)
Model for CDRouter Captures.

Parameters

- **id** – (optional) Result ID as an int.
- **seq** – (optional) TestResult sequence ID as an int.
- **interface** – (optional) Interface name as string.
- **filename** – (optional) Path to capture file as string.

Section

class cdrouter.captures.**Section**(***kwargs*)
Model for CDRouter Capture Sections.

Parameters **value** – (optional) Section value as a string.

Structure

class cdrouter.captures.**Structure**(***kwargs*)
Model for CDRouter Capture Structures.

Parameters **sections** – (optional) *captures.Section* list

SummaryPacket

class cdrouter.captures.**SummaryPacket**(***kwargs*)
Model for CDRouter Capture Summary Packets.

Parameters **sections** – (optional) *captures.Section* list

Summary

class cdrouter.captures.**Summary**(***kwargs*)
Model for CDRouter Capture Summaries.

Parameters

- **structure** – (optional) *captures.Structure* object
- **summaries** – (optional) *captures.SummaryPacket* list

Field

class cdrouter.captures.**Field**(***kwargs*)
Model for CDRouter Capture Fields.

Parameters

- **name** – (optional) Field name as string.
- **show_name** – (optional) Either *true* or *false* as string.
- **hide** – (optional) Either *true* or *false* as string.
- **size** – (optional) Field size as string.
- **pos** – (optional) Field position as string.
- **show** – (optional) Either *true* or *false* as string.
- **fields** – (optional) *captures.Field* list
- **protos** – (optional) *captures.Proto* list

Proto

class cdrouter.captures.**Proto**(***kwargs*)
Model for CDRouter Capture Proto.

Parameters

- **name** – (optional) Field name as string.
- **pos** – (optional) Field position as string.
- **show** – (optional) Either *true* or *false* as string.
- **show_name** – (optional) Either *true* or *false* as string.
- **value** – (optional) Field value as string.
- **size** – (optional) Field size as string.
- **fields** – (optional) *captures.Field* list
- **protos** – (optional) *captures.Proto* list

Packet

class cdrouter.captures.**Packet**(***kwargs*)
Model for CDRouter Capture Packet.

Parameters **protos** – (optional) *captures.Proto* list

Decode

class cdrouter.captures.**Decode**(***kwargs*)
Model for CDRouter Capture Decode.

Parameters **packets** – (optional) *captures.Packet* list

ASCIIByte

class cdrouter.captures.**ASCIIByte**(***kwargs*)
Model for CDRouter Capture ASCII bytes.

Parameters

- **byte** – (optional) Byte value as string.
- **pos** – (optional) Byte position as int.

ASCIILine

class cdrouter.captures.**ASCIILine**(***kwargs*)
Model for CDRouter Capture ASCII lines.

Parameters

- **raw** – (optional) Raw line as string.
- **offset** – (optional) Line offset as string.

- **ascii** – (optional) *captures.ASCIIByte* list
- **hex** – (optional) *captures.ASCIIByte* list

ASCIIFrame

class cdrouter.captures.**ASCIIFrame**(***kwargs*)
Model for CDRouter Capture ASCII frame.

Parameters

- **name** – (optional) Frame name as string.
- **lines** – (optional) *captures.ASCIILine* list

ASCII

class cdrouter.captures.**ASCII**(***kwargs*)
Model for CDRouter Capture ASCII.

Parameters

- **frame** – (optional) *captures.ASCIIFrame* object
- **reassembled** – (optional) *captures.ASCIIFrame* object

CloudShark

class cdrouter.captures.**CloudShark**(***kwargs*)
Model for CDRouter CloudShark uploads.

Parameters **url** – (optional) CloudShark URL as a string.

Highlights

HighlightsService

class cdrouter.highlights.**HighlightsService**(*service*)
Service for accessing CDRouter Highlights.

create(*id*, *seq*, *resource*)
Create a highlight.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *highlights.Highlight* object

Returns *highlights.Highlight* object

Return type *highlights.Highlight*

create_or_edit(*id*, *seq*, *resource*)
Create or edit a highlight.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *highlights.Highlight* object

Returns *highlights.Highlight* object

Return type *highlights.Highlight*

delete(*id*, *seq*, *line*)
Delete a highlight.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **line** – Line number in TestResult's logfile as an int.

edit(*id*, *seq*, *resource*)
Edit a highlight.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **resource** – *highlights.Highlight* object

Returns *highlights.Highlight* object

Return type *highlights.Highlight*

get(*id*, *seq*, *line*)
Get a highlight.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **line** – Line number in TestResult's logfile as an int.

Returns *highlights.Highlight* object

list(*id*, *seq*)
Get a list of highlights.

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.

Returns *highlights.Highlight* list

Highlight

class cdrouter.highlights.**Highlight**(***kwargs*)
Model for CDRouter Highlights.

Parameters

- **id** – (optional) Result ID as an int.
- **seq** – (optional) TestResult sequence ID as an int.
- **line** – (optional) Line number in TestResult's logfile as an int.
- **color** – (optional) Highlight color as string.

Imports

ImportsService

class cdrouter.imports.**ImportsService**(*service*)
Service for accessing CDRouter Imports.

commit(*id*, *impreq*)
Commit a staged import.

Parameters

- **id** – Staged import ID as an int.
- **impreq** – *imports.Request* object

Returns *imports.Request* object

Return type *imports.Request*

delete(*id*)
Delete a staged import.

Parameters **id** – Staged import ID as an int.

get(*id*)
Get a staged import.

Parameters **id** – Staged import ID as an int.

Returns *imports.Import* object

Return type *imports.Import*

get_commit_request(*id*)
Get a commit request for a staged import.

Parameters **id** – Staged import ID as an int.

Returns *imports.Request* object

Return type *imports.Request*

list()
Get a list of staged (in-progress) imports.

Returns *imports.Import* list

stage_import_from_file(*fd*, *filename='upload.gz'*)
Stage an import from a file upload.

Parameters

- **fd** – File-like object to upload.
- **filename** – (optional) Filename to use for import as string.

Returns *imports.Import* object

stage_import_from_filesystem(*filepath*)
Stage an import from a filesystem path.

Parameters **filepath** – Local filesystem path as string.

Returns *imports.Import* object

stage_import_from_url(*url*, *token=None*, *username=None*, *password=None*, *insecure=False*)
Stage an import from a URL to another CDRouter system.

Parameters

- **url** – URL to import as string.
- **token** – (optional) API token to use as string (may be required if importing from a CDRouter 10+ system).
- **username** – (optional) API username to use as string (may be required if importing from a CDRouter 10+ system).
- **password** – (optional) API password to use as string (may be required if importing from a CDRouter 10+ system).
- **insecure** – (optional) Allow insecure HTTPS connections if bool *True*.

Returns *imports.Import* object

Import

class cdrouter.imports.**Import**(***kwargs*)
Model for CDRouter Staged Imports.

Parameters

- **id** – (optional) Staged import ID as an int.
- **user_id** – (optional) User ID as an int.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **archive** – (optional) Path to archive as string.
- **path** – (optional) Local filepath as string.
- **url** – (optional) URL to fetch as string.
- **insecure** – (optional) Allow insecure HTTPS connections if bool *True*.
- **size** – (optional) Size of import as int.

Response

class cdrouter.imports.**Response**(***kwargs*)
Model for CDRouter Import Responses.

Parameters

- **imported** – (optional) Bool *True* if resource imported successfully.
- **id** – (optional) Resource ID as an int.

- **name** – (optional) Resource name as string.
- **message** – (optional) Response message as string.

Resource

class cdrouter.imports.**Resource**(***kwargs*)
Model for CDRouter Import Resources.

Parameters

- **name** – (optional) Set to string to rename resource, leave empty to keep original
- **should_import** – (optional) Bool *True* if resource should be imported.
- **existing_id** – (optional) Contains ID as an int of existing resource which will be overwritten if *should_import* is *True*.
- **response** – (optional) *imports.Response* object

Request

class cdrouter.imports.**Request**(***kwargs*)
Model for CDRouter Import Requests.

Parameters

- **replace_existing** – (optional) Bool *True* if existing resources should be overwritten.
- **configs** – (optional) Dict of strings to *imports.Resource* objects.
- **devices** – (optional) Dict of strings to *imports.Resource* objects.
- **packages** – (optional) Dict of strings to *imports.Resource* objects.
- **results** – (optional) Dict of strings to *imports.Resource* objects.
- **tags** – (optional) Tags to add to imported resources as string list.

Exports

ExportsService

class cdrouter.exports.**ExportsService**(*service*)
Service for accessing CDRouter Exports.

bulk_export(*config_ids=None*, *device_ids=None*, *package_ids=None*, *result_ids=None*, *exclude_captures=False*)
Bulk export a set of configs, devices, packages and results.

Parameters

- **config_ids** – (optional) Int list of config IDs.
- **device_ids** – (optional) Int list of device IDs.
- **package_ids** – (optional) Int list of package IDs.
- **result_ids** – (optional) Int list of result IDs.

- **exclude_captures** – (optional) Exclude capture files if bool *True*.

Return type tuple *(io.BytesIO, 'filename')*

History

HistoryService

class cdrouter.history.**HistoryService**(*service*)
Service for accessing CDRouter History.

iter_list(**args*, ***kwargs*)
Get a list of history entries. Whereas `list` fetches a single page of history entries according to its `limit` and `page` arguments, `iter_list` returns all history entries by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns `historys.History` list

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of history entries.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *history.Page* object

History

class cdrouter.history.**History**(***kwargs*)
Model for CDRouter History entries.

Parameters

- **user_id** – (optional) User ID as an int.
- **created** – (optional) Entry creation time as *DateTime*.
- **resource** – (optional) Resource type as string.
- **id** – (optional) Resource ID as an int.
- **name** – (optional) Resource name as string.
- **action** – (optional) Action name as string.
- **description** – (optional) Resource description as string.

Page

class cdrouter.history.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *history.History* list
- **links** – *cdrouter.Links* object

System

SystemService

class cdrouter.system.**SystemService**(*service*)
Service for accessing CDRouter System.

check_for_lounge_upgrade(*email*, *password*)
Check the CDRouter Support Lounge for eligible upgrades using your Support Lounge email & password.

Parameters

- **email** – CDRouter Support Lounge email as a string.
- **password** – CDRouter Support Lounge password as a string.

Returns *system.Release* object

Return type *system.Release*

diagnostics()
Get system diagnostics from cdrouter-diag output.

Return type string

edit_preferences(*resource*)
Edit preferences in /usr/cdrouter-data/etc/config.yml.

Parameters **resource** – *system.Preferences* object

Returns *system.Preferences* object

Return type *system.Preferences*

get_preferences()
Get preferences from /usr/cdrouter-data/etc/config.yml.

Returns *system.Preferences* object

Return type *system.Preferences*

hostname()
Get system hostname.

Return type string

info()
Get system info from cdrouter-cli -info output.

Return type string

interfaces(*addresses=False*)
Get system interfaces.

Parameters **addresses** – (optional) If bool *True*, include interface addresses.

Returns *system.Interface* list

latest_lounge_release()
Get the latest release of CDRouter from the CDRouter Support Lounge.

Returns *system.ReleaseLatest* object

Return type *system.ReleaseLatest*

live()
Get CDRouter Live info from cdrouter-cli -live output.

Return type string

lounge_update_license()
Download & install a license for your CDRouter system from the CDRouter Support Lounge.

Returns *system.Upgrade* object

Return type *system.Upgrade*

lounge_upgrade(*email*, *password*, *release_id*)
Download & install an upgrade from the CDRouter Support Lounge using your Support Lounge email & password. Please note that any running tests will be stopped.

Parameters

- **email** – CDRouter Support Lounge email as a string.
- **password** – CDRouter Support Lounge password as a string.
- **release_id** – Release ID as an int.

Returns *system.Upgrade* object

Return type *system.Upgrade*

manual_update_license(*fd*, *filename='cdrouter.lic'*)
Update the license on your CDRouter system manually by uploading a .lic license from the CDRouter Support Lounge.

Parameters

- **fd** – File-like object to upload.
- **filename** – (optional) Filename to use for license as string.

Returns *system.Upgrade* object

Return type *system.Upgrade*

manual_upgrade(*fd*, *filename='cdrouter.bin'*)
Upgrade CDRouter manually by uploading a .bin installer from the CDRouter Support Lounge. Please note that any running tests will be stopped.

Parameters

- **fd** – File-like object to upload.
- **filename** – (optional) Filename to use for installer as string.

Returns *system.Upgrade* object

Return type *system.Upgrade*

restart()
Restart CDRouter web interface. Please note that any running tests will be stopped.

space()
Get system disk space usage.

Returns *system.Space* object

Return type *system.Space*

time()
Get system time.

Return type string

Version

class cdrouter.system.**Version**(***kwargs*)
Model for CDRouter Release Versions.

Parameters

- **raw** – (optional) Raw version as string.
- **major** – (optional) Major version as int.
- **minor** – (optional) Minor version as int.
- **build** – (optional) Build version as int.

ReleaseLatest

class cdrouter.system.**ReleaseLatest**(***kwargs*)
Model for CDRouter Latest Releases.

Parameters

- **latest** – (optional) *system.Version* object.
- **current** – (optional) *system.Version* object.
- **newer** – (optional) Bool *True* if newer release available.

Testsuite

class cdrouter.system.**Testsuite**(***kwargs*)
Model for CDRouter Testsuite names.

Parameters

- **shortname** – (optional) Brief testsuite name as string.
- **name** – (optional) Full testsuite name as string.

Release

class cdrouter.system.**Release**(***kwargs*)
Model for CDRouter Releases.

Parameters

- **build_date** – (optional) Build date as string.
- **filename** – (optional) Installer filename as string.
- **version** – (optional) *system.Version* object
- **id** – (optional) Release ID as int.
- **article_id** – (optional) Release article ID as int.
- **testsuite** – (optional) *system.Testsuite* object

Upgrade

class cdrouter.system.**Upgrade**(***kwargs*)
Model for CDRouter Upgrades.

Parameters

- **success** – (optional) Bool *True* if successful.
- **installer** – (optional) Installer filename as string.
- **output** – (optional) Output as string.
- **error** – (optional) Error output as string.

InterfaceFlags

class cdrouter.system.**InterfaceFlags**(***kwargs*)
Model for CDRouter Interface Flags.

Parameters

- **up** – (optional) Bool *True* if interface is up.
- **broadcast** – (optional) Bool *True* if interface is broadcasting.
- **loopback** – (optional) Bool *True* if interface is a loopback interface.
- **point_to_point** – (optional) Bool *True* if interface is a point-to-point interface.
- **multicast** – (optional) Bool *True* if interface is multicast.

InterfaceAddr

class cdrouter.system.**InterfaceAddr**(***kwargs*)
Model for CDRouter Interface Addresses.

Parameters

- **network** – (optional) Interface network as a string.
- **address** – (optional) Interface address as a string.

Interface

class cdrouter.system.**Interface**(***kwargs*)
Model for CDRouter Interfaces.

Parameters

- **index** – (optional) Interface index as an int.
- **mtu** – (optional) Interface MTU as an int.
- **name** – (optional) Interface name as a string.
- **hardware_addr** – (optional) Interface hardware address as a string.
- **flags** – (optional) *system.InterfaceFlags* object
- **addresses** – (optional) *system.InterfaceAddr* list
- **multicast_addresses** – (optional) *system.InterfaceAddr* list

Preferences

class cdrouter.system.**Preferences**(***kwargs*)
Model for CDRouter Preferences.

Parameters

- **automatic_logic** – (optional) Bool *True* if Automatic Login is enabled.
- **cloudshark_appliance_autotags** – (optional) CloudShark Appliance autotags as a string.
- **cloudshark_appliance_insecure** – (optional) String *yes* if insecure CloudShark Appliance URLs are allowed.
- **cloudshark_appliance_password** – (optional) CloudShark Appliance password as a string.
- **cloudshark_appliance_tags** – (optional) CloudShark Appliance tags as a string.
- **cloudshark_appliance_token** – (optional) CloudShark Appliance API token as a string.
- **cloudshark_appliance_url** – (optional) CloudShark Appliance URL as a string.
- **cloudshark_appliance_username** – (optional) CloudShark Appliance username as a string.
- **hostname** – (optional) CDRouter system's hostname as a string.
- **migrated** – (optional) Migrated resources as a string list.
- **port** – (optional) CDRouter's HTTP port as an int.
- **https** – (optional) CDRouter's HTTPS port as an int.
- **force_https** – (optional) If string *yes*, redirect HTTP connections to CDRouter to HTTPS.
- **use_cloudshark** – (optional) If string *yes*, use CloudShark Appliance for capture viewing.
- **log_timestamp_format** – (optional) Log timestamp format, must be string *short* or *long*.

- **editor_keymap** – (optional) Config editor keymap, must be string *default*, *emacs* or *vim*.
- **lounge_url** – (optional) CDRouter Support Lounge URL as a string.
- **lounge_insecure** – (optional) If bool *True*, allow insecure HTTPS connections to the CDRouter Support Lounge.

Space

class cdrouter.system.**Space**(***kwargs*)
Model for CDRouter Disk Space Usage.

Parameters

- **avail** – (optional) Available disk space as an int.
- **path** – (optional) Path to CDRouter's data directory as a string.
- **pcent** – (optional) Percentage of disk space used as an int.
- **size** – (optional) Total disk space as an int.
- **unit** – (optional) Disk space units as a string, default is 'bytes'.
- **used** – (optional) Used disk space as an int.

Tags

TagsService

class cdrouter.tags.**TagsService**(*service*)
Service for accessing CDRouter Tags.

delete(*name*)
Delete a tag.

Parameters **name** – Tag name as string.

edit(*resource*)
Edit a tag.

Parameters **resource** – *tags.Tag* object

Returns *tags.Tag* object

Return type *tags.Tag*

get(*name*)
Get a tag.

Parameters **name** – Tag name as string.

Returns *tags.Tag* object

Return type *tags.Tag*

list(*resource=None*, *sort=None*)
Get a list of tags.

Parameters

- **resource** – (optional) Restrict to given resource type as string.

- **sort** – (optional) Sort fields to apply as string list.

Returns *tags.Tag* list

Tag

class cdrouter.tags.**Tag**(***kwargs*)
Model for CDRouter Tags.

Parameters

- **resource** – (optional) Resource type as string.
- **id** – (optional) Resource ID as an int.
- **name** – (optional) Resource name as string.
- **count** – (optional) Resource tag count as an int.
- **tags** – (optional) Resource tags as a string list.
- **configs** – (optional) *tags.ResourceTags* list
- **devices** – (optional) *tags.ResourceTags* list
- **packages** – (optional) *tags.ResourceTags* list
- **results** – (optional) *tags.ResourceTags* list

ResourceTags

class cdrouter.tags.**ResourceTags**(***kwargs*)
Model for CDRouter Resource Tags.

Parameters

- **id** – (optional) Resource ID as an int.
- **name** – (optional) Resource name as string.
- **tags** – (optional) Resource tags as a string list.

Testsuites

TestsuitesService

class cdrouter.testsuites.**TestsuitesService**(*service*)
Service for accessing CDRouter Testsuites.

get_error(*name*)
Get an error.

Parameters **name** – Error name as string.

Returns *testsuites.Error* object

Return type *testsuites.Error*

get_group(*name*)
Get a group.

Parameters **name** – Group name as string.

Returns *testsuites.Group* object

Return type *testsuites.Group*

get_label(*name*)
Get a label.

Parameters **name** – Label name as string.

Returns *testsuites.Label* object

Return type *testsuites.Label*

get_module(*name*)
Get a module.

Parameters **name** – Module name as string.

Returns *testsuites.Module* object

Return type *testsuites.Module*

get_test(*name*)
Get a test.

Parameters **name** – Test name as string.

Returns *testsuites.Test* object

Return type *testsuites.Test*

get_testvar(*name*)
Get a testvar.

Parameters **name** – Testvar name as string.

Returns *testsuites.Testvar* object

Return type *testsuites.Testvar*

info()
Get testsuite info.

Returns *testsuites.Info* object

Return type *testsuites.Info*

list_errors(*filter=None*, *type=None*, *sort=None*)
Get a list of errors.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Error* list

list_groups(*filter=None*, *type=None*, *sort=None*)
Get a list of groups.

Parameters

- **filter** – (optional) Filters to apply as a string list.

- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Group* list

list_labels (*filter=None*, *type=None*, *sort=None*)
Get a list of labels.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Label* list

list_modules (*filter=None*, *type=None*, *sort=None*)
Get a list of modules.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Module* list

list_tests (*filter=None*, *type=None*, *sort=None*)
Get a list of tests.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Test* list

list_testvars (*filter=None*, *type=None*, *sort=None*)
Get a list of testvars.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.

Returns *testsuites.Testvar* list

search (*query*)
Perform full text search of testsuite.

Parameters **query** – Search query as a string.

Returns *testsuites.Search* object

Return type *testsuites.Search*

update ()
Update testsuite info.

Info

class cdrouter.testsuites.**Info**(***kwargs*)
Model for CDRouter Testsuite Info.

Parameters

- **build_info** – (optional) CDRouter build info as string.
- **copyright** – (optional) CDRouter copyright info as string.
- **customer** – (optional) Customer name from license as string.
- **lifetime** – (optional) License lifetime as string.
- **os** – (optional) OS name as string.
- **serial_number** – (optional) NTA serial number as string.
- **system_id** – (optional) CDRouter system ID as string.
- **testsuite** – (optional) CDRouter testsuite name as string.
- **release** – (optional) CDRouter release as string.
- **addons** – (optional) Enabled CDRouter addons as string list.
- **all_addons** – (optional) All CDRouter addons as string list.

Group

class cdrouter.testsuites.**Group**(***kwargs*)
Model for CDRouter Groups.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as string.
- **index** – (optional) Index as int.
- **test_count** – (optional) Test count as int.
- **modules** – (optional) Modules as string list.

Module

class cdrouter.testsuites.**Module**(***kwargs*)
Model for CDRouter Modules.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as string.
- **index** – (optional) Index as int.
- **group** – (optional) Group name as string.
- **description** – (optional) Module description as string.
- **test_count** – (optional) Test count as int.

- **tests** – (optional) Tests as string list.
- **labels** – (optional) Labels as string list.
- **aliases** – (optional) Aliases as string list.

Test

class cdrouter.testsuites.**Test**(***kwargs*)
Model for CDRouter Tests.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as string.
- **index** – (optional) Index as int.
- **group** – (optional) Group name as string.
- **module** – (optional) Module name as string.
- **synopsis** – (optional) Test synopsis as string.
- **description** – (optional) Test description as string.
- **labels** – (optional) Labels as string list.
- **aliases** – (optional) Aliases as string list.
- **testvars** – (optional) Testvars as string list.
- **skip_name** – (optional) Skip name as string.
- **skip_reason** – (optional) Skip reason as string.

Label

class cdrouter.testsuites.**Label**(***kwargs*)
Model for CDRouter Skip Labels.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as string.
- **index** – (optional) Index as int.
- **reason** – (optional) Skip reason as string.
- **description** – (optional) Description as string.
- **modules** – (optional) Modules as string list.
- **tests** – (optional) Tests as string list.

Error

class cdrouter.testsuites.**Error**(***kwargs*)
Model for CDRouter Start Errors.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as string.
- **index** – (optional) Index as int.
- **description** – (optional) Description as string.

Testvar

class cdrouter.testsuites.**Testvar**(***kwargs*)
Model for CDRouter Start Errors.

Parameters

- **id** – (optional) Testsuite ID as an int.
- **name** – (optional) Name as a string.
- **index** – (optional) Index as an int.
- **humanname** – (optional) Human-readable name as a string.
- **display** – (optional) Display name as a string.
- **dataclass** – (optional) Data-type as a string.
- **addedin** – (optional) Added-in info as a string.
- **deprecatedin** – (optional) Deprecated-in info as a string.
- **obsoletedin** – (optional) Obsoleted-in info as a string.
- **min** – (optional) Minimum value as a string.
- **max** – (optional) Maximum value as a string.
- **length** – (optional) Required length as a string.
- **description** – (optional) Description as a string.
- **default** – (optional) Default value as a string.
- **defaultdisabled** – (optional) Bool *True* if testvar has no default value.
- **dyndefault** – (optional) Bool *True* if testvar's default value is dynamically calculated.
- **keywords** – (optional) Accepted values as a string list.
- **alsoaccept** – (optional) Also-accepted values as a string list.
- **wildcard** – (optional) Bool *True* if testvar is a wildcard testvar.
- **instances** – (optional) Number of allowed wildcard instances as an int.
- **parent** – (optional) Parent wildcard testvar name as a string.
- **children** – (optional) Child testvars as a string list.
- **tests** – (optional) Test names as a string list.

Search

class cdrouter.testsuites.**Search**(***kwargs*)
Model for CDRouter Testsuite Searches.

Parameters

- **addons** – (optional) *testsuites.Group* list
- **modules** – (optional) *testsuites.Module* list
- **tests** – (optional) *testsuites.Test* list
- **reasons** – (optional) *testsuites.Label* list
- **errors** – (optional) *testsuites.Error* list
- **testvars** – (optional) *testsuites.Testvar* list

Users

UsersService

class cdrouter.users.**UsersService**(*service*)
Service for accessing CDRouter Users.

bulk_copy(*ids*)
Bulk copy a set of users.

Parameters **ids** – Int list of user IDs.

Returns *users.User* list

bulk_delete(*ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk delete a set of users.

Parameters

- **ids** – (optional) Int list of user IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

bulk_edit(*_fields*, *ids=None*, *filter=None*, *type=None*, *all=False*)
Bulk edit a set of users.

Parameters

- **_fields** – *users.User* object
- **ids** – (optional) Int list of user IDs.
- **filter** – (optional) String list of filters.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

change_password(*id*, *new*, *old=None*, *change_token=True*)
Change a user's password.

Parameters

- **id** – User ID as an int.
- **new** – New password as string.
- **old** – (optional) Old password as string (required if performing action as non-admin).
- **change_token** – (optional) If bool *True*, also generate a new API token for user.

Returns *users.User* object

Return type *users.User*

change_token(*id*)
Change a user's token.

Parameters **id** – User ID as an int.

Returns *users.User* object

Return type *users.User*

create(*resource*)
Create a new user.

Parameters **resource** – *users.User* object

Returns *users.User* object

Return type *users.User*

delete(*id*)
Delete a user.

Parameters **id** – User ID as an int.

edit(*resource*)
Edit a user.

Parameters **resource** – *users.User* object

Returns *users.User* object

Return type *users.User*

get(*id*)
Get a user.

Parameters **id** – User ID as an int.

Returns *users.User* object

Return type *users.User*

get_by_name(*name*)
Get a user by name.

Parameters **name** – User name as string.

Returns *users.User* object

Return type *users.User*

iter_list(**args*, ***kwargs*)
Get a list of users. Whereas `list` fetches a single page of users according to its `limit` and `page` arguments, `iter_list` returns all users by internally making successive calls to `list`.

Parameters

- **args** – Arguments that `list` takes.
- **kwargs** – Optional arguments that `list` takes.

Returns *users.User* list

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
Get a list of users.

Parameters

- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **sort** – (optional) Sort fields to apply as string list.
- **limit** – (optional) Limit returned list length.
- **page** – (optional) Page to return.

Returns *users.Page* object

User

class cdrouter.users.**User**(***kwargs*)
Model for CDRouter Users.

Parameters

- **id** – (optional) User ID as an int.
- **admin** – (optional) Bool *True* if user is an administrator.
- **disabled** – (optional) Bool *True* if user is disabled.
- **name** – (optional) User name as string.
- **description** – (optional) User description as string.
- **created** – (optional) User creation time as *DateTime*.
- **updated** – (optional) User last-updated time as *DateTime*.
- **token** – (optional) User's API token as string.

Page

class cdrouter.users.**Page**
Named tuple for a page of list response data.

Parameters

- **data** – *users.User* list
- **links** – *cdrouter.Links* object

Python Module Index

C

cdrouter, 3

Index

A

all_stats() (cdrouter.results.ResultsService method), 26
AllStats (class in cdrouter.results), 34
Analysis (class in cdrouter.packages), 25
analyze() (cdrouter.packages.PackagesService method), 22
Annotation (class in cdrouter.annotations), 41
annotations (cdrouter.CDRouter attribute), 4
AnnotationsService (class in cdrouter.annotations), 40
ASCII (class in cdrouter.captures), 46
ascii() (cdrouter.captures.CapturesService method), 41
ASCIIByte (class in cdrouter.captures), 45
ASCIIFrame (class in cdrouter.captures), 46
ASCIILine (class in cdrouter.captures), 45
Attachment (class in cdrouter.attachments), 19
attachments (cdrouter.CDRouter attribute), 4
AttachmentsService (class in cdrouter.attachments), 17
authenticate() (cdrouter.CDRouter method), 4

B

bulk_copy() (cdrouter.configs.ConfigsService method), 7
bulk_copy() (cdrouter.devices.DevicesService method), 13
bulk_copy() (cdrouter.packages.PackagesService method), 22
bulk_copy() (cdrouter.results.ResultsService method), 26
bulk_copy() (cdrouter.users.UsersService method), 64
bulk_delete() (cdrouter.configs.ConfigsService method), 8
bulk_delete() (cdrouter.devices.DevicesService method), 13
bulk_delete() (cdrouter.jobs.JobsService method), 19
bulk_delete() (cdrouter.packages.PackagesService method), 22
bulk_delete() (cdrouter.results.ResultsService method), 26
bulk_delete() (cdrouter.users.UsersService method), 64
bulk_edit() (cdrouter.configs.ConfigsService method), 8
bulk_edit() (cdrouter.devices.DevicesService method), 13
bulk_edit() (cdrouter.packages.PackagesService method), 22
bulk_edit() (cdrouter.results.ResultsService method), 26
bulk_edit() (cdrouter.users.UsersService method), 64
bulk_edit_testvars() (cdrouter.configs.ConfigsService method), 8
bulk_export() (cdrouter.configs.ConfigsService method), 8
bulk_export() (cdrouter.devices.DevicesService method), 13
bulk_export() (cdrouter.exports.ExportsService method), 50
bulk_export() (cdrouter.packages.PackagesService method), 22
bulk_export() (cdrouter.results.ResultsService method), 26
bulk_launch() (cdrouter.jobs.JobsService method), 20

C

Capture (class in cdrouter.captures), 43
captures (cdrouter.CDRouter attribute), 4
CapturesService (class in cdrouter.captures), 41
CDRouter (class in cdrouter), 3
cdrouter (module), 3
CDRouterError (class in cdrouter.cdrouter), 5
change_password() (cdrouter.users.UsersService method), 64
change_token() (cdrouter.users.UsersService method), 65
check_config() (cdrouter.configs.ConfigsService method), 8
check_for_lounge_upgrade() (cdrouter.system.SystemService method), 52
CheckConfig (class in cdrouter.configs), 12
CloudShark (class in cdrouter.captures), 46
commit() (cdrouter.imports.ImportsService method), 48
Config (class in cdrouter.configs), 11
ConfigError (class in cdrouter.configs), 12
configs (cdrouter.CDRouter attribute), 4
ConfigsService (class in cdrouter.configs), 7

connect() (cdrouter.devices.DevicesService method), 13
Connection (class in cdrouter.devices), 17
contained_by() (cdrouter.filters.Field method), 6
contains() (cdrouter.filters.Field method), 6
create() (cdrouter.annotations.AnnotationsService method), 40
create() (cdrouter.attachments.AttachmentsService method), 17
create() (cdrouter.configs.ConfigsService method), 8
create() (cdrouter.devices.DevicesService method), 13
create() (cdrouter.highlights.HighlightsService method), 46
create() (cdrouter.packages.PackagesService method), 22
create() (cdrouter.users.UsersService method), 65
create_or_edit() (cdrouter.annotations.AnnotationsService method), 40
create_or_edit() (cdrouter.highlights.HighlightsService method), 46

D

Decode (class in cdrouter.captures), 45
decode() (cdrouter.captures.CapturesService method), 42
delete() (cdrouter.annotations.AnnotationsService method), 40
delete() (cdrouter.attachments.AttachmentsService method), 17
delete() (cdrouter.configs.ConfigsService method), 8
delete() (cdrouter.devices.DevicesService method), 14
delete() (cdrouter.highlights.HighlightsService method), 47
delete() (cdrouter.imports.ImportsService method), 48
delete() (cdrouter.jobs.JobsService method), 20
delete() (cdrouter.packages.PackagesService method), 23
delete() (cdrouter.results.ResultsService method), 26
delete() (cdrouter.tags.TagsService method), 57
delete() (cdrouter.users.UsersService method), 65
delete_testvar() (cdrouter.configs.ConfigsService method), 9
Device (class in cdrouter.devices), 15
DeviceCount (class in cdrouter.results), 33
devices (cdrouter.CDRouter attribute), 4
DevicesService (class in cdrouter.devices), 13
diagnostics() (cdrouter.system.SystemService method), 52
diff_stats() (cdrouter.results.ResultsService method), 26
disconnect() (cdrouter.devices.DevicesService method), 14
download() (cdrouter.attachments.AttachmentsService method), 18
download() (cdrouter.captures.CapturesService method), 42
download_logdir_archive() (cdrouter.results.ResultsService method), 27

E

edit() (cdrouter.annotations.AnnotationsService method), 40
edit() (cdrouter.attachments.AttachmentsService method), 18
edit() (cdrouter.configs.ConfigsService method), 9
edit() (cdrouter.devices.DevicesService method), 14
edit() (cdrouter.highlights.HighlightsService method), 47
edit() (cdrouter.jobs.JobsService method), 20
edit() (cdrouter.packages.PackagesService method), 23
edit() (cdrouter.results.ResultsService method), 27
edit() (cdrouter.tags.TagsService method), 57
edit() (cdrouter.testresults.TestResultsService method), 36
edit() (cdrouter.users.UsersService method), 65
edit_preferences() (cdrouter.system.SystemService method), 52
edit_shares() (cdrouter.configs.ConfigsService method), 9
edit_shares() (cdrouter.devices.DevicesService method), 14
edit_shares() (cdrouter.packages.PackagesService method), 23
edit_shares() (cdrouter.results.ResultsService method), 27
edit_testvar() (cdrouter.configs.ConfigsService method), 9
eq() (cdrouter.filters.Field method), 6
Error (class in cdrouter.testsuites), 63
export() (cdrouter.configs.ConfigsService method), 9
export() (cdrouter.devices.DevicesService method), 14
export() (cdrouter.packages.PackagesService method), 23
export() (cdrouter.results.ResultsService method), 27
exports (cdrouter.CDRouter attribute), 4
ExportsService (class in cdrouter.exports), 50

F

Field (class in cdrouter.captures), 44
Field (class in cdrouter.filters), 5

G

ge() (cdrouter.filters.Field method), 6
get() (cdrouter.annotations.AnnotationsService method), 41
get() (cdrouter.attachments.AttachmentsService method), 18
get() (cdrouter.captures.CapturesService method), 42
get() (cdrouter.configs.ConfigsService method), 9
get() (cdrouter.devices.DevicesService method), 14
get() (cdrouter.highlights.HighlightsService method), 47
get() (cdrouter.imports.ImportsService method), 48
get() (cdrouter.jobs.JobsService method), 20
get() (cdrouter.packages.PackagesService method), 23
get() (cdrouter.results.ResultsService method), 27

get() (cdrouter.tags.TagsService method), 57
get() (cdrouter.testresults.TestResultsService method), 36
get() (cdrouter.users.UsersService method), 65
get_by_name() (cdrouter.configs.ConfigsService method), 9
get_by_name() (cdrouter.devices.DevicesService method), 14
get_by_name() (cdrouter.packages.PackagesService method), 23
get_by_name() (cdrouter.users.UsersService method), 65
get_commit_request() (cdrouter.imports.ImportsService method), 48
get_connection() (cdrouter.devices.DevicesService method), 14
get_error() (cdrouter.testsuites.TestsuitesService method), 58
get_group() (cdrouter.testsuites.TestsuitesService method), 58
get_label() (cdrouter.testsuites.TestsuitesService method), 59
get_log_plaintext() (cdrouter.testresults.TestResultsService method), 36
get_logdir_file() (cdrouter.results.ResultsService method), 27
get_module() (cdrouter.testsuites.TestsuitesService method), 59
get_networks() (cdrouter.configs.ConfigsService method), 10
get_new() (cdrouter.configs.ConfigsService method), 10
get_plaintext() (cdrouter.configs.ConfigsService method), 10
get_preferences() (cdrouter.system.SystemService method), 52
get_shares() (cdrouter.configs.ConfigsService method), 10
get_shares() (cdrouter.devices.DevicesService method), 15
get_shares() (cdrouter.packages.PackagesService method), 23
get_shares() (cdrouter.results.ResultsService method), 27
get_test() (cdrouter.testsuites.TestsuitesService method), 59
get_test_metric() (cdrouter.results.ResultsService method), 28
get_test_metric_csv() (cdrouter.results.ResultsService method), 28
get_testvar() (cdrouter.configs.ConfigsService method), 10
get_testvar() (cdrouter.testsuites.TestsuitesService method), 59
Group (class in cdrouter.testsuites), 61
gt() (cdrouter.filters.Field method), 6

H

Highlight (class in cdrouter.highlights), 47
highlights (cdrouter.CDRouter attribute), 4
HighlightsService (class in cdrouter.highlights), 46
history (cdrouter.CDRouter attribute), 4
History (class in cdrouter.history), 51
HistoryService (class in cdrouter.history), 51
hostname() (cdrouter.system.SystemService method), 52

I

Import (class in cdrouter.imports), 49
imports (cdrouter.CDRouter attribute), 4
ImportsService (class in cdrouter.imports), 48
Info (class in cdrouter.testsuites), 61
info() (cdrouter.system.SystemService method), 52
info() (cdrouter.testsuites.TestsuitesService method), 59
Interface (class in cdrouter.system), 56
InterfaceAddr (class in cdrouter.system), 55
InterfaceFlags (class in cdrouter.system), 55
interfaces() (cdrouter.system.SystemService method), 52
iter_list() (cdrouter.attachments.AttachmentsService method), 18
iter_list() (cdrouter.configs.ConfigsService method), 10
iter_list() (cdrouter.devices.DevicesService method), 15
iter_list() (cdrouter.history.HistoryService method), 51
iter_list() (cdrouter.jobs.JobsService method), 20
iter_list() (cdrouter.packages.PackagesService method), 23
iter_list() (cdrouter.results.ResultsService method), 28
iter_list() (cdrouter.testresults.TestResultsService method), 37
iter_list() (cdrouter.users.UsersService method), 65
iter_list_log() (cdrouter.testresults.TestResultsService method), 37

J

Job (class in cdrouter.jobs), 21
jobs (cdrouter.CDRouter attribute), 4
JobsService (class in cdrouter.jobs), 19

L

Label (class in cdrouter.testsuites), 62
latest_lounge_release() (cdrouter.system.SystemService method), 53
launch() (cdrouter.jobs.JobsService method), 20
le() (cdrouter.filters.Field method), 6
Line (class in cdrouter.testresults), 39
Links (class in cdrouter.cdrouter), 5
list() (cdrouter.annotations.AnnotationsService method), 41
list() (cdrouter.attachments.AttachmentsService method), 18
list() (cdrouter.captures.CapturesService method), 43

list() (cdrouter.configs.ConfigsService method), 10
list() (cdrouter.devices.DevicesService method), 15
list() (cdrouter.highlights.HighlightsService method), 47
list() (cdrouter.history.HistoryService method), 51
list() (cdrouter.imports.ImportsService method), 48
list() (cdrouter.jobs.JobsService method), 20
list() (cdrouter.packages.PackagesService method), 24
list() (cdrouter.results.ResultsService method), 28
list() (cdrouter.tags.TagsService method), 57
list() (cdrouter.testresults.TestResultsService method), 37
list() (cdrouter.users.UsersService method), 66
list_csv() (cdrouter.results.ResultsService method), 28
list_csv() (cdrouter.testresults.TestResultsService method), 37
list_errors() (cdrouter.testsuites.TestsuitesService method), 59
list_groups() (cdrouter.testsuites.TestsuitesService method), 59
list_labels() (cdrouter.testsuites.TestsuitesService method), 60
list_log() (cdrouter.testresults.TestResultsService method), 38
list_logdir() (cdrouter.results.ResultsService method), 29
list_modules() (cdrouter.testsuites.TestsuitesService method), 60
list_tests() (cdrouter.testsuites.TestsuitesService method), 60
list_testvars() (cdrouter.configs.ConfigsService method), 11
list_testvars() (cdrouter.testsuites.TestsuitesService method), 60
live() (cdrouter.system.SystemService method), 53
Log (class in cdrouter.testresults), 40
LogDirFile (class in cdrouter.results), 34
lounge_update_license() (cdrouter.system.SystemService method), 53
lounge_upgrade() (cdrouter.system.SystemService method), 53
lt() (cdrouter.filters.Field method), 7

M

manual_update_license() (cdrouter.system.SystemService method), 53
manual_upgrade() (cdrouter.system.SystemService method), 53
match() (cdrouter.filters.Field method), 7
Metric (class in cdrouter.results), 34
Module (class in cdrouter.testsuites), 61

N

ne() (cdrouter.filters.Field method), 7
Networks (class in cdrouter.configs), 12
not_() (cdrouter.filters.Field method), 7

O

Options (class in cdrouter.jobs), 21
Options (class in cdrouter.packages), 25
Options (class in cdrouter.results), 35
overlaps() (cdrouter.filters.Field method), 7

P

Package (class in cdrouter.packages), 24
PackageCount (class in cdrouter.results), 33
packages (cdrouter.CDRouter attribute), 4
PackagesService (class in cdrouter.packages), 22
Packet (class in cdrouter.captures), 45
Page (class in cdrouter.attachments), 19
Page (class in cdrouter.configs), 11
Page (class in cdrouter.devices), 17
Page (class in cdrouter.history), 52
Page (class in cdrouter.jobs), 21
Page (class in cdrouter.packages), 25
Page (class in cdrouter.results), 31
Page (class in cdrouter.testresults), 39
Page (class in cdrouter.users), 66
pause() (cdrouter.results.ResultsService method), 29
pause_end_of_loop() (cdrouter.results.ResultsService method), 29
pause_end_of_test() (cdrouter.results.ResultsService method), 29
power_off() (cdrouter.devices.DevicesService method), 15
power_on() (cdrouter.devices.DevicesService method), 15
PowerCmd (class in cdrouter.devices), 17
Preferences (class in cdrouter.system), 56
Progress (class in cdrouter.results), 33
progress_stats() (cdrouter.results.ResultsService method), 29
Proto (class in cdrouter.captures), 45

R

Release (class in cdrouter.system), 55
ReleaseLatest (class in cdrouter.system), 54
Request (class in cdrouter.imports), 50
Resource (class in cdrouter.imports), 50
ResourceTags (class in cdrouter.tags), 58
Response (class in cdrouter.imports), 49
restart() (cdrouter.system.SystemService method), 54
Result (class in cdrouter.results), 30, 35
ResultBreakdown (class in cdrouter.results), 32
results (cdrouter.CDRouter attribute), 4
ResultsService (class in cdrouter.results), 26

S

Search (class in cdrouter.testsuites), 64
search() (cdrouter.testsuites.TestsuitesService method), 60

Section (class in cdrouter.captures), 44
send_to_cloudshark() (cdrouter.captures.CapturesService method), 43
set_stats() (cdrouter.results.ResultsService method), 29
SetStats (class in cdrouter.results), 32
Share (class in cdrouter.cdrouter), 5
single_stats() (cdrouter.results.ResultsService method), 29
SingleStats (class in cdrouter.results), 33
Space (class in cdrouter.system), 57
space() (cdrouter.system.SystemService method), 54
stage_import_from_file() (cdrouter.imports.ImportsService method), 48
stage_import_from_filesystem() (cdrouter.imports.ImportsService method), 49
stage_import_from_url() (cdrouter.imports.ImportsService method), 49
stop() (cdrouter.results.ResultsService method), 30
stop_end_of_loop() (cdrouter.results.ResultsService method), 30
stop_end_of_test() (cdrouter.results.ResultsService method), 30
Structure (class in cdrouter.captures), 44
Summary (class in cdrouter.captures), 44
Summary (class in cdrouter.testresults), 39
summary() (cdrouter.captures.CapturesService method), 43
summary_stats() (cdrouter.results.ResultsService method), 30
SummaryPacket (class in cdrouter.captures), 44
SummaryStats (class in cdrouter.results), 33
system (cdrouter.CDRouter attribute), 4
SystemService (class in cdrouter.system), 52

T

Tag (class in cdrouter.tags), 58
tags (cdrouter.CDRouter attribute), 4
TagsService (class in cdrouter.tags), 57
Test (class in cdrouter.testsuites), 62
TestCount (class in cdrouter.results), 31
TestDuration (class in cdrouter.results), 32
TestResult (class in cdrouter.testresults), 38
TestResultBreakdown (class in cdrouter.results), 32
TestResultsService (class in cdrouter.testresults), 36
tests (cdrouter.CDRouter attribute), 4
Testsuite (class in cdrouter.system), 54
testsuites (cdrouter.CDRouter attribute), 4
TestsuitesService (class in cdrouter.testsuites), 58
Testvar (class in cdrouter.configs), 12
Testvar (class in cdrouter.testsuites), 63
thumbnail() (cdrouter.attachments.AttachmentsService method), 18
time() (cdrouter.system.SystemService method), 54
TimeBreakdown (class in cdrouter.results), 32

U

unpause() (cdrouter.results.ResultsService method), 30
Update (class in cdrouter.results), 35
update() (cdrouter.testsuites.TestsuitesService method), 60
updates() (cdrouter.results.ResultsService method), 30
Upgrade (class in cdrouter.system), 55
upgrade_config() (cdrouter.configs.ConfigsService method), 11
UpgradeConfig (class in cdrouter.configs), 12
User (class in cdrouter.users), 66
users (cdrouter.CDRouter attribute), 4
UsersService (class in cdrouter.users), 64

V

Version (class in cdrouter.system), 54