
cdrouter.py Documentation

Release 0.7.0

QA Cafe

Sep 26, 2022

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	8
2.4	Devices	13
2.5	PowerCmd	17
2.6	Connection	17
2.7	Attachments	18
2.8	Jobs	20
2.9	Packages	23
2.10	Results	27
2.11	Update	37
2.12	Alerts	39
2.13	TestResults	42
2.14	Annotations	47
2.15	Captures	48
2.16	Highlights	50
2.17	Imports	52
2.18	Exports	54
2.19	History	55
2.20	System	56
2.21	Tags	62
2.22	Testsuites	63
2.23	Users	69
	Python Module Index	73
	Index	75

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/>.

1.1 Download & Install

cdrouter is available on [PyPI](#).

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

1.2 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
from cdrouter.jobs import Options
```

(continues on next page)

(continued from previous page)

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

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

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

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

    print('    Result-ID: {}'.format(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.

Python client for the CDRouter Web API.

2.1 CDRouter

2.1.1 CDRouter

```
class cdrouter.CDRouter(base, token=None, username=None, password=None, _getuser=<function  
_getuser_default>, _getpass=<function _getpass_default>, retries=3, in-  
secure=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.

- **retries** – (optional) The number of times to retry authentication with the CDRouter system before giving up as an int.
- **insecure** – (optional) If bool *True* and *base* is an HTTPS URL, skip certificate verification and allow insecure connections to the CDRouter system.

alerts = None

alerts.AlertsService object

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 retries 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
```

2.1.2 CDRouterError

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

2.1.3 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.

2.1.4 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.

2.2 Filters

2.2.1 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 (\leq) 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 (\sim) filter. Combine with `not_` method to construct a negative regexp match ($!\sim$) filter.

Parameters

- **value** – Filter value
- **ignorecase** – If bool *True*, make match case insensitive (\sim^* , $!\sim^*$)

Returns *filters.Field* object

Return type *filters.Field*

ne (*value*)

Construct a not equal to (\neq) 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*

2.3 Configs

2.3.1 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_interfaces (*contents*)
Process config contents with `cdrouter-cli -print-interfaces`.

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

Returns `configs.Interfaces` list

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, detailed=None)

Get a list of configs, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

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`

2.3.2 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.
- **interfaces** – (optional) `configs.Interfaces` list (if a config snapshot).

2.3.3 Page

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

Parameters

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

2.3.4 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.

2.3.5 CheckConfig

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

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

2.3.6 UpgradeConfig

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

Parameters

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

2.3.7 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.

2.3.8 Interfaces

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

Parameters

- **name** – (optional) Interface name (i.e. 'lan', 'wan', etc.) as string.
- **value** – (optional) Interface value (i.e. 'eth0', 'eth1', etc.) as string.
- **is_wireless** – (optional) Bool *True* if interface is a wireless interface.
- **is_ics** – (optional) Bool *True* if interface is the ICS interface.

2.3.9 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.

2.4 Devices

2.4.1 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*, *detailed=None*)

Get a list of devices, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

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*

2.4.2 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

2.4.3 Page

class `cdrouter.devices.Page`

Named tuple for a page of list response data.

Parameters

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

2.5 PowerCmd

class `cdrouter.devices.PowerCmd` (***kwargs*)

Model for CDRouter Device Power command result.

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

2.6 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.

2.7 Attachments

2.7.1 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*, *detailed=None*)

Get a list of a device's attachments, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

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'*)

2.7.2 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.

2.7.3 Page

class `cdrouter.attachments.Page`

Named tuple for a page of list response data.

Parameters

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

2.8 Jobs

2.8.1 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, _fields=None, filter=None, type=None, all=False*)

Bulk launch a set of jobs.

Parameters

- **jobs** – `jobs.Job` list
- **_fields** – `jobs.Job` object
- **filter** – (optional) Filters to apply as a string list.
- **type** – (optional) *union* or *inter* as string.
- **all** – (optional) Apply to all if bool *True*.

Returns `jobs.Job` list

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*

get_interfaces (*resource*)

Process job with `cdrouter-cli -print-interfaces`.

Parameters *resource* – `jobs.Job` object

Returns `configs.Interfaces` list

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, detailed=None*)

Get a list of jobs, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `jobs.Page` object

2.8.2 Job

class `cdrouter.jobs.Job` (***kwargs*)

Model for CDRouter Jobs.

Parameters

- **id** – (optional) Job ID as an int.
- **active** – (optional) Bool *True* if status is 'running'.

- **status** – (optional) Job status as a string.
- **options** – (optional) `jobs.Options` object
- **package_id** – (optional) Package ID as an int.
- **package_name** – (optional) Package name as string.
- **config_id** – (optional) Config ID as an int.
- **config_name** – (optional) Config 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*.
- **automatic** – (optional) Bool *True* if job scheduled automatically *DateTime*.
- **run_at** – (optional) Job scheduled run-time *DateTime*.
- **interfaces** – (optional) `configs.Interfaces` list
- **interface_names** – (optional) Job interface names as string list.
- **uses_wireless** – (optional) Bool *True* if job uses any wireless interfaces.
- **uses_ics** – (optional) Bool *True* if job uses any ICS interfaces.

2.8.3 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.

2.8.4 Page

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

Parameters

- **data** – `jobs.Job` list
- **links** – `cdrouter.Links` object

2.9 Packages

2.9.1 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_interfaces (*resource*)

Process package with `cdrouter-cli -print-interfaces`.

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

Returns `configs.Interfaces` list

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, detailed=None*)

Get a list of packages, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `packages.Page` object

testlist_expanded (*id*)

Get a list of all tests in a package, with any addons, modules or testlists expanded.

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

Return type string list

2.9.2 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.
- **schedule** – (optional) `packages.Schedule` object
- **interfaces** – (optional) `configs.Interfaces` list (if a package snapshot).

2.9.3 Options

class `cdrouter.packages.Options` (***kwargs*)

Model for CDRouter Package Options.

Parameters

- **forever** – (optional) Bool *True* if package looped forever.
- **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.

2.9.4 Schedule

class `cdrouter.packages.Schedule` (***kwargs*)

Model for CDRouter Package Schedule.

Parameters

- **enabled** – (optional) Bool *True* if schedule is enabled.
- **spec** – (optional) Cron spec schedule as a string.
- **options** – (optional) `results.Options` object

2.9.5 Page

class `cdrouter.packages.Page`

Named tuple for a page of list response data.

Parameters

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

2.9.6 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.

2.10 Results

2.10.1 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_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.DiffStats` object

Return type `results.DiffStats`

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*, *detailed=None*)

Get a list of results, using summary representation by default (see `detailed` parameter).

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.

- **detailed** – (optional) Return all fields if Bool *True*.

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*

2.10.2 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.
- **active** – (optional) Bool *True* if status is ‘running’ or ‘paused’.
- **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.
- **alerts** – (optional) Alert 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
- **features** – (optional) Dict of feature name strings to `results.Feature` objects.
- **interfaces** – (optional) `configs.Interfaces` list.

2.10.3 Page

class `cdrouter.results.Page`

Named tuple for a page of list response data.

Parameters

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

2.10.4 TestCount

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

Parameters

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

2.10.5 TestDuration

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

Parameters

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

2.10.6 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.
- **alerted** – (optional) Alerted count as an int.

2.10.7 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.

2.10.8 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

2.10.9 TestResultSummary

class cdrouter.results.TestResultSummary (**kwargs)
Model for CDRouter TestResult summaries.

Parameters

- **id** – (optional) Result ID as an int.
- **seq** – (optional) TestResult sequence ID as an int.
- **result** – (optional) Test result as string.
- **alerts** – (optional) Alert count as an int.
- **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.

2.10.10 TestResultDiff

class cdrouter.results.TestResultDiff (**kwargs)
Model for CDRouter TestResult diffs.

Parameters

- **name** – (optional) Test name as a string.
- **summaries** – (optional) results.TestResultSummary list

2.10.11 DiffStats

class cdrouter.results.DiffStats (**kwargs)
Model for CDRouter Result Diff Stats.

Parameters **tests** – (optional) results.TestResultDiff list

2.10.12 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

2.10.13 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.

2.10.14 SingleStats

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

Parameters

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

2.10.15 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

2.10.16 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.

2.10.17 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.

2.10.18 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

2.10.19 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) First metric value as a float.
- **units** – (optional) First 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.
- **protocol** – (optional) Protocol as a string.
- **direction** – (optional) Direction as a string.
- **value_2** – (optional) Second metric value as a float.
- **units_2** – (optional) Second metric units as a string.
- **device_1** – (optional) First device as a string.
- **device_2** – (optional) Second device as a string.

2.10.20 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*.

2.10.21 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.

2.10.22 Feature

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

Parameters

- **feature** – (optional) Feature as a string.
- **enabled** – (optional) *True* if feature is enabled.
- **reason** – (optional) Reason as a string

2.11 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`, `testresults.TestResult` and `alerts.Alert` list

2.11.1 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.
- **active** – (optional) Bool *True* if status is ‘running’ or ‘paused’.
- **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.
- **alerts** – (optional) Alert 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
- **features** – (optional) Dict of feature name strings to `results.Feature` objects.
- **interfaces** – (optional) `configs.Interfaces` list.

2.12 Alerts

2.12.1 AlertsService

class `cdrouter.alerts.AlertsService` (*service*)

Service for accessing CDRouter Alerts.

all_stats (*id*)

Compute alert stats for a single result.

Parameters *id* – Result ID as an int.

Returns `alerts.AllStats` object

Return type `alerts.AllStats`

get (*id*, *idx*)

Get an alert.

Parameters

- *id* – Result ID as an int.
- *idx* – Alert index as an int.

Returns `alerts.Alert` object

Return type `alerts.Alert`

iter_list (*id*, **args*, ***kwargs*)

Get a list of alerts. Whereas `list` fetches a single page of alerts according to its `limit` and `page` arguments, `iter_list` returns all alerts 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 `alerts.Alert` list

list (*id*, *filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*, *detailed=None*)

Get a list of alerts, using summary representation by default (see `detailed` parameter).

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.
- *detailed* – (optional) Return all fields if Bool *True*.

Returns `alerts.Page` object

2.12.2 Alert

class `cdrouter.alerts.Alert` (***kwargs*)
Model for CDRouter Alerts.

Parameters

- **id** – (optional) Result ID as an int.
- **idx** – (optional) Alert index as an int.
- **created** – (optional) Creation time as *DateTime*.
- **updated** – (optional) Last-updated time as *DateTime*.
- **seq** – (optional) TestResult sequence ID as an int.
- **loop** – (optional) Loop number as an int.
- **test_name** – (optional) Test name as string.
- **test_description** – (optional) Test description as string.
- **category** – (optional) Alert category as a string.
- **description** – (optional) Alert description as a string.
- **dest_ip** – (optional) Alert destination IP as a string.
- **dest_port** – (optional) Alert destination port as an int.
- **interface** – (optional) Alert incoming interface as a string.
- **payload** – (optional) Alert payload as a base64 string.
- **payload_ascii** – (optional) Alert payload as an ASCII string.
- **payload_hex** – (optional) Alert payload as a hex string.
- **proto** – (optional) Alert protocol as a string.
- **references** – (optional) Alert references as a string list.
- **rev** – (optional) Alert rule revision as an int.
- **rule** – (optional) Alert rule as a string.
- **rule_set** – (optional) Alert rule set as a string.
- **severity** – (optional) Alert severity as an int.
- **sid** – (optional) Alert rule SID as an int.
- **signature** – (optional) Alert signature as a string.
- **src_ip** – (optional) Alert source IP as a string.
- **src_port** – (optional) Alert source port as an int.

2.12.3 Page

class `cdrouter.alerts.Page`
Named tuple for a page of list response data.

Parameters

- **data** – `alerts.Alert` list

- **links** – `cdrouter.Links` object

2.12.4 SeverityCount

class `cdrouter.alerts.SeverityCount` (***kwargs*)
Model for CDRouter Severity Counts.

Parameters

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

2.12.5 CategoryCount

class `cdrouter.alerts.CategoryCount` (***kwargs*)
Model for CDRouter Category Counts.

Parameters

- **category** – (optional) Category name as a string.
- **severity** – (optional) Severity as an int.
- **count** – (optional) Count as an int.

2.12.6 RuleSetCount

class `cdrouter.alerts.RuleSetCount` (***kwargs*)
Model for CDRouter RuleSet Counts.

Parameters

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

2.12.7 SignatureCount

class `cdrouter.alerts.SignatureCount` (***kwargs*)
Model for CDRouter Signature Counts.

Parameters

- **signature** – (optional) Signature name as a string.
- **severity** – (optional) Severity as an int.
- **count** – (optional) Count as an int.

2.12.8 TestCount

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

Parameters

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

2.12.9 AddrCount

class `cdrouter.alerts.AddrCount` (***kwargs*)
Model for CDRouter Addr Counts.

Parameters

- **addr** – (optional) Address as a string.
- **count** – (optional) Count as an int.

2.12.10 AllStats

class `cdrouter.alerts.AllStats` (***kwargs*)
Model for CDRouter All Alerts Stats.

Parameters

- **severities** – (optional) Dict of ints to `alerts.SeverityCount`
- **categories** – (optional) `alerts.CategoryCount` list
- **rule_sets** – (optional) `alerts.RuleSetCount` list
- **signatures** – (optional) `alerts.SignatureCount` list
- **tests** – (optional) `alerts.TestCount` list
- **frequent_sources** – (optional) `alerts.AddrCount` list
- **frequent_destinations** – (optional) `alerts.AddrCount` list

2.13 TestResults

2.13.1 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*, *detailed=None*)

Get a list of test results, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

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`

2.13.2 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.
- **active** – (optional) Bool *True* if result is 'running' or 'waiting'.
- **result** – (optional) Test result as string.
- **alerts** – (optional) Alerts count as int.

- **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.

2.13.3 Page

class `cdrouter.testresults.Page`

Named tuple for a page of list response data.

Parameters

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

2.13.4 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.
- **warnings** – (optional) Warning log count as int.
- **alerts** – (optional) Alert log count as int.

2.13.5 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.
- **timestamp_display** – (optional) Log display 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).
- **alert_interface** – (optional) Log alert interface as string (if alert log).
- **alert_index** – (optional) Log alert index as an int (if alert log).
- **alert_src** – (optional) Log alert source address as string (if alert log).
- **alert_dst** – (optional) Log alert destination address as string (if alert log).
- **alert_proto** – (optional) Log alert protocol name as string (if alert log).
- **alert_src_port** – (optional) Log alert source port as int (if alert log).
- **alert_dst_port** – (optional) Log alert destination port as int (if alert log).
- **alert_signature** – (optional) Log alert signature as string (if alert log).
- **alert_severity** – (optional) Log alert severity as int (if alert log).
- **alert_severity_display** – (optional) Log alert display severity as string (if alert log).
- **alert_sid** – (optional) Log alert SID as int (if alert log).
- **alert_rev** – (optional) Log alert revision as int (if alert log).
- **summary** – (optional) `testresults.Summary` object (if section log)

2.13.6 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.

2.14 Annotations

2.14.1 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, detailed=None*)

Get a list of annotations, using summary representation by default (see `detailed` parameter).

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `annotations.Annotation` list

2.14.2 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.

2.15 Captures

2.15.1 CapturesService

class `cdrouter.captures.CapturesService` (*service*)

Service for accessing CDRouter Captures.

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, detailed=None*)

Get a list of captures, using summary representation by default (see `detailed` parameter).

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **detailed** – (optional) Return all fields if Bool *True*.

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`*

2.15.2 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.

2.15.3 Section

2.15.4 Structure

2.15.5 SummaryPacket

2.15.6 Summary

2.15.7 Field

2.15.8 Proto

2.15.9 Packet

2.15.10 Decode

2.15.11 ASCIIByte

2.15.12 ASCIILine

2.15.13 ASCIIFrame

2.15.14 ASCII

2.15.15 CloudShark

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

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

2.16 Highlights

2.16.1 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, detailed=None*)

Get a list of highlights, using summary representation by default (see `detailed` parameter).

Parameters

- **id** – Result ID as an int.
- **seq** – TestResult sequence ID as an int.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `highlights.Highlight` list

2.16.2 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.

2.17 Imports

2.17.1 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

2.17.2 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.

2.17.3 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.

2.17.4 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

2.17.5 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.

2.18 Exports

2.18.1 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'*)

2.19 History

2.19.1 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, detailed=None*)

Get a list of history entries, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `history.Page` object

2.19.2 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.

2.19.3 Page

class `cdrouter.history.Page`

Named tuple for a page of list response data.

Parameters

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

2.20 System

2.20.1 SystemService

class `cdrouter.system.SystemService` (*service*)

Service for accessing CDRouter System.

check_for_lounge_upgrade (*email*)

Check the CDRouter Support Lounge for eligible upgrades using your Support Lounge email.

Parameters **email** – CDRouter Support Lounge email as a string.

Returns `system.Release` object

Return type *system.Release*

diagnostics ()

Get system diagnostics from `cdrouter-diag` output.

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

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

in_use_interfaces()

Get system interfaces, with flags to determine which are in use.

Returns `system.InUseInterface` list

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*

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, nonce, filename='cdrouter.rpm'*)

Download & install an upgrade from the CDRouter Support Lounge using your Support Lounge email and upgrade nonce. Please note that any running tests will be stopped.

Parameters

- **email** – CDRouter Support Lounge email as a string.
- **nonce** – Upgrade nonce from `system.Release`.
- **filename** – Upgrade filename from `system.Release`.

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.

manual_upgrade (*fd, filename='cdrouter.rpm'*)

Upgrade CDRouter manually by uploading an `.rpm` 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`

poweroff()

Poweroff the NTA1000. Please note that any running tests will be stopped.

reboot()

Reboot the NTA1000. Please note that any running tests will be stopped.

restart()

Restart the CDRouter Web UI. Please note that any running tests will be stopped.

shutdown()

Shutdown the CDRouter Web UI. 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

2.20.2 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.

2.20.3 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.

2.20.4 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.

2.20.5 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
- **testsuite** – (optional) `system.Testsuite` object
- **nonce** – (optional) Upgrade nonce as string.

2.20.6 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.

2.20.7 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.

2.20.8 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.

2.20.9 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

2.20.10 InUseInterfaceFlags

class cdrouter.system.**InUseInterfaceFlags** (**kwargs)
Model for CDRouter In Use Interface Flags.

Parameters

- **in_use** – (optional) Bool *True* if interface is in use by a running job.
- **is_wireless** – (optional) Bool *True* if interface is a wireless interface.
- **is_ics** – (optional) Bool *True* if interface is in use as the ICS interface of a running job.

2.20.11 InUseInterface

class cdrouter.system.**InUseInterface** (**kwargs)
Model for CDRouter In Use Interfaces.

Parameters

- **name** – (optional) Interface name as a string.
- **flags** – (optional) system.InUseInterfaceFlags object

2.20.12 Preferences

class `cdrouter.system.Preferences` (***kwargs*)

Model for CDRouter Preferences.

Parameters

- **automatic_login** – (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.

2.20.13 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.

2.21 Tags

2.21.1 TagsService

class `cdrouter.tags.TagsService` (*service*)

Service for accessing CDRouter Tags.

delete (*name*)

Delete a tag.

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

edit (*resource*, *old_name=None*)

Edit a tag. When renaming a tag, *old_name* must be set to the old tag name.

Parameters

- **resource** – `tags.Tag` object
- **old_name** – (optional) Old tag name as a string. Only required when renaming a tag.

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

2.21.2 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

2.21.3 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.

2.22 Testsuites

2.22.1 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, detailed=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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `testsuites.Error` list

list_groups (*filter=None, type=None, sort=None, detailed=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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `testsuites.Group` list

list_labels (*filter=None, type=None, sort=None, detailed=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.

- **detailed** – (optional) Return all fields if Bool *True*.

Returns `testsuites.Label` list

list_modules (*filter=None, type=None, sort=None, detailed=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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `testsuites.Module` list

list_tests (*filter=None, type=None, sort=None, detailed=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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `testsuites.Test` list

list_testvars (*filter=None, type=None, sort=None, detailed=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.
- **detailed** – (optional) Return all fields if Bool *True*.

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.

2.22.2 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.
- **interfaces** – (optional) `configs.Interfaces` list
- **execute_instances** – (optional) Execute instances as an int.

2.22.3 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.

2.22.4 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.

2.22.5 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.

2.22.6 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.

2.22.7 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.

2.22.8 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.

2.22.9 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

2.23 Users

2.23.1 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, detailed=None*)

Get a list of users, using summary representation by default (see `detailed` parameter).

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.
- **detailed** – (optional) Return all fields if Bool *True*.

Returns `users.Page` object

2.23.2 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.

2.23.3 Page

class `cdrouter.users.Page`

Named tuple for a page of list response data.

Parameters

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

C

`cdrouter`, 3

A

AddrCount (class in *cdrouter.alerts*), 42
 Alert (class in *cdrouter.alerts*), 40
 alerts (*cdrouter.CDRouter* attribute), 4
 AlertsService (class in *cdrouter.alerts*), 39
 all_stats() (*cdrouter.alerts.AlertsService* method), 39
 all_stats() (*cdrouter.results.ResultsService* method), 27
 AllStats (class in *cdrouter.alerts*), 42
 AllStats (class in *cdrouter.results*), 36
 Analysis (class in *cdrouter.packages*), 27
 analyze() (*cdrouter.packages.PackagesService* method), 23
 Annotation (class in *cdrouter.annotations*), 48
 annotations (*cdrouter.CDRouter* attribute), 4
 AnnotationsService (class in *cdrouter.annotations*), 47
 Attachment (class in *cdrouter.attachments*), 19
 attachments (*cdrouter.CDRouter* attribute), 4
 AttachmentsService (class in *cdrouter.attachments*), 18
 authenticate() (*cdrouter.CDRouter* method), 4

B

bulk_copy() (*cdrouter.configs.ConfigsService* method), 8
 bulk_copy() (*cdrouter.devices.DevicesService* method), 13
 bulk_copy() (*cdrouter.packages.PackagesService* method), 23
 bulk_copy() (*cdrouter.users.UsersService* method), 69
 bulk_delete() (*cdrouter.configs.ConfigsService* method), 8
 bulk_delete() (*cdrouter.devices.DevicesService* method), 13
 bulk_delete() (*cdrouter.jobs.JobsService* method), 20

bulk_delete() (*cdrouter.packages.PackagesService* method), 23
 bulk_delete() (*cdrouter.results.ResultsService* method), 27
 bulk_delete() (*cdrouter.users.UsersService* method), 69
 bulk_edit() (*cdrouter.configs.ConfigsService* method), 8
 bulk_edit() (*cdrouter.devices.DevicesService* method), 14
 bulk_edit() (*cdrouter.packages.PackagesService* method), 23
 bulk_edit() (*cdrouter.results.ResultsService* method), 27
 bulk_edit() (*cdrouter.users.UsersService* method), 69
 bulk_edit_testvars() (*cdrouter.configs.ConfigsService* method), 8
 bulk_export() (*cdrouter.configs.ConfigsService* method), 8
 bulk_export() (*cdrouter.devices.DevicesService* method), 14
 bulk_export() (*cdrouter.exports.ExportsService* method), 54
 bulk_export() (*cdrouter.packages.PackagesService* method), 23
 bulk_export() (*cdrouter.results.ResultsService* method), 27
 bulk_launch() (*cdrouter.jobs.JobsService* method), 20

C

Capture (class in *cdrouter.captures*), 49
 captures (*cdrouter.CDRouter* attribute), 4
 CapturesService (class in *cdrouter.captures*), 48
 CategoryCount (class in *cdrouter.alerts*), 41
 CDRouter (class in *cdrouter*), 3
 cdrouter (module), 3
 CDRouterError (class in *cdrouter.cdrouter*), 5

[change_password\(\)](#) (*cdrouter.users.UsersService method*), 69
[change_token\(\)](#) (*cdrouter.users.UsersService method*), 70
[check_config\(\)](#) (*cdrouter.configs.ConfigsService method*), 8
[check_for_lounge_upgrade\(\)](#) (*cdrouter.system.SystemService method*), 56
[CheckConfig](#) (*class in cdrouter.configs*), 12
[CloudShark](#) (*class in cdrouter.captures*), 50
[commit\(\)](#) (*cdrouter.imports.ImportsService method*), 52
[Config](#) (*class in cdrouter.configs*), 11
[ConfigError](#) (*class in cdrouter.configs*), 12
[configs](#) (*cdrouter.CDRouter attribute*), 4
[ConfigsService](#) (*class in cdrouter.configs*), 8
[connect\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[Connection](#) (*class in cdrouter.devices*), 17
[contained_by\(\)](#) (*cdrouter.filters.Field method*), 6
[contains\(\)](#) (*cdrouter.filters.Field method*), 6
[create\(\)](#) (*cdrouter.annotations.AnnotationsService method*), 47
[create\(\)](#) (*cdrouter.attachments.AttachmentsService method*), 18
[create\(\)](#) (*cdrouter.configs.ConfigsService method*), 9
[create\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[create\(\)](#) (*cdrouter.highlights.HighlightsService method*), 50
[create\(\)](#) (*cdrouter.packages.PackagesService method*), 23
[create\(\)](#) (*cdrouter.users.UsersService method*), 70
[create_or_edit\(\)](#) (*cdrouter.annotations.AnnotationsService method*), 47
[create_or_edit\(\)](#) (*cdrouter.highlights.HighlightsService method*), 50

D

[delete\(\)](#) (*cdrouter.annotations.AnnotationsService method*), 47
[delete\(\)](#) (*cdrouter.attachments.AttachmentsService method*), 18
[delete\(\)](#) (*cdrouter.configs.ConfigsService method*), 9
[delete\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[delete\(\)](#) (*cdrouter.highlights.HighlightsService method*), 51
[delete\(\)](#) (*cdrouter.imports.ImportsService method*), 52
[delete\(\)](#) (*cdrouter.jobs.JobsService method*), 20
[delete\(\)](#) (*cdrouter.packages.PackagesService method*), 23
[delete\(\)](#) (*cdrouter.results.ResultsService method*), 27
[delete\(\)](#) (*cdrouter.tags.TagsService method*), 62
[delete\(\)](#) (*cdrouter.users.UsersService method*), 70
[delete_testvar\(\)](#) (*cdrouter.configs.ConfigsService method*), 9
[Device](#) (*class in cdrouter.devices*), 16
[DeviceCount](#) (*class in cdrouter.results*), 36
[devices](#) (*cdrouter.CDRouter attribute*), 4
[DevicesService](#) (*class in cdrouter.devices*), 13
[diagnostics\(\)](#) (*cdrouter.system.SystemService method*), 56
[diff_stats\(\)](#) (*cdrouter.results.ResultsService method*), 28
[DiffStats](#) (*class in cdrouter.results*), 34
[disconnect\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[download\(\)](#) (*cdrouter.attachments.AttachmentsService method*), 18
[download\(\)](#) (*cdrouter.captures.CapturesService method*), 48
[download_logdir_archive\(\)](#) (*cdrouter.results.ResultsService method*), 28

E

[edit\(\)](#) (*cdrouter.annotations.AnnotationsService method*), 47
[edit\(\)](#) (*cdrouter.attachments.AttachmentsService method*), 18
[edit\(\)](#) (*cdrouter.configs.ConfigsService method*), 9
[edit\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[edit\(\)](#) (*cdrouter.highlights.HighlightsService method*), 51
[edit\(\)](#) (*cdrouter.jobs.JobsService method*), 20
[edit\(\)](#) (*cdrouter.packages.PackagesService method*), 24
[edit\(\)](#) (*cdrouter.results.ResultsService method*), 28
[edit\(\)](#) (*cdrouter.tags.TagsService method*), 62
[edit\(\)](#) (*cdrouter.testresults.TestResultsService method*), 42
[edit\(\)](#) (*cdrouter.users.UsersService method*), 70
[edit_preferences\(\)](#) (*cdrouter.system.SystemService method*), 56
[edit_shares\(\)](#) (*cdrouter.configs.ConfigsService method*), 9
[edit_shares\(\)](#) (*cdrouter.devices.DevicesService method*), 14
[edit_shares\(\)](#) (*cdrouter.packages.PackagesService method*), 24
[edit_shares\(\)](#) (*cdrouter.results.ResultsService method*), 28
[edit_testvar\(\)](#) (*cdrouter.configs.ConfigsService method*), 9

`eq()` (*cdrouter.filters.Field* method), 6
Error (class in *cdrouter.testsuites*), 67
`export()` (*cdrouter.configs.ConfigsService* method), 9
`export()` (*cdrouter.devices.DevicesService* method), 15
`export()` (*cdrouter.packages.PackagesService* method), 24
`export()` (*cdrouter.results.ResultsService* method), 28
`exports` (*cdrouter.CDRouter* attribute), 4
ExportsService (class in *cdrouter.exports*), 54

F

Feature (class in *cdrouter.results*), 37
Field (class in *cdrouter.filters*), 5

G

`ge()` (*cdrouter.filters.Field* method), 6
`get()` (*cdrouter.alerts.AlertsService* method), 39
`get()` (*cdrouter.annotations.AnnotationsService* method), 47
`get()` (*cdrouter.attachments.AttachmentsService* method), 18
`get()` (*cdrouter.captures.CapturesService* method), 48
`get()` (*cdrouter.configs.ConfigsService* method), 9
`get()` (*cdrouter.devices.DevicesService* method), 15
`get()` (*cdrouter.highlights.HighlightsService* method), 51
`get()` (*cdrouter.imports.ImportsService* method), 52
`get()` (*cdrouter.jobs.JobsService* method), 21
`get()` (*cdrouter.packages.PackagesService* method), 24
`get()` (*cdrouter.results.ResultsService* method), 28
`get()` (*cdrouter.tags.TagsService* method), 62
`get()` (*cdrouter.testresults.TestResultsService* method), 42
`get()` (*cdrouter.users.UsersService* method), 70
`get_by_name()` (*cdrouter.configs.ConfigsService* method), 10
`get_by_name()` (*cdrouter.devices.DevicesService* method), 15
`get_by_name()` (*cdrouter.packages.PackagesService* method), 24
`get_by_name()` (*cdrouter.users.UsersService* method), 70
`get_commit_request()` (*cdrouter.imports.ImportsService* method), 52
`get_connection()` (*cdrouter.devices.DevicesService* method), 15
`get_error()` (*cdrouter.testsuites.TestsuitesService* method), 63
`get_group()` (*cdrouter.testsuites.TestsuitesService* method), 63
`get_interfaces()` (*cdrouter.configs.ConfigsService* method), 10

`get_interfaces()` (*cdrouter.jobs.JobsService* method), 21
`get_interfaces()` (*cdrouter.packages.PackagesService* method), 24
`get_label()` (*cdrouter.testsuites.TestsuitesService* method), 63
`get_log_plaintext()` (*cdrouter.testresults.TestResultsService* method), 43
`get_logdir_file()` (*cdrouter.results.ResultsService* method), 28
`get_module()` (*cdrouter.testsuites.TestsuitesService* method), 63
`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), 56
`get_shares()` (*cdrouter.configs.ConfigsService* method), 10
`get_shares()` (*cdrouter.devices.DevicesService* method), 15
`get_shares()` (*cdrouter.packages.PackagesService* method), 24
`get_shares()` (*cdrouter.results.ResultsService* method), 29
`get_test()` (*cdrouter.testsuites.TestsuitesService* method), 64
`get_test_metric()` (*cdrouter.results.ResultsService* method), 29
`get_test_metric_csv()` (*cdrouter.results.ResultsService* method), 29
`get_testvar()` (*cdrouter.configs.ConfigsService* method), 10
`get_testvar()` (*cdrouter.testsuites.TestsuitesService* method), 64
Group (class in *cdrouter.testsuites*), 66
`gt()` (*cdrouter.filters.Field* method), 6

H

Highlight (class in *cdrouter.highlights*), 52
`highlights` (*cdrouter.CDRouter* attribute), 4
HighlightsService (class in *cdrouter.highlights*), 50
`history` (*cdrouter.CDRouter* attribute), 4
History (class in *cdrouter.history*), 55
HistoryService (class in *cdrouter.history*), 55
`hostname()` (*cdrouter.system.SystemService* method), 57

I

`Import` (class in `cdrouter.imports`), 53
`imports` (`cdrouter.CDRouter` attribute), 4
`ImportsService` (class in `cdrouter.imports`), 52
`in_use_interfaces()`
 (`cdrouter.system.SystemService` method), 57
`Info` (class in `cdrouter.testsuites`), 65
`info()` (`cdrouter.system.SystemService` method), 57
`info()` (`cdrouter.testsuites.TestsuitesService` method), 64
`Interface` (class in `cdrouter.system`), 60
`InterfaceAddr` (class in `cdrouter.system`), 60
`InterfaceFlags` (class in `cdrouter.system`), 59
`Interfaces` (class in `cdrouter.configs`), 13
`interfaces()` (`cdrouter.system.SystemService` method), 57
`InUseInterface` (class in `cdrouter.system`), 60
`InUseInterfaceFlags` (class in `cdrouter.system`), 60
`iter_list()` (`cdrouter.alerts.AlertsService` method), 39
`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), 55
`iter_list()` (`cdrouter.jobs.JobsService` method), 21
`iter_list()` (`cdrouter.packages.PackagesService` method), 24
`iter_list()` (`cdrouter.results.ResultsService` method), 29
`iter_list()` (`cdrouter.testresults.TestResultsService` method), 43
`iter_list()` (`cdrouter.users.UsersService` method), 70
`iter_list_log()` (`cdrouter.testresults.TestResultsService` method), 43

J

`Job` (class in `cdrouter.jobs`), 21
`jobs` (`cdrouter.CDRouter` attribute), 4
`JobsService` (class in `cdrouter.jobs`), 20

L

`Label` (class in `cdrouter.testsuites`), 67
`latest_lounge_release()`
 (`cdrouter.system.SystemService` method), 57
`launch()` (`cdrouter.jobs.JobsService` method), 21
`le()` (`cdrouter.filters.Field` method), 7

`Line` (class in `cdrouter.testresults`), 45
`Links` (class in `cdrouter.cdrouter`), 5
`list()` (`cdrouter.alerts.AlertsService` method), 39
`list()` (`cdrouter.annotations.AnnotationsService` method), 48
`list()` (`cdrouter.attachments.AttachmentsService` method), 19
`list()` (`cdrouter.captures.CapturesService` method), 49
`list()` (`cdrouter.configs.ConfigsService` method), 11
`list()` (`cdrouter.devices.DevicesService` method), 15
`list()` (`cdrouter.highlights.HighlightsService` method), 51
`list()` (`cdrouter.history.HistoryService` method), 55
`list()` (`cdrouter.imports.ImportsService` method), 52
`list()` (`cdrouter.jobs.JobsService` method), 21
`list()` (`cdrouter.packages.PackagesService` method), 25
`list()` (`cdrouter.results.ResultsService` method), 29
`list()` (`cdrouter.tags.TagsService` method), 62
`list()` (`cdrouter.testresults.TestResultsService` method), 43
`list()` (`cdrouter.users.UsersService` method), 70
`list_csv()` (`cdrouter.results.ResultsService` method), 30
`list_csv()` (`cdrouter.testresults.TestResultsService` method), 44
`list_errors()` (`cdrouter.testsuites.TestsuitesService` method), 64
`list_groups()` (`cdrouter.testsuites.TestsuitesService` method), 64
`list_labels()` (`cdrouter.testsuites.TestsuitesService` method), 64
`list_log()` (`cdrouter.testresults.TestResultsService` method), 44
`list_logdir()` (`cdrouter.results.ResultsService` method), 30
`list_modules()` (`cdrouter.testsuites.TestsuitesService` method), 65
`list_tests()` (`cdrouter.testsuites.TestsuitesService` method), 65
`list_testvars()` (`cdrouter.configs.ConfigsService` method), 11
`list_testvars()` (`cdrouter.testsuites.TestsuitesService` method), 65
`Log` (class in `cdrouter.testresults`), 46
`LogDirFile` (class in `cdrouter.results`), 37
`lounge_update_license()`
 (`cdrouter.system.SystemService` method), 57
`lounge_upgrade()` (`cdrouter.system.SystemService` method), 57
`lt()` (`cdrouter.filters.Field` method), 7

M

`manual_update_license()`
(*cdrouter.system.SystemService method*), 57

`manual_upgrade()` (*cdrouter.system.SystemService method*), 57

`match()` (*cdrouter.filters.Field method*), 7

`Metric` (*class in cdrouter.results*), 36

`Module` (*class in cdrouter.testsuites*), 66

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*), 22

`Options` (*class in cdrouter.packages*), 26

`Options` (*class in cdrouter.results*), 37

`overlaps()` (*cdrouter.filters.Field method*), 7

P

`Package` (*class in cdrouter.packages*), 25

`PackageCount` (*class in cdrouter.results*), 35

`packages` (*cdrouter.CDRouter attribute*), 4

`PackagesService` (*class in cdrouter.packages*), 23

`Page` (*class in cdrouter.alerts*), 40

`Page` (*class in cdrouter.attachments*), 20

`Page` (*class in cdrouter.configs*), 12

`Page` (*class in cdrouter.devices*), 17

`Page` (*class in cdrouter.history*), 56

`Page` (*class in cdrouter.jobs*), 22

`Page` (*class in cdrouter.packages*), 26

`Page` (*class in cdrouter.results*), 32

`Page` (*class in cdrouter.testresults*), 45

`Page` (*class in cdrouter.users*), 71

`pause()` (*cdrouter.results.ResultsService method*), 30

`pause_end_of_loop()`
(*cdrouter.results.ResultsService method*), 30

`pause_end_of_test()`
(*cdrouter.results.ResultsService method*), 30

`power_off()` (*cdrouter.devices.DevicesService method*), 16

`power_on()` (*cdrouter.devices.DevicesService method*), 16

`PowerCmd` (*class in cdrouter.devices*), 17

`poweroff()` (*cdrouter.system.SystemService method*), 58

`Preferences` (*class in cdrouter.system*), 61

`Progress` (*class in cdrouter.results*), 35

`progress_stats()` (*cdrouter.results.ResultsService method*), 30

R

`reboot()` (*cdrouter.system.SystemService method*), 58

`Release` (*class in cdrouter.system*), 59

`ReleaseLatest` (*class in cdrouter.system*), 58

`Request` (*class in cdrouter.imports*), 54

`Resource` (*class in cdrouter.imports*), 54

`ResourceTags` (*class in cdrouter.tags*), 63

`Response` (*class in cdrouter.imports*), 54

`restart()` (*cdrouter.system.SystemService method*), 58

`Result` (*class in cdrouter.results*), 31, 38

`ResultBreakdown` (*class in cdrouter.results*), 33

`results` (*cdrouter.CDRouter attribute*), 4

`ResultsService` (*class in cdrouter.results*), 27

`RuleSetCount` (*class in cdrouter.alerts*), 41

S

`Schedule` (*class in cdrouter.packages*), 26

`Search` (*class in cdrouter.testsuites*), 68

`search()` (*cdrouter.testsuites.TestsuitesService method*), 65

`send_to_cloudshark()`
(*cdrouter.captures.CapturesService method*), 49

`set_stats()` (*cdrouter.results.ResultsService method*), 30

`SetStats` (*class in cdrouter.results*), 34

`SeverityCount` (*class in cdrouter.alerts*), 41

`Share` (*class in cdrouter.cdrouter*), 5

`shutdown()` (*cdrouter.system.SystemService method*), 58

`SignatureCount` (*class in cdrouter.alerts*), 41

`single_stats()` (*cdrouter.results.ResultsService method*), 31

`SingleStats` (*class in cdrouter.results*), 35

`Space` (*class in cdrouter.system*), 61

`space()` (*cdrouter.system.SystemService method*), 58

`stage_import_from_file()`
(*cdrouter.imports.ImportsService method*), 52

`stage_import_from_filesystem()`
(*cdrouter.imports.ImportsService method*), 53

`stage_import_from_url()`
(*cdrouter.imports.ImportsService method*), 53

`stop()` (*cdrouter.results.ResultsService method*), 31

`stop_end_of_loop()`
(*cdrouter.results.ResultsService method*), 31

`stop_end_of_test()`
(*cdrouter.results.ResultsService method*), 31

`Summary` (*class in cdrouter.testresults*), 45

`summary_stats()` (*cdrouter.results.ResultsService*
 method), 31
`SummaryStats` (*class in cdrouter.results*), 35
`system` (*cdrouter.CDRouter* attribute), 4
`SystemService` (*class in cdrouter.system*), 56

T

`Tag` (*class in cdrouter.tags*), 62
`tags` (*cdrouter.CDRouter* attribute), 4
`TagsService` (*class in cdrouter.tags*), 62
`Test` (*class in cdrouter.testsuites*), 67
`TestCount` (*class in cdrouter.alerts*), 42
`TestCount` (*class in cdrouter.results*), 33
`TestDuration` (*class in cdrouter.results*), 33
`testlist_expanded()`
 (*cdrouter.packages.PackagesService* *method*),
 25
`TestResult` (*class in cdrouter.testresults*), 44
`TestResultBreakdown` (*class in cdrouter.results*),
 35
`TestResultDiff` (*class in cdrouter.results*), 34
`TestResultsService` (*class in cdrouter.testresults*),
 42
`TestResultSummary` (*class in cdrouter.results*), 34
`tests` (*cdrouter.CDRouter* attribute), 4
`Testsuite` (*class in cdrouter.system*), 59
`testsuites` (*cdrouter.CDRouter* attribute), 4
`TestsuitesService` (*class in cdrouter.testsuites*), 63
`Testvar` (*class in cdrouter.configs*), 13
`Testvar` (*class in cdrouter.testsuites*), 68
`thumbnail()` (*cdrouter.attachments.AttachmentsService*
 method), 19
`time()` (*cdrouter.system.SystemService* *method*), 58
`TimeBreakdown` (*class in cdrouter.results*), 33

U

`unpause()` (*cdrouter.results.ResultsService* *method*),
 31
`Update` (*class in cdrouter.results*), 37
`update()` (*cdrouter.testsuites.TestsuitesService*
 method), 65
`updates()` (*cdrouter.results.ResultsService* *method*),
 31
`Upgrade` (*class in cdrouter.system*), 59
`upgrade_config()` (*cdrouter.configs.ConfigsService*
 method), 11
`UpgradeConfig` (*class in cdrouter.configs*), 12
`User` (*class in cdrouter.users*), 71
`users` (*cdrouter.CDRouter* attribute), 4
`UsersService` (*class in cdrouter.users*), 69

V

`Version` (*class in cdrouter.system*), 58