cdrouter.py Documentation

Release 0.1.5

QA Cafe

January 25, 2017

Contents:

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

```python
import time
from cdrouter import CDRouter
from cdrouter.jobs import Job

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

for p in c.packages.list(filter=['tags@>{noretry}'], limit='none'):
    print('Launching package ' + p.name)

    j = c.jobs.launch(Job(package_id=p.id, extra_cli_args='-testvar myvar=example'))

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

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

print('done.')
```

Reference

2.1 cdrouter

Python client for the CDRouter Web API.

2.2 cdrouter.CDRouter

class cdrouter.**CDRouter**(*base*, *token=None*, *insecure=False*)
   Service for accessing the CDRouter Web API.

   Parameters

   - **base** – Base HTTP or HTTPS URL for CDRouter system as a string, optionally including a port. For example *http://localhost*, *http://cdrouter.lan:8015* or *https://127.0.0.1*.

   - **token** – (optional) CDRouter API token as a string. Not required if Automatic Login is enabled. If omitted, value will be taken from CDROUTER_API_TOKEN environment variable.

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

   annotations = None
      annotations.AnnotationsService object

   attachments = None
      attachments.AttachmentsService object

   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

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

>> **Parameters**

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

2.3 cdrouter.configs

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

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

>>> **Parameters ids** – String 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) String 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) String 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 string.
- **testvars** – [*configs.Testvar*](#) list

  Returns [*configs.Testvar*](#) list

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

  Parameters ids – String 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 string.

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 string.

- **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 string.

> > > - **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 string.

> > > - **resource** – *configs.Testvar* object.

> > **Returns** *configs.Testvar* object

> > **Return type** *configs.Testvar*

export (*id*)
> Export a config.

> > **Parameters id** – Config ID as string.

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

get (*id*)
> Get a config.

> > **Parameters id** – Config ID as string.

> > **Returns** *configs.Config* object

> > **Return type** *configs.Config*

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

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

> > **Returns** *configs.Networks* object

> > **Return type** *configs.Networks*

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

> > **Return type** string

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

> **Parameters id** – Config ID as string.
>
> **Return type** string

get_shares(*id*)
> Get shares for a config.
>
> > **Parameters id** – Config ID as string.
> >
> > **Returns** [*cdrouter.Share*](#) list

get_testvar(*id*, *name*, *group=None*)
> Get a testvar from a config.
>
> > **Parameters**
> >
> > - **id** – Config ID as string.
> > - **name** – Testvar name as string.
> > - **group** – (optional) Testvar group as string.
> >
> > **Returns** [*configs.Testvar*](#) object
> >
> > **Return type** [*configs.Testvar*](#)

list(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
> Get a list of configs.
>
> > **Parameters**
> >
> > - **filter** – (optional) Filters to apply as a string list.
> > - **type** – (optional) *union* or *inter* as string.
> > - **sort** – (optional) Sort fields to apply as string list.
> > - **limit** – (optional) Limit returned list length.
> > - **page** – (optional) Page to return.
> >
> > **Returns** [*configs.Config*](#) list

list_testvars(*id*)
> Get a list of a config's testvars.
>
> > **Parameters id** – Config ID as string.
> >
> > **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*](#)

class cdrouter.configs.**Config**(***kwargs*)
> Model for CDRouter Configs.
>
> > **Parameters**
> >
> > - **id** – (optional) Config ID as string.
> > - **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 string.

- **result_id** – (optional) Result ID as string (if a config snapshot).

- **tags** – (optional) Tags as string list.

- **note** – (optional) Note as string.

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.

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

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

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

    Parameters

- **success** – (optional) Bool *True* if successful.

- **output** – (optional) Output as string.

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.

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 cdrouter.devices

class cdrouter.devices.**DevicesService**(*service*)

> Service for accessing CDRouter Devices.

> **bulk_copy**(*ids*)
>
> > Bulk copy a set of devices.
> >
> > > **Parameters ids** – String 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) String 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) String 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** – String list of device IDs.
> > >
> > > **Return type** tuple *(io.BytesIO, 'filename')*

> **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 string.

> **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 string.

> > > • **user_ids** – User IDs as int list.

> > **Returns** *[cdrouter.Share](#)* list

export(*id*)
> Export a device.

> > **Parameters** **id** – Device ID as string.

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

get(*id*)
> Get a device.

> > **Parameters** **id** – Device ID as string.

> > **Returns** *[devices.Device](#)* object

> > **Return type** *[devices.Device](#)*

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

> > **Parameters** **id** – Device ID as string.

> > **Returns** *[cdrouter.Share](#)* list

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

> > **Parameters**

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

> > > • **type** – (optional) *union* or *inter* as string.

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

> > > • **limit** – (optional) Limit returned list length.

> > > • **page** – (optional) Page to return.

> > **Returns** *[devices.Device](#)* list

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

> > **Parameters**

> > > • **id** – (optional) Device ID as a string.

> > > • **name** – (optional) Name as string.

> > > • **created** – (optional) Creation time as *DateTime*.

> > > • **updated** – (optional) Last-updated time as *DateTime*.

> > > • **user_id** – (optional) User ID as string.

> > > • **result_id** – (optional) Result ID as string (if a device snapshot).

- **attachments_dir** – (optional) Filepath for attachments as string.
- **picture_id** – (optional) Attachment ID for used for device picture as string.
- **tags** – (optional) Tags as string list.
- **default_ip** – Default IP as a string (optional)
- **default_login** – Default login as a string (optional)
- **default_password** – Default password as a string (optional)
- **location** – Location as a string (optional)
- **device_category** – Device category as a string (optional)
- **manufacturer** – Manufacturer as a string (optional)
- **manufacturer_oui** – Manufacturer OUI as a string (optional)
- **model_name** – Model name as a string (optional)
- **model_number** – Model number as a string (optional)
- **description** – Description as a string (optional)
- **product_class** – Product class as a string (optional)
- **serial_number** – Serial number as a string (optional)
- **hardware_version** – Hardware version as a string (optional)
- **software_version** – Software version as a string (optional)
- **provisioning_code** – Provisioning code as a string (optional)
- **note** – Note as a string (optional)

2.5 cdrouter.attachments

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

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

        Parameters

- **id** – Device ID as string.
- **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 string.
- **attid** – Attachment ID as string.

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

        Parameters

- **id** – Device ID as string.
- **attid** – Attachment ID as string.

        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 string.
- **attid** – Attachment ID as string.

        Returns   *attachments.Attachment* object

        Return type   *attachments.Attachment*

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

        Parameters

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

        Returns   *attachments.Attachment* list

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 string.
- **attid** – Attachment ID as string.
- **size** – (optional) Height in pixels of generated thumbnail.

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

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

        Parameters

- **id** – (optional) Attachment ID as a string.

- **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 string.

2.6 cdrouter.jobs

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) String list of job IDs.

- **filter** – (optional) String list of filters.

- **type** – (optional) *union* or *inter* as string.

- **all** – (optional) Apply to all if bool *True*.

bulk_launch(*jobs=None*, *filter=None*, *all=False*)

Bulk launch a set of jobs.

Parameters

- **jobs** – *jobs.Job* list

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

- **all** – (optional) Apply to all if bool *True*.

delete(*id*)

Delete a job.

Parameters id – Job ID as string.

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 string.

Returns *jobs.Job* object

> > **Return type** *jobs.Job*

> **launch** (*resource*)
>> Launch a new job.

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

>>> **Returns** `jobs.Job` object

>>> **Return type** *jobs.Job*

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

>>> **Parameters**

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

>>>> - **type** – (optional) *union* or *inter* as string.

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

>>>> - **limit** – (optional) Limit returned list length.

>>>> - **page** – (optional) Page to return.

>>> **Returns** `jobs.Job` list

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

>> **Parameters**

>>> - **id** – (optional) Job ID as string.

>>> - **status** – (optional) Bool *True* if user is an administrator.

>>> - **options** – (optional) `jobs.Options` object

>>> - **package_id** – (optional) Package ID as string.

>>> - **package_name** – (optional) Package name as string.

>>> - **device_id** – (optional) Device ID as string.

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

>>> - **result_id** – (optional) Result ID as string.

>>> - **user_id** – (optional) User ID as string.

>>> - **created** – (optional) Job creation time as *DateTime*.

>>> - **updated** – (optional) Job last-updated time as *DateTime*.

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.

2.7 cdrouter.packages

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 string.
>
> **Returns** *packages.Analysis* object
>
> **Return type** *packages.Analysis*

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

> **Parameters ids** – String 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) String 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) String 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** – String 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 string.

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 string.

>>> • **user_ids** – User IDs as int list.

>> Returns *cdrouter.Share* list

export(*id*)
> Export a package.

>> Parameters **id** – Package ID as string.

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

get(*id*)
> Get a package.

>> Parameters **id** – Package ID as string.

>> Returns *packages.Package* object

>> Return type *packages.Package*

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

>> Parameters **id** – Package ID as string.

>> Returns *cdrouter.Share* list

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

>> Parameters

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

>>> • **type** – (optional) *union* or *inter* as string.

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

>>> • **limit** – (optional) Limit returned list length.

>>> • **page** – (optional) Page to return.

>> Returns *packages.Package* list

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

>> Parameters

>>> • **id** – (optional) Package ID as a string.

- **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 a string.

- **testlist** – (optional) Testlist as a string list.

- **extra_cli_args** – (optional) Extra CLI args as a string.

- **user_id** – (optional) User ID as a string.

- **agent_id** – (optional) Agent ID as a string.

- **config_id** – (optional) Config ID as a string.

- **result_id** – (optional) Result ID as a string (if a package snapshot).

- **device_id** – (optional) Device ID as a string.

- **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.

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

Model for CDRouter Package Options.

Parameters

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

- **loop** – (optional) Loop count as a string.

- **repeat** – (optional) Repeat count as a string.

- **maxfail** – (optional) Max fail count as a string.

- **duration** – (optional) Max testing time duration as a string.

- **wait** – (optional) Wait between tests duration as a string.

- **pause** – (optional) Bool *True* is pausing between tests.

- **shuffle** – (optional) Bool *True* if testlist is shuffled.

- **seed** – (optional) Shuffle seed as a string.

- **retry** – (optional) Retry count as a string.

- **rdelay** – (optional) Retry delay as a string.

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.

2.8 cdrouter.results

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

> **all_stats**()
> > Compute stats for all results.
> >
> > > **Returns** *results.AllStats* object
> > >
> > > **Return type** *results.AllStats*

> **bulk_copy**(*ids*)
> > Bulk copy a set of results.
> >
> > > **Parameters ids** – String list of result IDs.
> > >
> > > **Returns** *results.Result* list

> **bulk_delete**(*ids=None*, *filter=None*, *type=None*, *all=False*)
> > Bulk delete a set of results.
> >
> > > **Parameters**
> > >
> > > - **ids** – (optional) String 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) String 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** – String list of result IDs.
> > >
> > > **Return type** tuple *(io.BytesIO, 'filename')*

> **delete**(*id*)
> > Delete a result.
> >
> > > **Parameters id** – Result ID as string.

> **download_logdir_archive**(*id*, *format='zip'*, *exclude_captures=False*)
> > Download logdir archive in tgz or zip format.
> >
> > > **Parameters**
> > >
> > > - **id** – Result ID as string.
> > > - **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 string.

>   - **user_ids** – User IDs as int list.

>   **Returns** *cdrouter.Share* list

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

>   **Parameters**

>   - **id** – Result ID as string.

>   - **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 string.

>   **Returns** *results.Result* object

>   **Return type** *results.Result*

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

>   **Parameters**

>   - **id** – Result ID as string.

>   - **filename** – Logdir filename as string.

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

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

>   **Parameters id** – Result ID as string.

>   **Returns** *cdrouter.Share* list

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

>   **Parameters**

>   - **id** – Result ID as string.

- **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 string.

- **name** – Test name as string.

- **id** – Metric name as string.

    Return type string

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

    Parameters

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

- **type** – (optional) *union* or *inter* as string.

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

- **limit** – (optional) Limit returned list length.

- **page** – (optional) Page to return.

    Returns *results.Result* list

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 string.

- **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 string.

>   > • **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 string.

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

>   > **Parameters id** – Result ID as string.

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

>   > **Parameters id** – Result IDs as string list.

>   > **Returns** *results.SetStats* object

>   > **Return type** *results.SetStats*

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

>   > **Parameters id** – Result ID as string.

>   > **Returns** *results.SingleStats* object

>   > **Return type** *results.SingleStats*

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

>   > **Parameters**

>   > > • **id** – Result ID as string.

>   > > • **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 string.

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

>   > **Parameters id** – Result ID as string.

unpause(*id*)
>   Unpause a running result.

>   > **Parameters id** – Result ID as string.

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

>   **Parameters**

>   > • **id** – (optional) Result ID as a string.

>   > • **created** – (optional) Creation time as *DateTime*.

>   > • **updated** – (optional) Last-updated time as *DateTime*.

- **result** – (optional) Result as a string.

- **status** – (optional) Status as a string.

- **loops** – (optional) Loop count as an int.

- **tests** – (optional) Test count as an int.

- **passed** – (optional) Passed count as an int.

- **fail** – (optional) Failed count as an int.

- **duration** – (optional) Duration in seconds as an int.

- **size_on_disk** – (optional) Size on disk in bytes as an int.

- **starred** – (optional) Bool *True* if result is starred.

- **archived** – (optional) Bool *True* if result is archived.

- **result_dir** – (optional) Filepath to result directory as a string.

- **agent_name** – (optional) Agent name as a string.

- **package_name** – (optional) Package name as a string.

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

- **config_name** – (optional) Config name as a string.

- **package_id** – (optional) Package ID as a string.

- **device_id** – (optional) Device ID as a string.

- **config_id** – (optional) Config ID as a string.

- **user_id** – (optional) User ID as a string.

- **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

class cdrouter.results.**TestCount**(***kwargs*)

Model for CDRouter Test Counts.

>   **Parameters**
>
>   - **name** – (optional) Name as a string.
>
>   - **count** – (optional) Count as a string.

class cdrouter.results.**TestDuration**(***kwargs*)

Model for CDRouter Test Durations.

>   **Parameters**
>
>   - **name** – (optional) Name as a string.
>
>   - **duration** – (optional) Duration as a string.

class cdrouter.results.**ResultBreakdown**(***kwargs*)

Model for CDRouter Result Breakdowns.

Parameters

- **passed** – (optional) Pass count as a string.

- **failed** – (optional) Fail count as a string.

- **skipped** – (optional) Skipped count as a string.

class cdrouter.results.**TimeBreakdown**(***kwargs*)

Model for CDRouter Time Breakdowns.

Parameters

- **passed** – (optional) Pass duration as a string.

- **failed** – (optional) Fail duration as a string.

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

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

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.

class cdrouter.results.**SingleStats**(***kwargs*)

Model for CDRouter Single Results Stats.

Parameters

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

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

class cdrouter.results.**PackageCount**(***kwargs*)

Model for CDRouter Package Counts.

Parameters

- **package_name** – (optional) Package name as a string.

- **count** – (optional) Count as a string.

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

> > **Parameters**

> > > • **`device_name`** – (optional) Device name as a string.

> > > • **`count`** – (optional) Count as a string.

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

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

> > **Parameters**

> > > • **`log_file`** – (optional) Filepath to logfile as a string.

> > > • **`timestamp`** – (optional) Timestamp for metric as a *DateTime*.

> > > • **`metric`** – (optional) Metric name as a string.

> > > • **`value`** – (optional) Metric value as a string.

> > > • **`units`** – (optional) Metric units as a string.

> > > • **`result`** – (optional) Metric result as a string.

> > > • **`interface_1`** – (optional) First interface as a string.

> > > • **`interface_2`** – (optional) Second interface as a string.

> > > • **`streams`** – (optional) Stream count as a string.

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*.

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.

class cdrouter.results.**Result**(***kwargs*)

Model for CDRouter Results.

> **Parameters**
>
> - **id** – (optional) Result ID as a string.
> - **created** – (optional) Creation time as *DateTime*.
> - **updated** – (optional) Last-updated time as *DateTime*.
> - **result** – (optional) Result as a string.
> - **status** – (optional) Status as a string.
> - **loops** – (optional) Loop count as an int.
> - **tests** – (optional) Test count as an int.
> - **passed** – (optional) Passed count as an int.
> - **fail** – (optional) Failed count as an int.
> - **duration** – (optional) Duration in seconds as an int.
> - **size_on_disk** – (optional) Size on disk in bytes as an int.
> - **starred** – (optional) Bool *True* if result is starred.
> - **archived** – (optional) Bool *True* if result is archived.
> - **result_dir** – (optional) Filepath to result directory as a string.
> - **agent_name** – (optional) Agent name as a string.
> - **package_name** – (optional) Package name as a string.
> - **device_name** – (optional) Device name as a string.
> - **config_name** – (optional) Config name as a string.
> - **package_id** – (optional) Package ID as a string.
> - **device_id** – (optional) Device ID as a string.
> - **config_id** – (optional) Config ID as a string.
> - **user_id** – (optional) User ID as a string.
> - **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

2.9 cdrouter.testresults

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

Service for accessing CDRouter TestResults.

edit (*id*, *resource*)
   Edit a test result.

>   **Parameters**
>
>   - **id** – Result ID as string.
>
>   - **resource** – *testresults.TestResult* object
>
>   **Returns** *testresults.TestResult* object
>
>   **Return type** *testresults.TestResult*

get (*id*, *seq*)
   Get a test result.

>   **Parameters**
>
>   - **id** – Result ID as string.
>
>   - **seq** – TestResult sequence ID as string.
>
>   **Returns** *testresults.TestResult* object
>
>   **Return type** *testresults.TestResult*

get_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 string.
>
>   - **seq** – TestResult sequence ID as string.
>
>   - **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*

get_log_plaintext (*id*, *seq*)
   Get a test result's log as plaintext.

>   **Parameters**
>
>   - **id** – Result ID as string.
>
>   - **seq** – TestResult sequence ID as string.
>
>   **Return type** string

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

>   **Parameters**
>
>   - **id** – Result ID as string.
>
>   - **filter** – (optional) Filters to apply as a string list.

- **type** – (optional) *union* or *inter* as string.

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

- **limit** – (optional) Limit returned list length.

- **page** – (optional) Page to return.

Returns *testresults.TestResult* list

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 string.

- **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

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

Parameters

- **id** – (optional) Result ID as string.

- **seq** – (optional) TestResult sequence ID as string.

- **loop** – (optional) Loop number as string.

- **result** – (optional) Test result as string.

- **retries** – (optional) Retry count as int.

- **started** – (optional) Test start time as *DateTime*.

- **duration** – (optional) Test duration as int.

- **flagged** – (optional) *True* if test is flagged.

- **name** – (optional) Test name as string.

- **description** – (optional) Test description as string.

- **skip_name** – (optional) Skip name for TestResult as string.

- **skip_reason** – (optional) Skip reason for TestResult as string.

- **log** – (optional) Logfile path for TestResult as string.

- **note** – (optional) Note for TestResult as string.

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

Parameters

- **errors** – (optional) Error log count as int.

- **fails** – (optional) Fail log count as int.

- **passes** – (optional) Pass log count as int.

- **warning** – (optional) Warning log count as int.

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

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

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.

2.10 cdrouter.annotations

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

> **create**(*id*, *seq*, *resource*)
>   Create an annotation.
>
> > **Parameters**
> >
> > - **id** – Result ID as string.
> >
> > - **seq** – TestResult sequence ID as string.
> >
> > - **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 string.

>   >   >   • **seq** – TestResult sequence ID as string.

>   >   >   • **resource** – `annotations.Annotation` object

>   >   **Returns** `annotations.Annotation` object

>   >   **Return type** *annotations.Annotation*

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

>   >   **Parameters**

>   >   >   • **id** – Result ID as string.

>   >   >   • **seq** – TestResult sequence ID as string.

>   >   >   • **line** – Line number in TestResult's logfile as string.

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

>   >   **Parameters**

>   >   >   • **id** – Result ID as string.

>   >   >   • **seq** – TestResult sequence ID as string.

>   >   >   • **resource** – `annotations.Annotation` object

>   >   **Returns** `annotations.Annotation` object

>   >   **Return type** *annotations.Annotation*

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

>   >   **Parameters**

>   >   >   • **id** – Result ID as string.

>   >   >   • **seq** – TestResult sequence ID as string.

>   >   >   • **line** – Line number in TestResult's logfile as string.

>   >   **Returns** `annotations.Annotation` object

>   >   **Return type** *annotations.Annotation*

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

>   >   **Parameters**

>   >   >   • **id** – Result ID as string.

>   >   >   • **seq** – TestResult sequence ID as string.

>   >   **Returns** `annotations.Annotation` list

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

>   **Parameters**

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

2.11 cdrouter.captures

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

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

>   **Parameters**

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

>   **Returns** *captures.ASCII* object

>   **Return type** *captures.ASCII*

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

>   **Parameters**

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

>   **Returns** *captures.Decode* object

>   **Return type** *captures.Decode*

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

>   **Parameters**

>   - **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

- **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 string.

- **seq** – TestResult sequence ID as string.

- **intf** – Interface name as string.

Returns *captures.Capture* object

Return type *captures.Capture*

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

Parameters

- **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

Returns *captures.Capture* list

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

Parameters

- **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

- **intf** – Interface name as string.

- **inline** – (optional) Use inline version of capture file.

Returns *captures.CloudShark* object

Return type *captures.CloudShark*

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

Parameters

- **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

- **intf** – Interface name as string.

- **filter** – (optional) PCAP filter to apply as string.

- **inline** – (optional) Use inline version of capture file.

Returns *captures.Summary* object

Return type *captures.Summary*

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

   Parameters

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

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

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

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

   Parameters **sections** – (optional) [*captures.Section*](#) list

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

   Parameters **sections** – (optional) [*captures.Section*](#) list

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

   Parameters

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

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

   Parameters

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

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

   Parameters

   - **name** – (optional) Field name as string.
   - **pos** – (optional) Field position as string.
   - **show** – (optional) Either *true* or *false* as string.

- **show_name** – (optional) Either *true* or *false* as string.

- **value** – (optional) Field value as string.

- **size** – (optional) Field size as string.

- **fields** – (optional) *captures.Field* list

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

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

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

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

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

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

    Parameters

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

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

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

    Parameters

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

- **offset** – (optional) Line offset as string.

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

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

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

    Parameters

- **name** – (optional) Frame name as string.

- **lines** – (optional) *captures.ASCIILine* list

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

    Parameters

- **frame** – (optional) *captures.ASCIIFrame* list

- **reassembled** – (optional) *captures.ASCIIFrame* list

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

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

2.12 cdrouter.highlights

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

> **create**(*id*, *seq*, *resource*)
> > Create a highlight.
> >
> > **Parameters**
> > > - **id** – Result ID as string.
> > > - **seq** – TestResult sequence ID as string.
> > > - **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 string.
> > > - **seq** – TestResult sequence ID as string.
> > > - **resource** – *highlights.Highlight* object
> >
> > **Returns** *highlights.Highlight* object
> >
> > **Return type** *highlights.Highlight*

> **delete**(*id*, *seq*, *line*)
> > Delete a highlight.
> >
> > **Parameters**
> > > - **id** – Result ID as string.
> > > - **seq** – TestResult sequence ID as string.
> > > - **line** – Line number in TestResult's logfile as string.

> **edit**(*id*, *seq*, *resource*)
> > Edit a highlight.
> >
> > **Parameters**
> > > - **id** – Result ID as string.
> > > - **seq** – TestResult sequence ID as string.
> > > - **resource** – *highlights.Highlight* object
> >
> > **Returns** *highlights.Highlight* object
> >
> > **Return type** *highlights.Highlight*

> **get**(*id*, *seq*, *line*)
> > Get a highlight.
> >
> > **Parameters**
> > > - **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

- **line** – Line number in TestResult's logfile as string.

  Returns *highlights.Highlight* object

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

  Parameters

- **id** – Result ID as string.

- **seq** – TestResult sequence ID as string.

  Returns *highlights.Highlight* list

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

  Parameters

- **id** – (optional) Result ID as string.

- **seq** – (optional) TestResult sequence ID as string.

- **line** – (optional) Line number in TestResult's logfile as string.

- **color** – (optional) Highlight color as string.

2.13 cdrouter.imports

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

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

  Parameters

- **id** – Staged import ID as string.

- **impreq** – *imports.Request* object

  Returns *imports.Request* object

  Return type *imports.Request*

  delete(*id*)
      Delete a staged import.

  Parameters **id** – Staged import ID as string.

  get(*id*)
      Get a staged import.

  Parameters **id** – Staged import ID as string.

  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 string.

> **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*, *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).
> >
> > - **insecure** – (optional) Allow insecure HTTPS connections if bool *True*.
> >
> > **Returns** `imports.Import` object

class cdrouter.imports.**Import**(***kwargs*)
> Model for CDRouter Staged Imports.
>
> > **Parameters**
> >
> > - **id** – (optional) Staged import ID as string.
> >
> > - **user_id** – (optional) User ID as string.
> >
> > - **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.

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

- **imported** – (optional) Bool *True* if resource imported successfully.

- **id** – (optional) Resource ID as string.

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

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

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 of existing resource which will be overwritten if *should_import* is *True*.

- **response** – (optional) *imports.Response* object

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.

2.14 cdrouter.exports

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')*

2.15 cdrouter.history

class `cdrouter.history.`**`HistoryService`**(*service*)

> Service for accessing CDRouter History.

> **`list`**(*filter=None*, *type=None*, *sort=None*, *limit=None*, *page=None*)
>> Get a list of history entries.
>>
>>> **Parameters**
>>>
>>>> - **`filter`** – (optional) Filters to apply as a string list.
>>>> - **`type`** – (optional) *union* or *inter* as string.
>>>> - **`sort`** – (optional) Sort fields to apply as string list.
>>>> - **`limit`** – (optional) Limit returned list length.
>>>> - **`page`** – (optional) Page to return.
>>>
>>> **Returns** *history.History* list

class `cdrouter.history.`**`History`**(***kwargs*)

> Model for CDRouter History entries.

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

2.16 cdrouter.system

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

> Service for accessing CDRouter System.

> **`check_for_lounge_upgrade`**(*email*, *password*)
>> Check the CDRouter Support Lounge for eligible upgrades using your Support Lounge email & password.
>>
>>> **Parameters**
>>>
>>>> - **`email`** – CDRouter Support Lounge email as a string.
>>>> - **`password`** – CDRouter Support Lounge password as a string.
>>>
>>> **Returns** *system.Release* object
>>>
>>> **Return type** *system.Release*

> **`diagnostics`**()
>> Get system diagnostics from cdrouter-diag output.
>>
>>> **Return type** string

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

> **Parameters resource** – *system.Preferences* object
>
> **Returns** *system.Preferences* object
>
> **Return type** *system.Preferences*

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

> **Returns** *system.Preferences* object
>
> **Return type** *system.Preferences*

hostname()
Get system hostname.

> **Return type** string

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

> **Return type** string

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

> **Parameters addresses** – (optional) If bool *True*, include interface addresses.
>
> **Returns** *system.Interface* list

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

> **Returns** *system.ReleaseLatest* object
>
> **Return type** *system.ReleaseLatest*

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

> **Return type** string

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

> **Returns** *system.Upgrade* object
>
> **Return type** *system.Upgrade*

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

> **Parameters**
>
> - **email** – CDRouter Support Lounge email as a string.
>
> - **password** – CDRouter Support Lounge password as a string.
>
> - **release_id** – Release ID as an int.
>
> **Returns** *system.Upgrade* object
>
> **Return type** *system.Upgrade*

manual_update_license(*fd*, *filename='cdrouter.lic'*)
> Update the license on your CDRouter system manually by uploading a .lic license from the CDRouter Support Lounge.
>
> > **Parameters**
> >
> > - **fd** – File-like object to upload.
> >
> > - **filename** – (optional) Filename to use for license as string.
> >
> > **Returns** *system.Upgrade* object
> >
> > **Return type** *system.Upgrade*

manual_upgrade(*fd*, *filename='cdrouter.bin'*)
> Upgrade CDRouter manually by uploading a .bin installer from the CDRouter Support Lounge. Please note that any running tests will be stopped.
>
> > **Parameters**
> >
> > - **fd** – File-like object to upload.
> >
> > - **filename** – (optional) Filename to use for installer as string.
> >
> > **Returns** *system.Upgrade* object
> >
> > **Return type** *system.Upgrade*

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

time()
> Get system time.
>
> > **Return type** string

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.

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.

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.

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

Parameters

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

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.

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.

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.

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

Parameters

- **index** – (optional) Interface index as a string.
- **mtu** – (optional) Interface MTU as a string.
- **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

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

　　Model for CDRouter Preferences.

　　　　Parameters

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

2.17 cdrouter.tags

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

　　Service for accessing CDRouter Tags.

　　delete(*name*)

　　　　Delete a tag.

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

　　edit(*resource*)

　　　　Edit a tag.

> > > > Parameters **resource** – *tags.Tag* object
> > > >
> > > > Returns *tags.Tag* object
> > > >
> > > > Return type *tags.Tag*

> > **get**(*name*)
> > Get a tag.
> >
> > > Parameters **name** – Tag name as string.
> > >
> > > Returns *tags.Tag* object
> > >
> > > Return type *tags.Tag*

> > **list**(*resource=None*, *sort=None*)
> > Get a list of tags.
> >
> > > Parameters
> > >
> > > > • **resource** – (optional) Restrict to given resource type as string.
> > > >
> > > > • **sort** – (optional) Sort fields to apply as string list.
> > >
> > > Returns *tags.Tag* list

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

> Parameters
>
> > • **resource** – (optional) Resource type as string.
> >
> > • **id** – (optional) Resource ID as string.
> >
> > • **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

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

> Parameters
>
> > • **id** – (optional) Resource ID as string.
> >
> > • **name** – (optional) Resource name as string.
> >
> > • **tags** – (optional) Resource tags as a string list.

2.18 cdrouter.testsuites

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

get_error(*name*)
> Get an error.

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

>> Returns *testsuites.Error* object

>> Return type *testsuites.Error*

get_group(*name*)
> Get a group.

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

>> Returns *testsuites.Group* object

>> Return type *testsuites.Group*

get_label(*name*)
> Get a label.

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

>> Returns *testsuites.Label* object

>> Return type *testsuites.Label*

get_module(*name*)
> Get a module.

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

>> Returns *testsuites.Module* object

>> Return type *testsuites.Module*

get_test(*name*)
> Get a test.

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

>> Returns *testsuites.Test* object

>> Return type *testsuites.Test*

get_testvar(*name*)
> Get a testvar.

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

>> Returns *testsuites.Testvar* object

>> Return type *testsuites.Testvar*

info()
> Get testsuite info.

>> Returns *testsuites.Info* object

>> Return type *testsuites.Info*

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

>> Parameters

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

>>> • **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Error* list

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

    Parameters

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

    - **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Group* list

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

    Parameters

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

    - **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Label* list

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

    Parameters

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

    - **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Module* list

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

    Parameters

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

    - **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Test* list

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

    Parameters

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

    - **type** – (optional) *union* or *inter* as string.

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

    Returns *testsuites.Testvar* list

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

> **Parameters**   **query** – Search query as a string.
>
> **Returns**   *testsuites.Search* object
>
> **Return type**   *testsuites.Search*

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.

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.

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.

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.

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.

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.

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.

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

2.19 cdrouter.users

class cdrouter.users.**UsersService**(*service*)

Service for accessing CDRouter Users.

---

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

> > **Parameters ids** – String 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) String 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) String 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 string.

> > > • **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 string.

> > **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 string.

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 string.

>> Returns *users.User* object

>> Return type *users.User*

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

>> Parameters

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

>>> • **type** – (optional) *union* or *inter* as string.

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

>>> • **limit** – (optional) Limit returned list length.

>>> • **page** – (optional) Page to return.

>> Returns *users.User* list

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

>> Parameters

>>> • **id** – (optional) User ID as string.

>>> • **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.

C
cdrouter, 3

A

B

C

V