



cpm\_api\_client

API Version: 1.3.0

# Table of Contents

1. Overview .....	1
1.1. Usage .....	1
1.2. API Commands .....	1
1.3. Authentication .....	3
1.4. Media Type .....	3
1.5. Versioning .....	3
1.6. Configuration File .....	3
2. Examples .....	5
2.1. Schedules APIs .....	5
2.2. Logs APIs .....	8
2.3. Reports APIs .....	9

# Chapter 1. Overview

This guide describes how to use *cpm\_api\_client* a command line client for the *CPM RESTful API* server.

Currently the following operation systems are supported:

- Windows 7 x86 and above
- Ubuntu 14.04.5 and above

For more information regarding the *CPM RESTful API*, refer to the [CPM RESTful API User Guide](#).



*cpm\_api\_client* works with CPM version 2.5.0 and above.

## 1.1. Usage

To see list of supported commands run *cpm\_api\_client* without any arguments or with **-h/--help**.

```
cpm_api_client -h
```

```
usage: cpm_api_client [-c] [-h] <COMMAND> ...
```

```
N2W Software command line client (API version 1.3.0)
```

```
Optional Arguments:
```

```
-c, --color          If set, print output with colors (only applicable for  
ANSI/VT100 terminals and terminal emulators).  
-h, --help          Show this help message and exit.
```

```
API commands:
```

```
<COMMAND>
```

## 1.2. API Commands

All the APIs which are supported by the *CPM RESTful API* can be invoked from *cpm\_api\_client*.

When a command is invoked an HTTP request is sent to the *CPM RESTful API* server with all the supplied arguments converted to the proper API parameters.

To see available arguments for a specific command run *cpm\_api\_client* with **-h/--help**.

```
cpm_api_client <COMMAND> -h
```

In order to be compliant with *CPM RESTful API User Guide* the available arguments are divided to the following sections:

### Optional Arguments

Arguments which are optional for *cpm\_api\_client* and are not specific for the invoked command.

```
-c, --color          If set, print output with colors (only applicable for
                    ANSI/VT100 terminals and terminal emulators).
-h, --help          Show this help message and exit.
```

### Connection Arguments

Arguments which are needed to establish connection to the *CPM RESTful API* server.

```
--host HOST          HOST of the RESTful API server.
--skip_ssl_verification
                    If set, skip server side SSL certificate verification.
--request_timeout REQUEST_TIMEOUT
                    Timeout in seconds for API request (default: 60 seconds).
```

### API Arguments

Arguments which are part of the API request **Accept** headers. See [Media Type](#) and [Versioning](#) for more information.

```
--api_version API_VERSION
                    API version to be used (default: 1.3.0).
--media_type MEDIA_TYPE
                    Media type for the response (default: application/json).
```

### Authentication Arguments

Arguments needed for authentication. See [Authentication](#) for more information.

```
--access_token ACCESS_TOKEN
                    Access Token to authenticate with the CPM server.
```

### Query Arguments

Arguments which are converted into URL parameters for the invoked API request.

### Body Arguments

Arguments which are converted into JSON and are sent as the body of the invoked API request.

## 1.3. Authentication

**CPM RESTful API** authentication scheme requires passing an access token on each request. Without the access token, the request will be denied with the **HTTP 401 Unauthorized** error response and appropriate **WWW-Authenticate** header.

In order to authenticate **cpm\_api\_client** commands **--access\_token** must be provided.

### Obtain Tokens:

```
cpm_api_client token_obtain_api_key_create --host <HOST of the RESTful API server> --api_key  
<The API Authentication Key>
```

### Refresh Access Token:

```
cpm_api_client token_refresh_create --host <HOST of the RESTful API server> --refresh <The  
Refresh Token>
```



For more information regarding authentication, refer to the *CPM RESTful API User Guide*.

## 1.4. Media Type

**CPM RESTful API** supports 2 types of media types in the **Accept** header **application/json** and **text/csv**.

By default, **cpm\_api\_client** is sending **application/json** in **Accept** header for most of the commands. **text/csv** is sent for reports and logs.

**--media\_type** can be provided in order to change the media type for a command.

## 1.5. Versioning

In order to support backward compatibility, **CPM RESTful API** is versioned.

By default, **cpm\_api\_client** is sending the latest API version for all the commands.

**--api\_version** can be provided in order to change the API version for a command.

## 1.6. Configuration File

**cpm\_api\_client** enables using a configuration file named **cpm\_api\_client.cfg**, the file must be located in the same directory where the **cpm\_api\_client** binary is located and a default one is already provided.

**cpm\_api\_client.cfg** contains two sections:

**[arguments]**

Arguments that can be used with the invoked command and are used as default values if not provided when the command is invoked.

If ***cpm\_api\_client*** is invoked with arguments the values from ***cpm\_api\_client.cfg*** are ignored.

**[logging]**

Provides logging arguments. Should not be modified unless requested by support team.

*Default Configuration File*

```
[arguments]
host=127.0.0.1
skip_ssl_verification=no
request_timeout=60
color=no
;access_token=<your token here>

[logging]
log_level=INFO
```



***access\_token*** can be added within [arguments] section. In the provided default configuration file it is commented out using ***;***.

## Chapter 2. Examples

Below are sample commands which use the following variables:

Variable	Description
\$HOST	RESTful API server host.
\$TOKEN	Access token for authentication.

### 2.1. Schedules APIs

Below are sample commands to create a schedule, update it, show a list of schedules, and delete the created schedule.

#### 2.1.1. Create a CPM Schedule

This sample shows how to create a weekly CPM schedule for the *root* user.

##### Sample Request:

```
cpm_api_client schedules_create --host $HOST --access_token $TOKEN --name sample --every_unit W --every_how_many 1 --user 1
```

##### Sample Response:

```
{
  "content_type": "application/json",
  "response": {
    "description": null,
    "disable_times": [],
    "end_date": null,
    "every_how_many": 1,
    "every_unit": "W",
    "id": 1,
    "last_modified": "2018-08-01T12:00:00.000000Z",
    "name": "sample",
    "start_date": "2018-08-01T12:00:00.000000Z",
    "user": "1"
  },
  "status": 201
}
```

#### 2.1.2. Update a CPM Schedule

This sample shows how to update a CPM schedule with *id=1* to run once a month.

**Sample Request:**

```
cpm_api_client schedules_update --host $HOST --access_token $TOKEN --name sample --id 1
--every_unit 0 --every_how_many 1
```

**Sample Response:**

```
{
  "content_type": "application/json",
  "response": {
    "description": null,
    "disable_times": [],
    "end_date": null,
    "every_how_many": 1,
    "every_unit": "0",
    "id": 1,
    "last_modified": "2018-08-01T12:00:00.000000Z",
    "name": "sample",
    "start_date": "2018-08-01T13:00:00.000000Z",
    "user": 1
  },
  "status": 200
}
```

**2.1.3. List CPM Schedules**

This sample shows how to get a list of all CPM schedules ordered by the scheduling frequency.

**Sample Request:**

```
cpm_api_client schedules_list --host $HOST --access_token $TOKEN --ordering
every_unit, every_how_many
```

**Sample Response:**

```
{
  "content_type": "application/json",
  "response": [
    {
      "id": 2,
      "name": "5minutes",
      "description": "",
      "user": 1,
      "every_unit": "M",
      "every_how_many": 5,
      "start_date": "2018-08-20T12:50:00Z",
    }
  ]
}
```



```

    "end_date": null,
    "last_modified": "2018-08-20T11:50:35Z",
    "allow_on_sunday": true,
    "allow_on_monday": true,
    "allow_on_tuesday": true,
    "allow_on_wednesday": true,
    "allow_on_thursday": true,
    "allow_on_friday": true,
    "allow_on_saturday": true,
    "disable_times": []
  },
  {
    "id": 5,
    "name": "10hours",
    "description": "",
    "user": 1,
    "every_unit": "H",
    "every_how_many": 10,
    "start_date": "2018-08-20T13:11:00Z",
    "end_date": null,
    "last_modified": "2018-08-20T12:11:21Z",
    "allow_on_sunday": true,
    "allow_on_monday": true,
    "allow_on_tuesday": true,
    "allow_on_wednesday": true,
    "allow_on_thursday": true,
    "allow_on_friday": true,
    "allow_on_saturday": true,
    "disable_times": []
  },
  {
    "id": 4,
    "name": "7weeks",
    "description": "",
    "user": 1,
    "every_unit": "W",
    "every_how_many": 7,
    "start_date": "2018-08-20T12:50:00Z",
    "end_date": null,
    "last_modified": "2018-08-20T11:51:03Z",
    "disable_times": []
  },
  {
    "id": 1,
    "name": "1month",
    "description": "",
    "user": 1,
    "every_unit": "0",

```

```

    "every_how_many": 1,
    "start_date": "2018-08-20T14:43:56Z",
    "end_date": null,
    "last_modified": "2018-08-20T11:43:56Z",
    "disable_times": []
  },
  {
    "id": 3,
    "name": "2months",
    "description": "",
    "user": 1,
    "every_unit": "0",
    "every_how_many": 2,
    "start_date": "2018-08-20T12:50:00Z",
    "end_date": null,
    "last_modified": "2018-08-20T11:50:50Z",
    "disable_times": []
  }
],
"status": 200
}

```

### 2.1.4. Delete a CPM Schedule

This sample shows how to delete a CPM schedule with *id=1*.

**Sample Request:**

```
cpm_api_client schedules_delete --host $HOST --access_token $TOKEN --id 1
```

**Sample Response:**

```

{
  "content_type": "application/json",
  "response": null,
  "status": 204
}

```

## 2.2. Logs APIs

### 2.2.1. Download Support Logs

This sample downloads support logs as *bz2* file into current working directory.

**Sample Request:**

```
cpm_api_client system_support_logs_list --host $HOST --access_token $TOKEN
```

**Sample Response:**

```
{
  "content_type": "application/x-bzip2",
  "response": "attachment; filename=cpm_logs_2018_12_16_15_32.tar.bz2",
  "status": 200
}
```

## 2.2.2. Download Cleanup Logs

This sample downloads cleanup logs as CSV file into current working directory.

**Sample Request:**

```
cpm_api_client settings_cleanup_list_logs --host $HOST --access_token $TOKEN
```

**Sample Response:**

```
{
  "content_type": "text/csv; charset=utf-8",
  "response": "attachment; filename=cleanup_log_2018-12-17_07-16-22.497640.csv",
  "status": 200
}
```

## 2.3. Reports APIs

### 2.3.1. Download Snapshots Report

This sample downloads snapshots report as CSV file into current working directory.

**Sample Request:**

```
cpm_api_client reports_snapshots_list --host $HOST --access_token $TOKEN
```

**Sample Response:**

```
{
  "content_type": "text/csv; charset=utf-8",
  "response": "attachment; filename=cpm_snapshots_report_for_user_root_2018-12-17_10-17.csv",
  "status": 200
}
```

### 2.3.2. Download Backups Report

This sample downloads backups report with *backup\_id=1* as CSV file into current working directory.

**Sample Request:**

```
cpm_api_client reports_backups_list --host $HOST --access_token $TOKEN --backup_id 1
```

**Sample Response:**

```
{
  "content_type": "text/csv; charset=utf-8",
  "response": "attachment; filename=cpm_backup_report_for_user_root_2018-12-17_10-40.csv",
  "status": 200
}
```