
Hook Worker Documentation

Release 0.0.1

Thibault Clérice

December 20, 2016

1	Commands	3
1.1	HookWorker API	3
1.2	Deploying Hook-Worker	5

Lightweight API to deploy on the machine distributing the test on a redis connection

Commands

```
hookworker-api [-h] [-p PORT] [-d DOMAIN] [-g DEBUG] [-s SECRET] [-n] [-o PATH] [-e LEVEL] [-r REDIS] [-w]
```

Parameter / Syntax	Description	Concerns
-p PORT, --port	PORT Port to use to run the API	Redis
-g DEBUG, --debug	DEBUG Debug mode for the API	API
-s SECRET, --secret	SECRET Secret used to secure data exchanges in Hook	API
-o PATH, --path	PATH Path to use for storing logs	API
-e LEVEL, --level	LEVEL Level of logging	API
-r REDIS, --redis	REDIS Redis address	Redis
-w WORKER, --workers	Number of workers to use for HookTest	API
-g GIT, --git	Path where you would like to do the cloning	API
-h, --help	show this help message and exit	
-q, --rq	Run the worker	Redis
-a, --api	Run the api	API

1.1 HookWorker API

1.1.1 Rest API

```
class HookWorker.api.WorkerAPI (prefix='hook', secret='', workers=5, redis='127.0.0.1', hook-
                                test_path='./', app=None, name=None)
```

Worker API for Capitains Hook

Parameters

- **prefix** – Prefix for the WorkerAPI
- **secret** – Salt to use in encrypting the body
- **workers** – Number of workers to use in HookTest
- **redis** – Redis Connection URL
- **hooktest_path** – Path where to clone the data
- **app** – Application object
- **name** – Name of the Blueprint

Variables routes – Liste of tuples to store routes (url, function name, [Methods])

`WorkerAPI.init_app(app)`

Register the blueprint to the app

Parameters `app` – Flask Application

Returns Blueprint for HookWorker registered in app

Return type Blueprint

Controller

`WorkerAPI.get_queue()`

Get Redis' queue

Returns Queue of Redis

`WorkerAPI.check_signature(body, foreign_signature)`

Check the signature sent by a request with the body

Parameters

- **body** – Raw body of the request
- **foreign_signature** – Signature sent by Hook

Returns Security check status

Return type `bool`

Routes

`WorkerAPI.r_delete(id)`

Remove a test from the testing queue

Parameters `id` – Job id to cancel

Returns Json response with status and message

`WorkerAPI.r_submit()`

Dispatch a test to the redis queue

Returns Response with a status and the `job_id` if everything worked

Server Helper

`HookWorker.api.set_logging(level, name, path, logger)`

Reroute logging of tornado into specified file with a RotatingFileHandler

Parameters

- **level** (`str`) – Level of logging
- **name** – Name of logs file
- **path** – Path where to store the logs file
- **logger** – logging.logger object of Tornado

`HookWorker.api.run(secret='', debug=False, port=5000, path='./hook.worker.api.log',
level='WARNING', git='./hooktest', workers=5)`

Set up a Tornado process around a flask app for quick run of the WorkerAPI Blueprint

Parameters

- **secret** – Salt to use in encrypting the body
- **debug** – Set Flask App in debug Mode
- **port** – Port to use for Flask App
- **path** – Path where to store the logs
- **level** – Level of Log
- **git** – Path where to clone the data
- **workers** – Number of worker to use for HookTest runs

1.1.2 Worker

The worker is a simpler Python RQ based worker

```
HookWorker.worker.worker(redis_url='localhost:6379')
```

Run a work for python-rq

Parameters `redis_url` (*str*) – Redis URI (redis://{redis_url})

1.2 Deploying Hook-Worker

1.2.1 Pre-requisite

Warning: you might need to sudo all following commands

Python and Python tools

- Make sure to have python 2.7 **and** python 3
 - `apt-get install python3 python2`
- Make sure to have pip installed for both python 2 and 3
 - Ubuntu 12.10 and later > `apt-get install python3-pip`
 - [Ubuntu 12.04](#) `apt-get install python3-setuptools` **and** `easy_install3 pip`
- Install virtualenv for python3
 - `pip3 install virtualenv`
- **Install development dependencies**
 - `sudo apt-get install python-dev libxml2-dev libxslt-dev`

Redis

- [Install Redis](#)

Supervisor

- Install through with pip
 - pip install supervisor
- Configure supervisor
 - Default configuration can be created with `echo_supervisord_conf > /etc/supervisord.conf`

Configuration of supervisor

DO NOT FORGET TO CHANGE THE PASSWORD

```
[unix_http_server]
file=/tmp/supervisor.sock          ; (the path to the socket file)
;chmod=0700                        ; socket file mode (default 0700)
chown=capitainshook:nogroup        ; socket file uid:gid owner
username=capitainshook             ; (default is no username (open server))
password=capitains                 ; (default is no password (open server))

[supervisord]
logfile=/var/log/supervisord.log ; (main log file;default $CWD/supervisord.log)
user=capitainshook                ; (default is current user, required if root)

[rpcinterface:supervisor]
supervisor.rpcinterface_factory = supervisor.rpcinterface:make_main_rpcinterface

[supervisorctl]
serverurl=unix:///tmp/supervisor.sock ; use a unix:// URL  for a unix socket
username=capitainshook              ; should be same as http_username if set
password=capitains                  ; should be same as http_password if set
```

1.2.2 Hook Worker

Code base

We advise to set-up a user account only for Hook. This will relieve you from many security flaws :

- `adduser capitainshook`
- Set-up the password and remember it
- `su capitainshook`
- `cd /home/capitainshook/`
- `virtualenv --python=/usr/bin/python3 venv (or virtualenv-3.4)`
- `source venv/bin/activate`
- *For logs*: `mkdir -R logs/api`
- **Using pip**
 - `pip install HookWorker`
- **Using git**
 - `git clone https://github.com/Capitains/Hook-Worker.git worker`

```
- cd worker
- python setup.py install
```

Services

Add the following to your supervisord.conf

The secret should be set to the secret key used by authorized callers of the HookWorker API

```
[program:hookworkerapi]
command=/home/captainshook/venv/bin/hookworker-api --api --path /home/captainshook/logs/api/ --port
stderr_logfile=/home/captainshook/logs/api.error.log

[program:hookworkerrq]
command=/home/captainshook/venv/bin/hookworker-api --rq --redis 127.0.0.1:6379
stdout_logfile=/home/captainshook/logs/worker.log
stderr_logfile=/home/captainshook/logs/worker.error.log
```


C

`check_signature()` (`HookWorker.api.WorkerAPI` method), [4](#)

G

`get_queue()` (`HookWorker.api.WorkerAPI` method), [4](#)

I

`init_app()` (`HookWorker.api.WorkerAPI` method), [3](#)

R

`r_delete()` (`HookWorker.api.WorkerAPI` method), [4](#)

`r_submit()` (`HookWorker.api.WorkerAPI` method), [4](#)

`run()` (in module `HookWorker.api`), [4](#)

S

`set_logging()` (in module `HookWorker.api`), [4](#)

W

`worker()` (in module `HookWorker.worker`), [5](#)

`WorkerAPI` (class in `HookWorker.api`), [3](#)