
pyArchOps/helpers Documentation

Release 0.0.8

Azul

Jun 02, 2019

Contents:

1	pyArchOps/helpers	1
1.1	Features	1
1.2	Installation	1
1.3	Usage	1
1.4	Development	2
1.5	Credits	2
2	Installation	3
2.1	Stable release	3
2.2	Latest helpers release	3
2.3	From sources	3
3	Usage	5
4	pyarchops_helpers	7
4.1	pyarchops_helpers package	7
5	Contributing	9
5.1	Types of Contributions	9
5.2	Get Started!	10
5.3	Pull Request Guidelines	10
5.4	Deploying	11
6	Credits	13
6.1	Development Lead	13
6.2	Contributors	13
7	History	15
7.1	0.0.1 (2018-12-11)	15
8	Indices and tables	17
	Python Module Index	19
	Index	21

CHAPTER 1

pyArchOps/helpers

Helpers for pyArchOps

- Free software: MIT license
- Documentation: <https://pyarchops-helpers.readthedocs.io>.

1.1 Features

- docker based test helpers

1.2 Instalation

```
$ pip install pyarchops-helpers
```

1.3 Usage

```
from suitable import Api
from pyarchops_helpers import helpers

with helpers.ephemeral_docker_container(
    image='registry.gitlab.com/pyarchops/pyarchops-base'
) as container:
    connection_string = "{}:{}".format(
```

(continues on next page)

(continued from previous page)

```
        container['ip'], container['port']
    )
    print('connection strings is ' + connection_string)
    api = Api(connection_string,
               connection='smart',
               remote_user=container['user'],
               private_key_file=container['pkey'],
               become=True,
               become_user='root',
               sudo=True,
               ssh_extra_args='-o StrictHostKeyChecking=no')

    try:
        result = api.setup()['contacted'][connection_string]
    except Exception as error:
        raise error

    assert result['ansible_facts']
```

1.4 Development

Install requirements:

```
$ sudo pacman -S tmux python-virtualenv python-pip libjpeg-turbo gcc make vim git tk_
↪tcl
```

Git clone this repository

```
$ git clone https://github.com/pyarchops/helpers.git pyarchops.helpers
$ cd pyarchops.helpers
```

2. See the *Makefile*, to get started simply execute:

```
$ make up
```

1.5 Credits

- TODO

2.1 Stable release

The pyArchOps software should be installed through the main repository, .. _pyArchOps : <https://github.com/pyarchops/pyarchops.git>

```
$ pip install pyarchops
```

2.2 Latest helpers release

To install pyArchOps/helpers, run this command in your terminal:

```
$ pip install pyarchops_helpers
```

This is the preferred method to install pyArchOps/helpers, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

2.3 From sources

The sources for pyArchOps/helpers can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/pyarchops/helpers
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/pyarchops/helpers/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```


CHAPTER 3

Usage

To use pyArchOps/helpers in a project:

```
import pyarchops_helpers
```


4.1 pyarchops_helpers package

4.1.1 Submodules

4.1.2 pyarchops_helpers.helpers module

Helpers for *pyarchops tests* packages.

`pyarchops_helpers.helpers.docker_container_port(container_id: str) → int`
returns the ssh port number for a docker instance

params: string container: docker container id

returns: string: port number

`pyarchops_helpers.helpers.docker_down(container_id: str) → None`
kills the docker container

params: string container: docker id of the container to stop

`pyarchops_helpers.helpers.docker_rm(container_id: str) → None`
removes a docker container

params: string container: docker id of the container to remove

`pyarchops_helpers.helpers.docker_up(image: str, privileged: bool = False) → str`
runs a docker container

params: string image: name of the docker image bool privileged: use docker `--privileged` flag

returns: string: stdout of the docker run

`pyarchops_helpers.helpers.dockerd_ip() → str`
returns the ip address of the docker daemon

params: string docker_host_string: URL of the docker daemon

returns: string: ip address of the docker host

`pyarchops_helpers.helpers.ephemeral_docker_container(**kwargs) → Iterator[dict]`
prepares a docker container, yielding a dict with its configuration before deleting the container

`pyarchops_helpers.helpers.wait_for_ssh(host: dict, initial_wait: int = 0, interval: int = 0, retries: int = 1) → bool`
waits for ssh to become available

4.1.3 Module contents

`__init__.py` for `pyarchops_helpers`

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

5.1 Types of Contributions

5.1.1 Report Bugs

Report bugs at <https://github.com/pyArchOps/pyarchops/helpers/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

5.1.4 Write Documentation

pyArchOps/helpers could always use more documentation, whether as part of the official pyArchOps/helpers docs, in docstrings, or even on the web in blog posts, articles, and such.

5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/pyArchOps/helpers/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

5.2 Get Started!

Ready to contribute? Here's how to set up *pyarchops/helpers* for local development.

1. Fork the *pyarchops/helpers* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/pyarchops/helpers.git
```

3. set a development environment:

```
$ sudo pacman -Sy tmux libffi pkgconf base-devel make $ cd pyarchops/helpers/ $ make up
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass pylint and the tests.
6. Commit your changes and push your branch to GitHub:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.

3. The pull request should work for Python 3.7. Check <https://gitlab.com/pyarchops/helpers/pipelines> and make sure that the tests pass for all supported Python versions.

5.4 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bumpversion patch # possible: major / minor / patch
$ git push
$ git push --tags
```


6.1 Development Lead

- Azul <pyarchops@azulinho.com>

6.2 Contributors

None yet. Why not be the first?

7.1 0.0.1 (2018-12-11)

- First release on PyPI.

CHAPTER 8

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`pyarchops_helpers`, 8

`pyarchops_helpers.helpers`, 7

D

`docker_container_port()` (in module *pyarchops_helpers.helpers*), 7
`docker_down()` (in module *pyarchops_helpers.helpers*), 7
`docker_rm()` (in module *pyarchops_helpers.helpers*), 7
`docker_up()` (in module *pyarchops_helpers.helpers*), 7
`dockerd_ip()` (in module *pyarchops_helpers.helpers*), 7

E

`ephemeral_docker_container()` (in module *pyarchops_helpers.helpers*), 8

P

`pyarchops_helpers` (module), 8
`pyarchops_helpers.helpers` (module), 7

W

`wait_for_ssh()` (in module *pyarchops_helpers.helpers*), 8