
wangls Documentation

Release 0.1.6

Ahmed Ali

Feb 05, 2020

Contents:

1	wangls	1
1.1	Usage	1
2	Installation	3
2.1	Prerequisite	3
2.2	Stable release	3
2.3	From sources	3
3	Usage	5
4	wangls	7
4.1	wangls package	7
5	Contributing	9
5.1	Types of Contributions	9
5.2	Get Started!	10
5.3	Pull Request Guidelines	11
5.4	Tips	11
5.5	Deploying	11
6	History	13
6.1	0.1.0 (2020-02-04)	13
6.2	0.1.1 (2020-02-05)	13
6.3	0.1.2 (2020-02-05)	13
6.4	0.1.3 (2020-02-05)	13
6.5	0.1.4 (2020-02-05)	13
6.6	0.1.5 (2020-02-05)	13
6.7	0.1.6 (2020-02-05)	14
7	Indices and tables	15
	Python Module Index	17
	Index	19

CHAPTER 1

wangls

A command line tool that gives you the IPs of the devices connected to the same network segment.

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

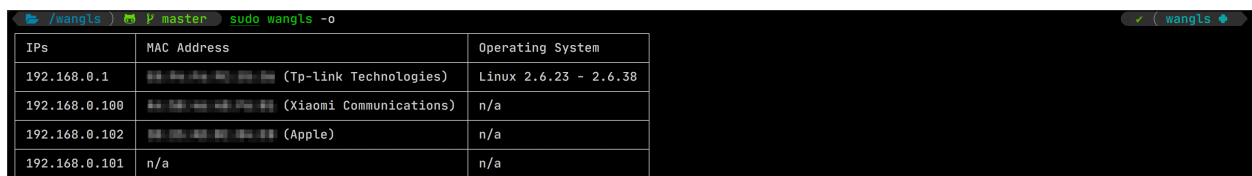
1.1 Usage

- Simple scan that returns only IPs.

```
$ wangls
```

- Scanning your whole network with OS detection might take some time especially if there are many devices connected.

```
$ wangls -o
```



IPs	MAC Address	Operating System
192.168.0.1	88:87:ae:43:43:43 (Tp-link Technologies)	Linux 2.6.23 - 2.6.38
192.168.0.100	88:87:ae:43:43:43 (Xiaomi Communications)	n/a
192.168.0.102	88:87:ae:43:43:43 (Apple)	n/a
192.168.0.101	n/a	n/a

2.1 Prerequisite

Install nmap <https://nmap.org/download.html>

2.2 Stable release

To install wangs, run this command in your terminal:

```
$ pip install wangs
```

This is the preferred method to install wangs, 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 wangs can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/ally-ahmed/wangs
```

Or download the [tarball](#):

```
$ curl -OJL https://github.com/ally-ahmed/wangs/tarball/master
```

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

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


CHAPTER 3

Usage

To use wangs in a project:

```
import wangs
```


4.1 wangls package

4.1.1 Submodules

4.1.2 wangls.cli module

4.1.3 wangls.wangls module

4.1.4 Module contents

Top-level package for wangls.

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/ally-ahmed/wangls/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

wangls could always use more documentation, whether as part of the official wangls 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/ally-ahmed/wangls/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 *wangls* for local development.

1. Fork the *wangls* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/wangls.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv wangls
$ cd wangls/
$ python setup.py develop
```

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 flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 wangls tests
$ python setup.py test or pytest
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

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 3.5, 3.6, 3.7 and 3.8, and for PyPy. Check https://travis-ci.org/ally-ahmed/wangls/pull_requests and make sure that the tests pass for all supported Python versions.

5.4 Tips

To run a subset of tests:

```
$ pytest tests.test_wangls
```

5.5 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:

```
$ bump2version patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.

6.1 0.1.0 (2020-02-04)

- First release on PyPI.

6.2 0.1.1 (2020-02-05)

- Updated demos on docs

6.3 0.1.2 (2020-02-05)

- Add dependencies requirements

6.4 0.1.3 (2020-02-05)

- Added help option

6.5 0.1.4 (2020-02-05)

- Fixed minor typos

6.6 0.1.5 (2020-02-05)

- Added version option

6.7 0.1.6 (2020-02-05)

- Updated version on option

CHAPTER 7

Indices and tables

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

W

wangls, [7](#)

W

wangls (*module*), [7](#)