lionshead documentation

Release 0.1

Nate Coraor

January 05, 2017

Contents

1 lionshead **3**

1.1 Quick start . . . 3

1.2 FAQ . . . 4

1.3 Code of Conduct . . . 4

2 Platform detection logic **5**

2.1 Linux . . . 5

3 History **7**

3.1 0.1 (2017-01-03) . . . 7

4 Indices and tables **9**

Contents:

CHAPTER 1

lionshead

A platform detection library for Python. It uses various methods to attempt to determine the operating system, distribution, release, version, etc., to aid callers in making platform-specific decisions.

lionshead was originally written for “platform-specific wheels” that were a solution to the Linux wheel problem before the creation of manylinux. Rather than be wasted effort, it’s been spun in to a separate library (which would’ve been necessary anyway to be consumed by both pip and wheel).

The development repository can be found at https://github.com/natefoo/lionshead

Documentation is available at http://lionshead.readthedocs.io

The package is published to PyPI at https://pypi.python.org/pypi/lionshead

1.1 Quick start

On Debian stretch/sid:

```
>>> from lionshead import *
>>> get_specific_platform()
SpecificPlatform(dist='debian', major_vers='stretch/sid', full_vers='stretch/sid', stability='unstabl
>>> get_specific_platform_string()
'debian-stretch_sid'
>>> get_platform_stability_string()
'unstable'
```

On CentOS 7:

```
>>> from lionshead import *
>>> get_specific_platform()
SpecificPlatform(dist='centos', major_vers='7', full_vers='7', stability='stable')
>>> get_specific_platform_string()
'centos-7'
>>> get_platform_stability_string()
'stable'
```

Installation of the module also installs `lionshead-platform` and `lionshead-stability` commands that return the output of `get_specific_platform_string` and `get_platform_stability_string`, respectively.

1.2 FAQ

1.2.1 How can I help?

See if your OS/distribution is listed and/or correct in this gist, and if not, follow the instructions and create an issue with your findings.

1.2.2 Does this work on anything other than Linux?

Not currently, but as an avid illumos fan, I plan to add support for other operating systems such as illumos and the BSDs. Here's the data collection I did for Ansible's illumos detection to get started.

1.2.3 What is a "stable" vs. "unstable" platform?

"Stable" platforms are operating system releases which commit to a defined, non-changing ABI for the lifetime of the release. This means that the ABI remains consisntent even after OS updates (this typically means that all software is maintained at a specific version). Examples of "stable" releases include:

- Red Hat Enterprise Linux and derivatives
- Ubuntu
- Debian (stable and past stable releases)
- SUSE Linux Enterprise Server
- openSUSE (releases)

Conversely, "unstable" platforms are operating systems that use a "rolling release" model where software versions can change with each OS update. Examples of "unstable" releases include:

- Debian (testing/sid)
- openSUSE (tumbleweed)
- Arch

1.2.4 What's with the name?

The name is a reference to *Indiana Jones and the Last Crusade*, specifically the scene where Indy takes the "leap of faith" on to the invisible bridge in the Temple of the Sun.

> "Only in the leap from the lion's head will he prove his worth."

The leap detects the (presence of the) platform underneath... get it? ;P

1.3 Code of Conduct

Everyone interacting in the lionshead project's codebases, issue trackers, chat rooms, and mailing lists is expected to follow the PyPA Code of Conduct.

CHAPTER 2

Platform detection logic

2.1 Linux

On Linux, platform detection follows the sequence:

1. Parse `/etc/os-release`, if it exists
2. Parse the output of the `lsb_release -a` command, if it exists
3. Parse the file `/etc/lsb-release`, if it exists
4. Attempt to read distro-specific `/etc/*-release` files

`/etc/os-release` is a standardized file format defined by the systemd project. All distributions that support systemd provide it, and even some distributions that do not support systemd still ship the file. This is the preferred method.

The Linux Standard Base (LSB) system is used as a fallback. This system is optional on most distributions, and can be installed in components. For example, sometimes `/etc/lsb-release` is installed but the `lsb_release` command, which is the only official LSB-defined method for reading LSB release values, is not.

If none of the above can be found, other known OS-specific methods will be used, e.g. the presence of `/etc/debian_version`, `/etc/arch-release`, or `/etc/slackware-version`.

CHAPTER 3

History

3.1 0.1 (2017-01-03)

- Yanked `platform` module out of my pip/wheel forks and adapted as a standalone module for initial version.
- Wrote a bit of documentation.

CHAPTER 4

Indices and tables

- genindex
- modindex
- search