Tools for Open Source Systems Administration

POSSCON 2015

Elizabeth K. Joseph
lyz@princessleia.com
@pleia2
Elizabeth K. Joseph

- OpenStack Infrastructure Team sysadmin
- Long time contributor to various open source projects
- Co-author of The Official Ubuntu Book, 8th ed
How most open source projects do infrastructure

- Team (or company) manages it ...or they just use code hosting
- Requests are submitted via mailing list, bug report or ticketing system
- Request priority is determined by the core team

This may be similar to your organization.

Is there a better way?
OpenStack Infrastructure Team

• Our job is to make sure the OpenStack developers can do their job

• All of our system configurations are open source and tracked in git: https://git.openstack.org/cgit/openstack-infra

• Anyone in the world can propose patches for direct inclusion in our infrastructure, instructions at: http://docs.openstack.org/infra/manual/developers.html
What we run

- Askbot
- Continuous Integration systems
- Cacti
- Elasticsearch, Logstash and Kibana
- IRC Bots
- Etherpad
- Git
- Paste
- Planet
- Puppetboard
- Mailing Lists
- Various smaller web services
- Wiki
OpenStack Continuous Integration (CI) System

- Lots of individual projects
- All projects must work together
- Changes can't break master branch
- Code should be syntactically clean
- Testing must be completed automated
Tools we're using for CI

- Launchpad (someday: openstackid)
- Git
- Gerrit
- Zuul*
- Gearman
- Jenkins (with jenkins-job-builder*, devstack-gate*)
- Nodepool*

* Started by the OpenStack Infrastructure team
Workflow

- Github mirror git.openstack.org mirror
- Git repository
- Gerrit code review
- Zuul
- Gearman Server
- Jenkins (gearman-plugin)
- pip mirrors & other semi-privileged servers
- Jenkins01,02...
- Ubuntu, Fedora, Centos & devstack slaves

Local changes submitted via git-review

Elizabeth K. Joseph | @pleia2
Automated tests for infrastructure

- flake8 (pep 8 and pyflakes)
- puppet parser validate
- puppet lint
- Puppet application tests
- XML checkers
- Alphabetized files
- IRC channel permissions
Peer review means...

- Multiple eyes on changes prior to merging
- Good infrastructure for developing new solutions
- No special process to go through for commit access
- Trains us to be collaborative by default
- Since anyone can contribute, anyone can devote resources to it
Automated deployment

• Change gets approved, tested and merged
• ...Either puppet master gets updated and applies change
• ...Or vcsrepo module in puppet pulls in latest version of project
Can you really manage an infrastructure via git commits?

- Cacti ([http://cacti.openstack.org/](http://cacti.openstack.org/)) to keep an eye on server usage
- PuppetBoard ([http://puppetboard.openstack.org/](http://puppetboard.openstack.org/)) so you can watch your changes get applied, or not
- Thorough, specific documentation at [http://ci.openstack.org](http://ci.openstack.org)
Well, not everything

- Automation is imperfect and doesn't cover everything, sometimes you just need to log into a server
- Complicated migrations and upgrades need manual components
- Initial persistent server deployment still has manual components
- Passwords need to be privately managed (but we use git!)
Human collaboration

• IRC channel
• Weekly IRC-based meetings
• Etherpad
• Pastebin
• In person collaboration at the OpenStack summit every 6 months

No voice calls.
Questions

OpenStack CI Resources: http://ci.openstack.org