Continuous Integration with Jenkins

Miro Cupák & Oliver Gondža
IRC: #jenkins

Oct 8 & Oct 10, 2014

What's the plan?

- 3 sessions
  - A: Using Jenkins
    - Oct 8: 10:00-12:00, Oct 10: 10:00-12:00
    - mcupak, ogondza
  - B: Jenkins administration and development
    - Oct 8: 13:00-17:00, Oct 10: 13:00-17:00
    - mcupak, ogondza
    - BYOL (bring your own laptop)
  - C: Central CI
    - Oct 13: 13:00-14:00 + workshops
    - alivigni

Why us?

- not Platform CI/SCALE
- MW QE has long history of using Jenkins
- one of the largest instances in the world
  - ~12k jobs, ~400 slaves, ~2000 builds/day

- active in upstream (core, plugins, LTS)

Using Jenkins

What is Jenkins?

- Jenkins is the leading open source continuous integration server (70% of the market).
- It is primarily written in Java, but also used for Ruby, Python, Objective-C, and much, much more.
- Formerly known as Hudson.

What does it help with?

- notice a change
- check out source code
- execute builds/tests
- record and publish results
- notify developers

GUI overview

- master/slave
- jobs/projects, builds, workspaces, artifacts, reports
- scheduled builds, build queue and executor status
- slaves
  - label (expression)
  - executors
- management
- folders

Jobs

- triggers (SCM, cron)
- build steps
- environment vars ($BUILD_NUMBER, $JOB_NAME, $JOB_URL...)
- post build steps
  - notifiers (mail, email-ext, irc)
  - build description
  - archivers (artifacts, test results, specific reports, html publisher)
  - external publishers (deploying)

Jobs

- parallel builds
- discarding old builds
- build parameters
- quiet period
- restrict where this project can be run
- upstream-downstream
- results and weather report

Matrix Jobs

- parametrized job running all combinations of parameters
- axes and configurations
- combination filter
- touchstone builds
- configurations executed in parallel

REST API

- 3 flavours
  - XML
  - JSON with JSONP support
  - Python/Ruby
- do things like retrieve information from Jenkins for programmatic consumption, trigger a new build, create/copy jobs
- examples

CLI

- download CLI jar from http://yourserver.com/cli
- syntax
  - java -jar jenkins-cli.jar [-s JENKINS_URL] command [options] [argument]
- examples

Security

- authentication realms
  - built-in DB
  - LDAP
  - PAM etc.
- authorization
  - anyone can do anything
  - logged-in users can do everything
  - matrix-based security
  - project-based security

Plugins

- SCM
  - Git (Git Client), Subversion, Multiple SCMs
- JobConfigHistory
- CloudBees Folders
- Dashboard View
- Build-timeout
- Timestamper
- Matrix Reloaded
- NodeLabel Parameter

Plugins

- Workspace Cleanup
- Monitoring
- Tmp Cleaner
- Audit Trail
- EnvInject
- Xvnc
- Ownership
- Email-ext
- Plot Plugin

Plugins

- Build Flow
- Beaker Builder
- JClouds
- Amazon EC2
- UI Samples
- performance tests, code coverage and much more

- ChuckNorris ;-)

Best Practices

- do not overuse shell steps
- use plugins productively
- use incremental builds
- make builds self-checking
- take steps to ensure failures are reported as soon as possible
- write jobs for your maintenance tasks
- tag, label, or baseline the codebase after the successful build

Best Practices

- keep source in SCM, not in job configration
- avoid manual steps - automate build triggering and result notifications
- restrict jobs to labels, not particular slave names
- stick with stable releases
- join the community

Resources

- https://jenkins.mw.lab.eng.bos.redhat.com/jenkins-book/
- https://wiki.jenkins-ci.org/display/JENKINS/Use+Jenkins
- https://wiki.jenkins-ci.org/display/JENKINS/Plugins

Questions?