DevOps Working Group

Thursday April 16, 2020
# Agenda

<table>
<thead>
<tr>
<th>Time</th>
<th>Topic</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Min</td>
<td>Geneva / DevOps Updates</td>
<td>James</td>
</tr>
<tr>
<td>10 Min</td>
<td>Geneva Release Planning</td>
<td>Lisa</td>
</tr>
<tr>
<td>15 Min</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Min</td>
<td>AOB / Opens</td>
<td>All</td>
</tr>
</tbody>
</table>
Attendees

Participants (5/6)

JG  James Gregg (Intel) (Me)
EO  Ernesto Ojeda (Intel)
TE  tony espy
ER  Emilio Reyes (Intel)
JW  Jim White
LR  Lisa Rashidi-Ranjbar
DevOps WG Update

Geneva
• Jenkins Transformation to Pipelines
  • Work continues on the transformation to Jenkins Pipelines
    • Multiple stories related to release automation for Geneva - WIP
      • Add edgex-global-pipeline version information in the pipeline log - DONE
      • Implement Snap stages for x86 and ARM (Issue #155) - DONE
    • Integrate edgeXSnap function into edgeXBuildGoApp and edgeXBuildCApp (Issue #125) - WIP
      • Identified a bug in the way we’re linting the groovy code for edgex global pipelines - WIP
    • Reviewed LF roadmap 2020-beyond
      • self service committers
      • self service repo creation
      • K8s support for CI Validation Only
      • support for CI/CD for projects that want their own SaaS based build automation (Azure DevOps)
      • continued support of solutions already implemented (Clair, TIG stack, SonarCloud, Snyk integrations)
    • Still have multiple stories to get the snap release functionality into the release automation
    • Resolved issue with missing GPG key on Jenkins service account - Thank you Eric Ball / Linux Foundation
    • LFTools / Sigul latest version that supports Python 3.x - RISK
      • Need input from LF on alternative signing tool / No update ww16

Hanoi
• Pre-Planning and some backlog grooming with additional scope related to release automation
  • Backlog grooming discussed in 04/02/20 DevOps WG meeting with decision to drop some of the backlog work items since
  • Moved Bluetooth device service into Hanoi per TSC discussion on 04/15/2020
Geneva Freeze and Release dates

TSC approved

• Freeze: 12pm GMT, April 22 (Wed, week before planning meeting)
• Release: 12pm GMT, May 13 (Wed two weeks after planning meeting)

See Geneva release notes for details (on Slack)

REMINDER:
We will NOT be branching off master for the Geneva release.
Includes EVERYTHING
Release Artifact Status

• Docker images are staging to Nexus we are good to go here.
• We have some broken snap jobs for the following repos:
  • device-grove-c << Geneva Snap??  
    *Tony says to assign to him and can be done post-release*
  • device-mqttt-go - WIP
  • device-modbus-go – WIP – *Missing version file but Ernesto is addressing*
  • edgex-ui-go << Geneva Snap??  
    Per Tony – was never released to stable channel (beta hasn’t been updated)  
    Suggestion to include in main edgex
Geneva release

<table>
<thead>
<tr>
<th>Completed</th>
<th>Work In Progress</th>
<th>Release Backlog</th>
</tr>
</thead>
<tbody>
<tr>
<td>178</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>
Geneva Release Schedule
Geneva Dependency

- **JSON Logic and/or EMGX Kupfer Demos (App WG)**
- **Deprecate**
  - Archive Support Rules Engine (App WG)

- **V2 API Specification (Core WG)**

- **Blackbox TAF Framework (Test/QA WG)**

- **API Documentation Template - Swagger (Test/QA WG)**

- **Jenkins Pipelines Transformation (DevOps WG)**

- **Open Horizon "Walk" Phase (System WG)**

- **Open Horizon Build Automation (DevOps WG)**

- **API Minor Versioning (Certification WG)**

- **System Integration Tests (Test/QA WG)**

- **Data Feed Back into Core Data (App WG)**

- **Alternative Message Bus Provider (Core + App WG)**

- **Data Filter Design DS and Core Data (Device WG)**

- **App Services Batch and Send (App WG)**

- **Blacklist/Whitelist of devices (Core + Device WG)**

- **Dynamic Device Provisioning (Device WG)**

- **Archive Export Services (App WG)**

- **Update to Golang 1.13**

- **Separate Config and Registry APIs**

- **Redis Default DB**

- **Hardware Secret Storage (Security WG)**

- **Per Service Vault Token (Security WG)**

- **Vault Token Rotation (Security WG)**
Backlog Review

<table>
<thead>
<tr>
<th>Issue</th>
<th>Priority</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1234</td>
<td>High</td>
<td>Implement new feature X.</td>
</tr>
<tr>
<td>#5678</td>
<td>Medium</td>
<td>Fix bugs in version 1.0.</td>
</tr>
<tr>
<td>#9012</td>
<td>Low</td>
<td>Optimize performance.</td>
</tr>
</tbody>
</table>

**QA/Code Review**

- Test case 1: Passed
- Test case 2: Failed
- Test case 3: In progress

**Tasks**

- Task 1: Complete by end of week
- Task 2: Under review
- Task 3: Ready for deployment

**Notes**

- Note 1: Consider using a different approach.
- Note 2: Review data from the last quarter.

**Important Dates**

- March 30: Submission deadline
- April 5: Review meeting
- April 10: Public release
Meeting Minutes

• Some issues with snap build failures may be network related as the failures are occurring when pulling dependencies from apt repos
  • Ernesto / Tony to work offline on the snap issues if we continue to see the issues

• Tony would like to own the work for the device-grove-c snap development
  • Assign Issue to Tony Espy

• DevOps is making good progress on the Pipelines to support snap releases
  • Risk reduced given the focus in this last sprint – Good job Ernesto, Emilio, Lisa, Bill!! Thank you Tony Espy for helping out and being so responsive to the team.
DevOps WG Update (Geneva)

Geneva (~Apr 2020) Focus:

- DevOps Jenkins Pipeline Transformation completed
  - Introduced new Jenkins Global Libraries for build automation
    - Includes test framework for Groovy code
    - Explore underway to look into code coverage of Groovy code using Codecov.io
  - Semantic Versioning using Intel contributed utility (git-semver) enhanced to include test framework
  - Continuous Delivery via "release-kraken"
  - Developer Enablement – GitHub Project Tracker, GitHub Issue label creation automated, gitcommit linter implemented *
  - New ci-build images and global libraries developed to support Jenkins Pipelines
  - New life cycle policies implemented on Linux Foundation Nexus repositories
  - Developer Documentation created for new Jenkins Pipelines
  - Improved performance of all builds to include collaboration with Linux Foundation to drive performance improvements for ARM builds (~15 mins build performance improvements using a new flavor of LF build nodes)
    - X86 build nodes (VM) uses 4cpu – 2gb
    - Arm64 build nodes (VM) now uses 4 cpu – 16gb

DevSecOps scope includes:

- Snyk Advanced Reporting via Community Bridge - $8K savings on licensing for developer licenses
- Snyk Docker Hub image scans with weekly reports of new vulnerabilities
- Snyk CLI of Go integrated into scan stage of Jenkins Pipelines
- Clair image scans within scan stage of Jenkins Pipelines
- DevOps contributed code fixes to address CVEs found in images based on Snyk reporting
- Lftools updated to use latest version – code signing, git tag signing, Docker image signing
Hanoi Planning

Scope Discussions
Hanoi - DevOps

- Performance Optimizations
  - Jenkins Pipeline optimizations for edgex-go
  - Explore options from LF for supporting Jenkins on K8s
- Performance of the Build Environment
  - Monitoring / Alerting optimizations (Continuous Improvement Opportunity)
- Technical Debt
  - Caching Dependencies – speed it up (upstream dependencies)
- Open Horizons Enablement
  - Shared Infra with Open Horizons
  - Build Automation for OH
- Stretch Goals
  - Code Coverage for Jenkins Global Libraries (codecov.io)
  - Snap improvements
  - Support for –race flag
Geneva Planning

Scope Discussions
Fuji Release

- Freeze: Oct 23\textsuperscript{rd} (Wednesday)
- Release: Nov 15\textsuperscript{th} (Friday)

<table>
<thead>
<tr>
<th>Assigned to:</th>
<th>week 1</th>
<th>week 2</th>
<th>week 3</th>
<th>week 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Date:</td>
<td>10/23/19 (with extension)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Developers</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WG Chairs</td>
<td></td>
<td>Cut Fuji Branches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WG Chairs</td>
<td></td>
<td>GitHub Issues: Close / Mark for Geneva</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DevOps</td>
<td>Create Fuji Jobs For Existing Repos</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DevOps</td>
<td></td>
<td>Clair Scan of EdgeX Images</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Release Tsar</td>
<td></td>
<td></td>
<td>Open Tickets with LF for release on 11/15/19</td>
<td></td>
</tr>
<tr>
<td>Release Tsar</td>
<td></td>
<td></td>
<td>Finalize Release Notes</td>
<td></td>
</tr>
</tbody>
</table>
Geneva – DevOps

In

• Full Pipeline transformation for EdgeX services
  • Convert Jenkins JJB Freestyle jobs to Jenkins Pipelines
• Introduce GitHub Org Plugin
• Simplified Jenkinsfile
• Global Libraries to support Jenkins Pipeline transformation
• Add Unit testing to global-libraries (uncommitted) **
• Snyk integration for edgex services
  • As part of Jenkins Pipeline conversion
• Slack integration with Jenkins pipelines
• Nexus Cleanup / Lifecycle Policy

Out

• Alternate deployment/orchestration
  • Beyond Docker/Snaps
  • Kubernetes
  • Kata Containers
  • …
• Integration Test Pipelines
• Code signing / Artifact signing **
Geneva Transformation: Architecture
How long does it take? Is this all Geneva scope?

Geneva Transformation

Phase 1
- Research Spikes
- Plugin Setup and Configuration
  - Jenkinsfile
  - Jenkinsfile.sandbox

Phase 2
- Jenkinsfile templates
- Implementation details get solidified
- Refactor existing pipelines to use new templates

Phase 3
- Existing Job Migration

Full Transformation by Geneva Release - April 2020
Fuji Planning

Scope Discussions
Fuji – DevOps

In
- Static code analysis tool identified and integrated into the EdgeX Jenkins Pipeline for Docker image scanning (Clair Server)
- Explore SAST for true static code analysis to include additional tooling such as Fortify / Coverity
- Code and artifact signing with semantic versioning
- Fix Documentation – edgex-go
  - Create a new repo for edgex-docs
- Build Performance Optimizations
  - Pipelines for EdgeX Foundry base build images
  - Basebuild images managed locally within Nexus
  - Leverage PyPi Proxy for local pip dependencies
  - ARM builds – optimization leveraging different high CPU build nodes / OS (ARM Team)

Out
- Alternate deployment/orchestration
  - Beyond Docker/Snaps
  - Kubernetes
  - Kata Containers
  - …
- SonarQube – SonarCloud is already in play in the LF Decision: wait to see what codecov.io offers
- Suggestion to rename all of the Jenkins “arm” jobs so as to differentiate 32bit / 64bit architectures
- Full Pipeline transformation for EdgeX services
# EdgeX DevOps Commitments (Fuji)

<table>
<thead>
<tr>
<th>Scope of Work</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add static artifact analysis into the EdgeX Jenkins Pipeline</td>
<td>🟢</td>
</tr>
<tr>
<td>(analysis of Docker /runtime artifacts, not the source code)</td>
<td></td>
</tr>
<tr>
<td>Add code and artifact signing with semantic versioning</td>
<td>🟢</td>
</tr>
<tr>
<td>Conduct build performance optimizations by:</td>
<td>🟢</td>
</tr>
<tr>
<td>• Adding Pipelines for EdgeX Foundry base build images</td>
<td></td>
</tr>
<tr>
<td>• Allow base build images to be managed locally within Nexus</td>
<td></td>
</tr>
<tr>
<td>• Leverage PyPi Proxy for local pip dependencies</td>
<td></td>
</tr>
<tr>
<td>Explore static code analysis like Checkmarx, Coverity, GuardRails, Synk,</td>
<td>🟢</td>
</tr>
<tr>
<td>SonarQube</td>
<td></td>
</tr>
</tbody>
</table>

- Clair Server landing no longer at risk for Fuji
  - LF committed to implement on AWS and fund with expected completion next week
- gitsemver along with lftools used for artifact signing and semantic versioning
- Jenkins build performance optimizations for base build images completed
- All base build images will now be stored in Nexus (Snapshot):10003
- PyPi enabled as part of Edinburgh scope
- Initial review of GuardRails showed that the product was identifying issues which were not applicable for microservices architecture
### Past / Future Agenda Topics

<table>
<thead>
<tr>
<th>Topic</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size change to use Ubuntu / Debian base build images to support –race flag for Go Lang</td>
<td></td>
</tr>
<tr>
<td>Clair scan findings – Discussion developer community if we want to break the build when there’s findings</td>
<td></td>
</tr>
<tr>
<td>- Bring into Security WG for discussion</td>
<td></td>
</tr>
<tr>
<td>Open Horizons enablement</td>
<td></td>
</tr>
<tr>
<td>Alignment to new LF roadmap self-service offerings – EdgeX use case for handling holding repositories</td>
<td></td>
</tr>
<tr>
<td>Release automation - key learnings and sharing with LF</td>
<td></td>
</tr>
</tbody>
</table>
Attendees & Community Participation – ww14
Attendees & Community Participation – ww15

Community Participation

- Intel
- IoTech
- Dell
- VMWare
- ARM
- Canonical
- LF
- Kong