DevOps Working Group

Thursday November 19, 2020
## Agenda

<table>
<thead>
<tr>
<th>Time</th>
<th>Topic</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-10 Min</td>
<td>Hanoi Retrospective</td>
<td>Ernesto</td>
</tr>
<tr>
<td>10 Min</td>
<td>Official Docker Images</td>
<td>Venkat</td>
</tr>
<tr>
<td>20 Min</td>
<td>Ireland Scope</td>
<td>All</td>
</tr>
<tr>
<td>5 Min</td>
<td>AOB / Opens</td>
<td>All</td>
</tr>
</tbody>
</table>
Attendees

- Ernesto Ojeda (Host, me)
- Jim White (Co-host)
- Lenny Goodell (Intel)
- Emilio Reyes (Intel)
- tonyespy
- Bill Mahoney (intel)
- Chinu Joy (Intel)
- Eric Ball
- Ricardo
- Siggi Skulason (Canonical)
- Venkat (Intel)
- Walt (Intel)
Hanoi DevOps Highlights

**Overall Speed Improvements for edgex-go**

- **Docker Build Times:**
  - Before: 20m (Serial)
  - After: 4m (Parallel)

- **Snap builds:**
  - Before: 45m
  - After: 15m
Hanoi DevOps Deliverables

Build Optimizations
- New faster build pipeline for edgex-go (build images in parallel)
- New edgex-cli build pipeline, create and publish edgex-cli binaries
- Snap build speed improvements for amd64 and arm64
- EdgeX Release optimization (rebuild artifacts)
- Facilitate the upgrade to Go 1.15
- Implement workaround for race condition testing for Go prior to v1.15

Developer Enablement
- DevOps World presentation of edgex-global-pipelines
- Auto generation of documentation for edgex-global-pipelines
- EdgeX AWS ECS reference stack deployment

New Pipeline Features
- Introduce concept of "build commits" to trigger rebuild of artifacts
- GitHub Release automation
- GitHub prune stale/pre-release git tags
- GitHub label/milestone sync (with throttling)
- SwaggerHub API global function

Technical Debt/ DevOps Continuous Improvements
- Linting of groovy code in pipeline to avoid syntax failures
- Implement mocking of transitive functions in edgex-global-pipelines
- Cleanup and standardization of Mocking in unit tests
- Standardization use of env. across all pipelines
- Complete re-write of sample-service to enable full end-to-end testing of pipeline features
- git-semver bug fixes and more robust unit tests
- Generate report for stale docker images
- Standardization of ENTRYPOINT across all Dockerfiles in all repos

DevSecOps
- Snyk scanning pre-release docker images
- Introduce process for vetting of OSS dependencies

Releases
- Released 8 device services
- Geneva Dot release
Hanoi Retrospective

What went right?
• Performance increases were outstanding
• Release process went very smooth.
• DevOps and Dev teams more inline
• Kudos to Emilio for GitHub Releases
• Move to pipelines helped visualize errors more efficiently
• Kudos to more stable LF infra (network)

What could have been better?
• Worry about DevOps pipeline complexity (documentation helps).
• Intermittent network failures (docker pulls)
• Couple of repos had master failures would be nice to have an overview before release.
• Lack of integration tests with device services
• Inconsistencies with repo pipeline capabilities
• Committer rights audit may be needed
• Code freeze 3 weeks? Leave at 3 weeks through the Ireland release.
• Implement formalized post-release process
• Update semver version to 2.0.0-dev.x
Official Docker Images

Venkat Presentation
Notes