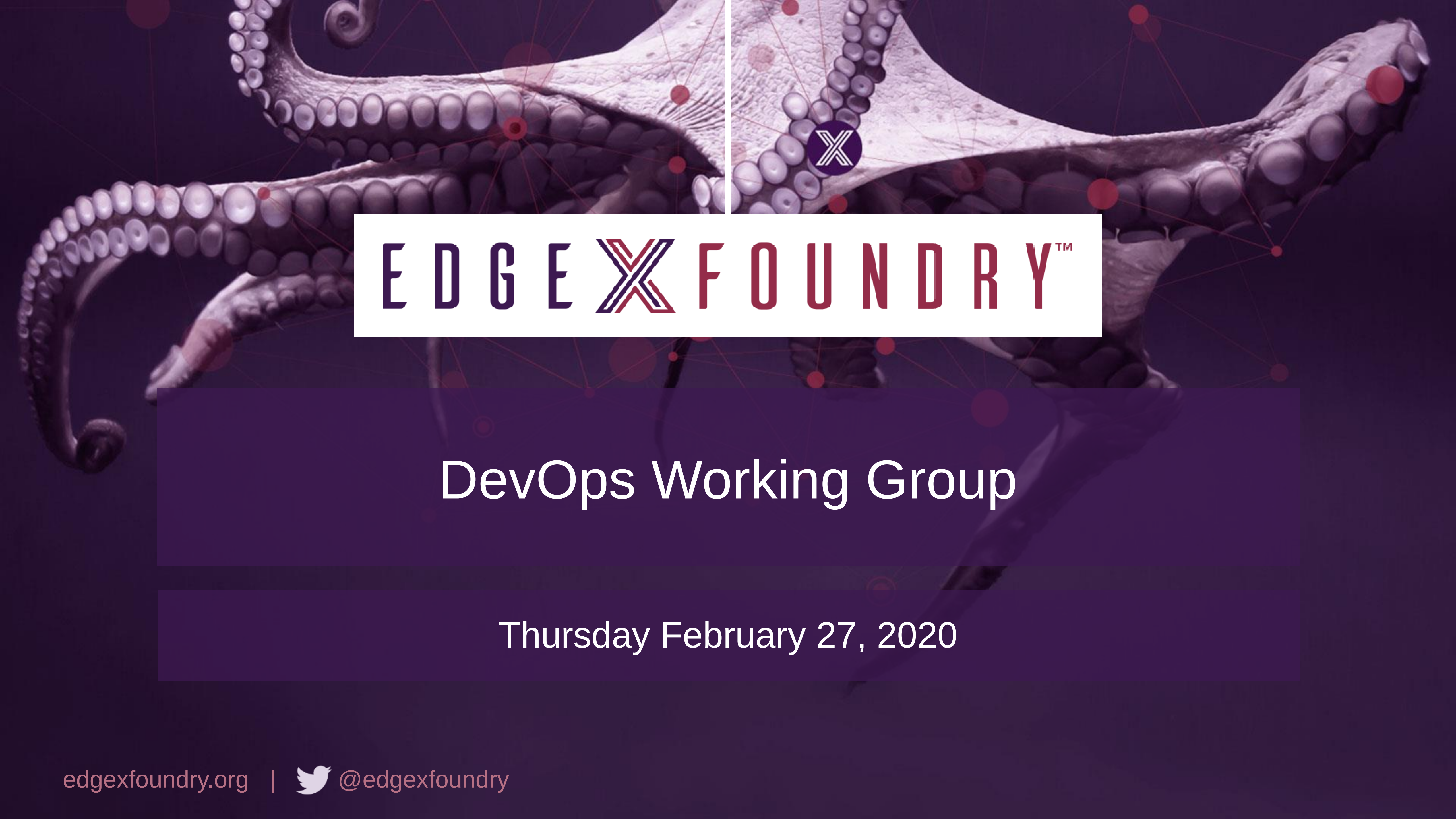


EDGEXFOUNDRY™

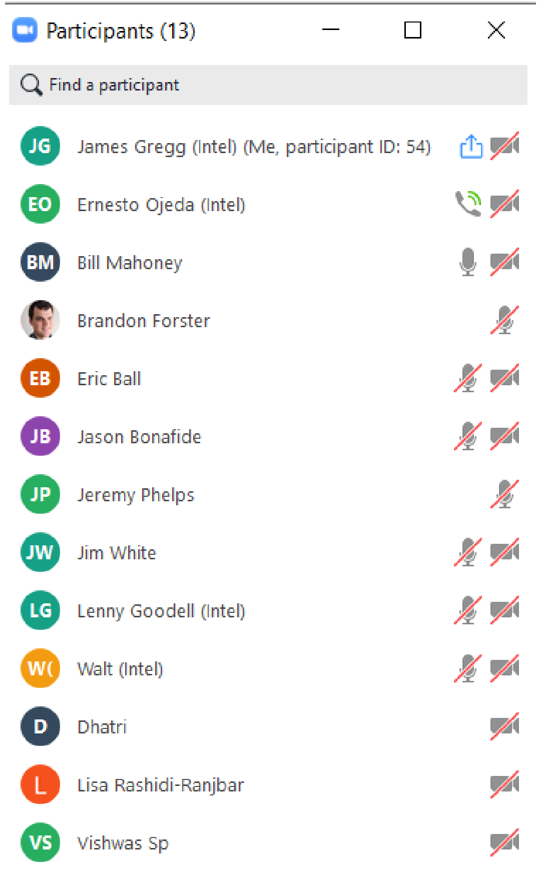
DevOps Working Group

Thursday February 27, 2020

Agenda

Time	Topic	Owner
10 Min	Geneva / DevOps Updates	James
30 Min	“Release the Kraken” - Review Proposal for Automating the Edgex Releases	Lisa
5 Min	ARM Build Speed Improvements Changes	Eric Ball
10 Min	Backlog Review (Time Permitting)	James
5 Min	AOB / Opens - Snap Global Library - race detection library – alpine issue	All

Attendees



Participants (13)

Find a participant

- JG James Gregg (Intel) (Me, participant ID: 54)
- EO Ernesto Ojeda (Intel)
- BM Bill Mahoney
- Brandon Forster
- EB Eric Ball
- JB Jason Bonafide
- JP Jeremy Phelps
- JW Jim White
- LG Lenny Goodell (Intel)
- W(Walt (Intel)
- D Dhatri
- L Lisa Rashidi-Ranjbar
- VS Vishwas Sp

DevOps WG Update

Geneva

- Jenkins Transformation to Pipelines
  - Work continues on the transformation to Jenkins Pipelines
    - Lisa is exploring work to look into full automation of the release – WIP
      - Demo / Discussion
      - Request to move cd-management out of holding to main Org EdgeX Foundry
    - Automation for the GitHub Issue labels - WIP
    - EdgeX-Go Pipeline now in place
      - Need to update documentation and work on the issue with managing multiple Jenkinsfiles
    - PR “recheck” now resolved with new Jenkins Plugin
    - git-semver unit testing - decided to fix existing code vs. rewrite - WIP

- **CommunityBridge - Advanced Snyk Reporting**
- EdgeX Foundry added to the CommunityBridge Vulnerability Reporting
    - Ticket still open with Heather Willson & David Deal / CommunityBridge team
    - We now have Advanced Snyk Reports but working through multiple new issues

DevOps WG Update

Pipeline recheck functionality

- Enabled by pipeline-github-plugin
    - https://github.com/jenkinsci/pipeline-github-plugin
    - Add new trigger to pipeline script

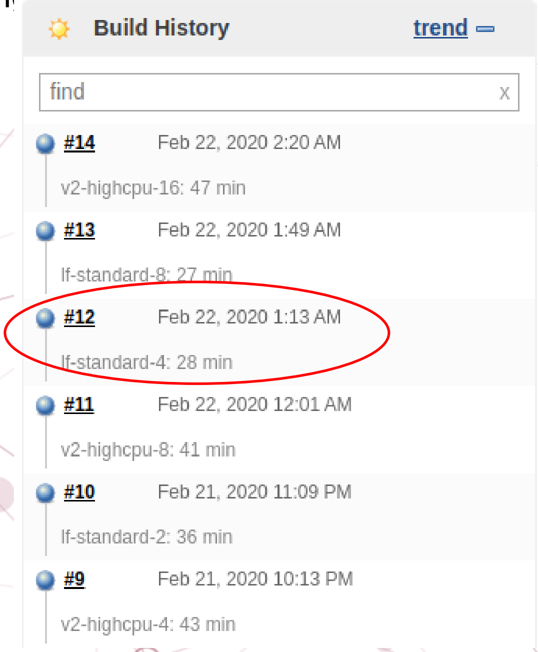
```
pipeline {
   triggers {
      issueCommentTrigger('^recheck$')
   }
}
```

ARM Build Speed Improvements

- Slow ARM builds
  #584 Change ubuntu18.04-docker-arm64 to lf-standard-4

Eric Ball updated a comment:

- Let me begin by pointing out that the current builder is mislabeled. Though it is named "4c-2g", the hardware flavor lf-standard-2 is in fact 2c-4g. However, not all CPUs are created equal. After running a number of tests, I found that the most balanced result was using the **lf-standard-4 flavor, which is 4c-16g**. Note that both lf-standard flavors were faster than our highcpu flavors, which I tested with 4c-4g, 8c-8g, and 16c-16g configurations



Build History trend

find x

#14 Feb 22, 2020 2:20 AM
v2-highcpu-16: 47 min

#13 Feb 22, 2020 1:49 AM
lf-standard-8: 27 min

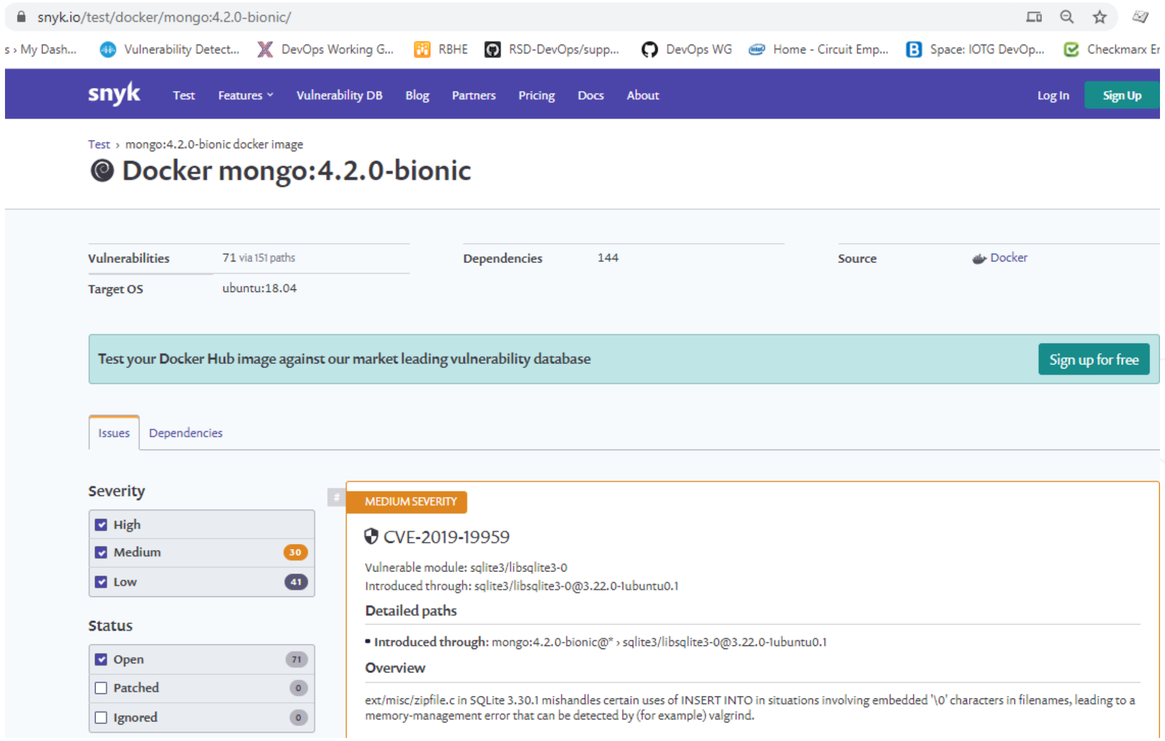
#12 Feb 22, 2020 1:13 AM
lf-standard-4: 28 min

#11 Feb 22, 2020 12:01 AM
v2-highcpu-8: 41 min

#10 Feb 21, 2020 11:09 PM
lf-standard-2: 36 min

#9 Feb 21, 2020 10:13 PM
v2-highcpu-4: 43 min

Snyk Scan Results - mongo



Test › mongo:4.2.0-bionic docker image

Docker mongo:4.2.0-bionic

Vulnerabilities 71 via 151 paths

Target OS ubuntu:18.04

Dependencies 144

Source Docker

Test your Docker Hub image against our market leading vulnerability database **Sign up for free**

Issues | Dependencies

Severity

- ☑ High
- ☑ Medium 30
- ☑ Low 41

Status

- ☑ Open 71
- ☐ Patched 0
- ☐ Ignored 0

MEDIUM SEVERITY

CVE-2019-19959

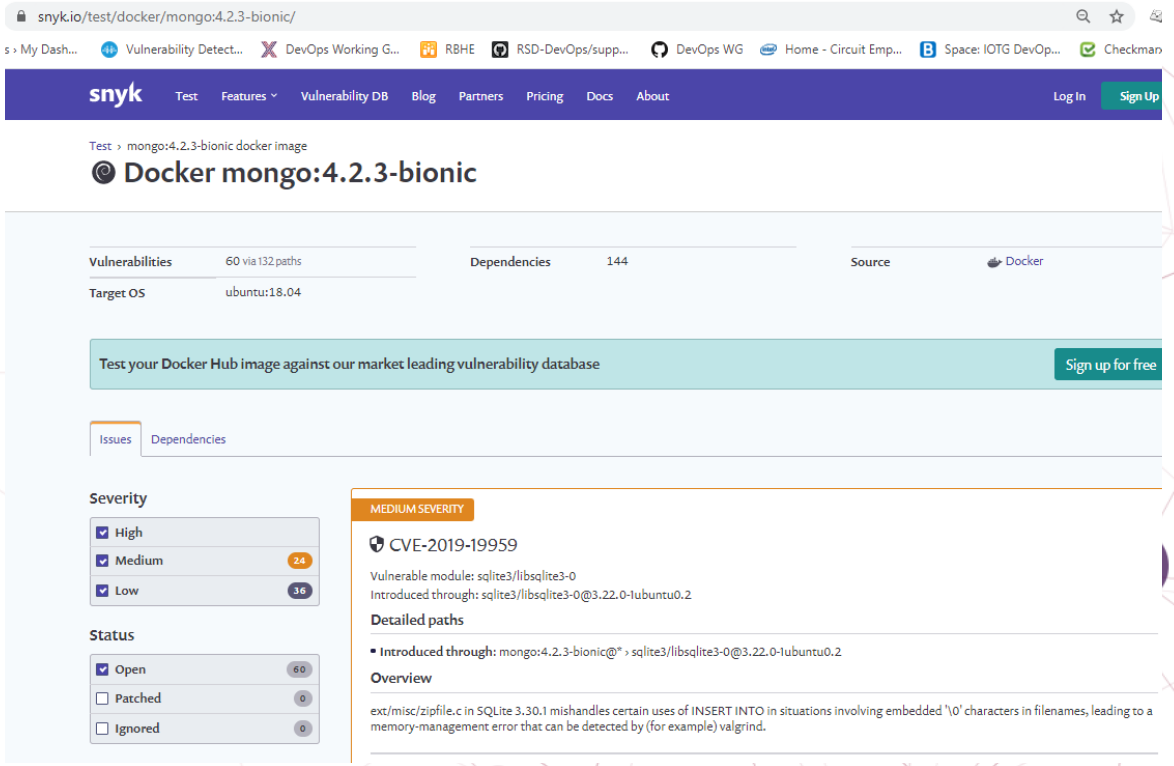
Vulnerable module: sqlite3/libsqlite3-0
Introduced through: sqlite3/libsqlite3-0@3.22.0-1ubuntu0.1

Detailed paths

- **Introduced through:** mongo:4.2.0-bionic@* › sqlite3/libsqlite3-0@3.22.0-1ubuntu0.1

Overview

ext/misc/zipfile.c in SQLite 3.30.1 mishandles certain uses of INSERT INTO in situations involving embedded '\0' characters in filenames, leading to a memory-management error that can be detected by (for example) valgrind.



Test › mongo:4.2.3-bionic docker image

Docker mongo:4.2.3-bionic

Vulnerabilities 60 via 132 paths

Target OS ubuntu:18.04

Dependencies 144

Source Docker

Test your Docker Hub image against our market leading vulnerability database **Sign up for free**

Issues | Dependencies

Severity

- ☑ High
- ☑ Medium 24
- ☑ Low 36

Status

- ☑ Open 60
- ☐ Patched 0
- ☐ Ignored 0

MEDIUM SEVERITY

CVE-2019-19959

Vulnerable module: sqlite3/libsqlite3-0
Introduced through: sqlite3/libsqlite3-0@3.22.0-1ubuntu0.2

Detailed paths

- **Introduced through:** mongo:4.2.3-bionic@* › sqlite3/libsqlite3-0@3.22.0-1ubuntu0.2

Overview

ext/misc/zipfile.c in SQLite 3.30.1 mishandles certain uses of INSERT INTO in situations involving embedded '\0' characters in filenames, leading to a memory-management error that can be detected by (for example) valgrind.

Docker mongo:4.2.3-bionic : latest image reported 47 vuln’s.

Release the Kraken - Lisa

cd-management (release-kraken)

- Repository to manage the delivery of EdgeX Foundry release artifacts
- Automate the last steps of the release

Requirements

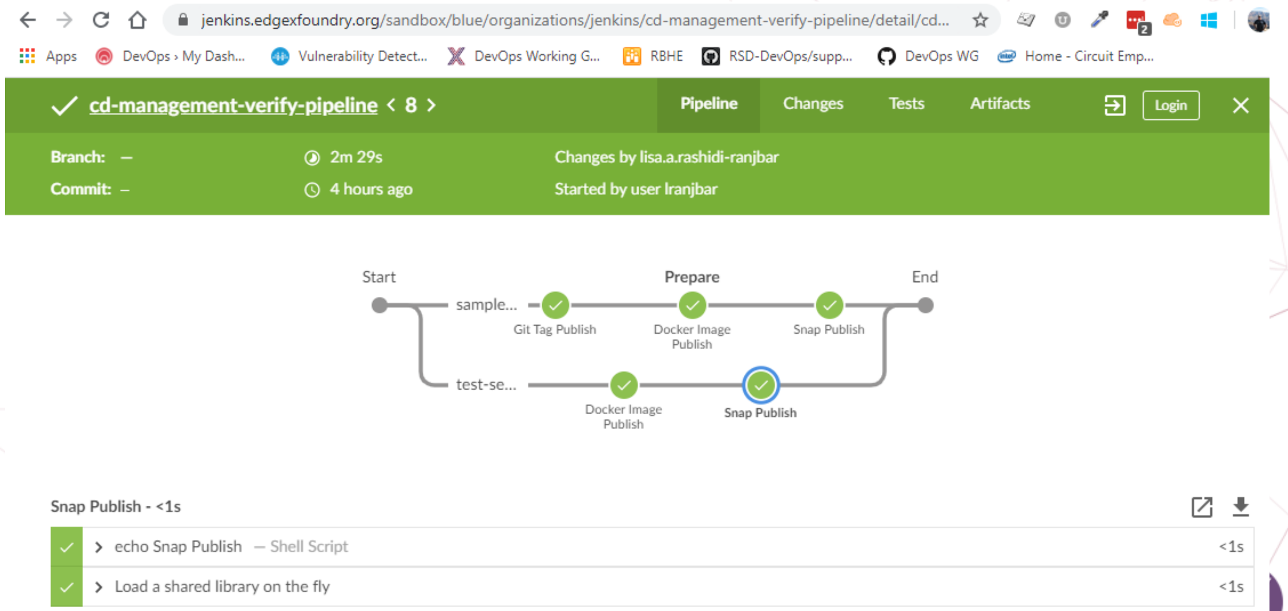
- EdgeX Release artifacts are currently defined as:
  - Git Tags
  - Docker images
  - Snaps
  - Edgex-docs and SwaggerHub were not in the scope for this explore**
- Release artifacts from one or more git repositories at a time
- Repositories can have more than one artifact type to release
- Repositories can have more than one artifact of a specific type to release (ie: multiple docker images)
- Repositories may be released at different times
- Would like a history of releases… audit trail

Release-kraken (continued)

Functional Flow

- YAML files inside cd-management will describe the release
  - YAML files are 1:1 to current git repositories
- When a change is detected in the YAML file we trigger the release for that repository

Demo

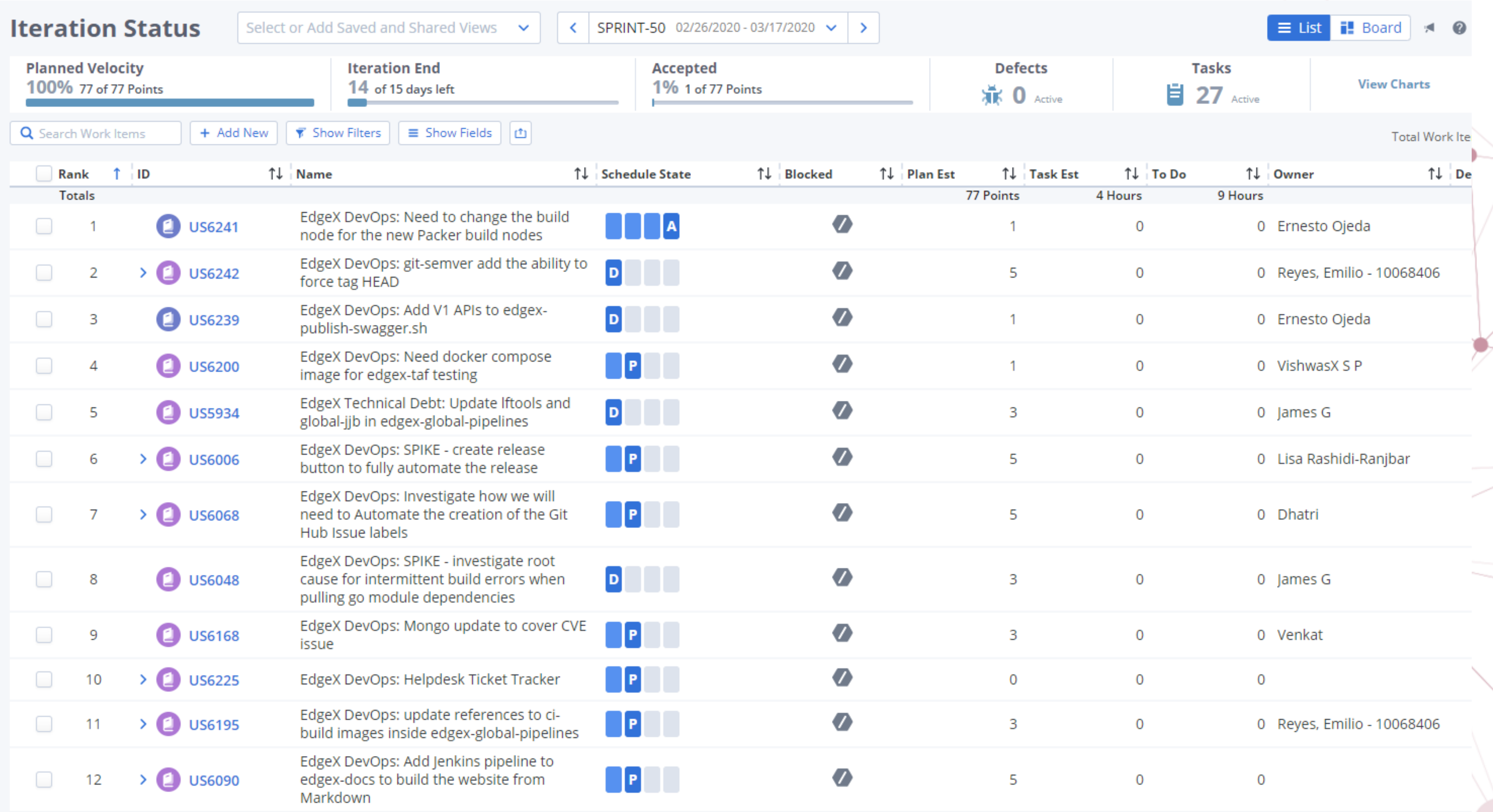


Branch: –
Commit: –
2m 29s
4 hours ago
Changes by lisa.a.rashidi-ranjbar
Started by user lranjbar

Snap Publish - <1s

```
✓ > echo Snap Publish — Shell Script        <1s
✓ > Load a shared library on the fly        <1s
```

Backlog Review



Planned Velocity 100% 77 of 77 Points

Iteration End 14 of 15 days left

Accepted 1% 1 of 77 Points

Defects 0 Active

Tasks 27 Active

Rank	ID	Name	Schedule State	Blocked	Plan Est	Task Est	To Do	Owner	De
Totals					77 Points	4 Hours	9 Hours		
1	US6241	EdgeX DevOps: Need to change the build node for the new Packer build nodes	A	⊘	1	0	0	Ernesto Ojeda	
2	US6242	EdgeX DevOps: git-semver add the ability to force tag HEAD	D	⊘	5	0	0	Reyes, Emilio - 10068406	
3	US6239	EdgeX DevOps: Add V1 APIs to edgex-publish-swagger.sh	D	⊘	1	0	0	Ernesto Ojeda	
4	US6200	EdgeX DevOps: Need docker compose image for edgex-taf testing	P	⊘	1	0	0	VishwasX S P	
5	US5934	EdgeX Technical Debt: Update lftools and global-jjb in edgex-global-pipelines	D	⊘	3	0	0	James G	
6	US6006	EdgeX DevOps: SPIKE - create release button to fully automate the release	P	⊘	5	0	0	Lisa Rashidi-Ranjbar	
7	US6068	EdgeX DevOps: Investigate how we will need to Automate the creation of the Git Hub Issue labels	P	⊘	5	0	0	Dhatri	
8	US6048	EdgeX DevOps: SPIKE - investigate root cause for intermittent build errors when pulling go module dependencies	D	⊘	3	0	0	James G	
9	US6168	EdgeX DevOps: Mongo update to cover CVE issue	P	⊘	3	0	0	Venkat	
10	US6225	EdgeX DevOps: Helpdesk Ticket Tracker	P	⊘	0	0	0		
11	US6195	EdgeX DevOps: update references to ci-build images inside edgex-global-pipelines	P	⊘	3	0	0	Reyes, Emilio - 10068406	
12	US6090	EdgeX DevOps: Add Jenkins pipeline to edgex-docs to build the website from Markdown	P	⊘	5	0	0		

Meeting Minutes

Sprint Planning for next 3 week sprint - Geneva scope was completed yesterday.

Working session set up for next week regarding Snap Global Library for Geneva scope which is remaining work not yet started.

Advanced Snyk Reporting will be for SIR Team members

- Need LF IDs for SIR Team members so that they can be added to the contributor's login for the reports

Request for moving cd-management out of holding will be submitted today for TSC vote via email

race: not working with Alpine based image Open: https://github.com/golang/go/issues/14481
Decision made to push the suggested workaround to HANOI scope for now. This was previously discussed in a previous working group meeting, and it's known known issue with Alpine based images. We don't want to pull in this scope right now as it will force us to use a larger build image (non-Alpine based) and we want to try to see what performance improvements are afforded with the change LF helped us with for the ARM builds. The issue could be fixed by the Go development community at some point, which would introduce some rework to change back to using Alpine for multi-stage builds.

Snyk report for consul image to address jq High Severity CVE is documented
https://snyk.io/vuln/SNYK-ALPINE39-JQ-338867?utm_source=slack

EDGEXFOUNDRY™

Hanoi Planning

Scope Discussions

Hanoi - DevOps

- Performance Optimizations for EdgeX-Go Jenkins Pipelines

EDGE X FOUNDRY™

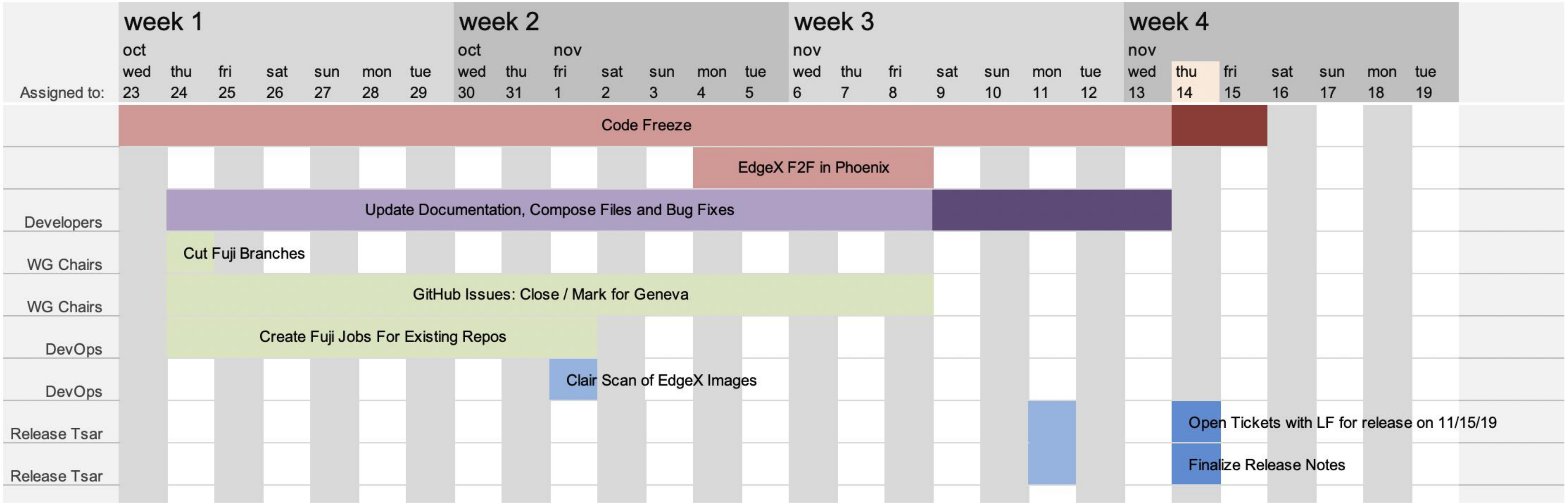
Geneva Planning

Scope Discussions

Fuji Release

- Freeze: Oct 23rd (Wednesday)
- Release: Nov 15th (Friday)

Start Date: 10/23/19 (with extension)



Assigned to:	week 1 oct							week 2 oct		nov					week 3 nov							week 4 nov						
	wed 23	thu 24	fri 25	sat 26	sun 27	mon 28	tue 29	wed 30	thu 31	fri 1	sat 2	sun 3	mon 4	tue 5	wed 6	thu 7	fri 8	sat 9	sun 10	mon 11	tue 12	wed 13	thu 14	fri 15	sat 16	sun 17	mon 18	tue 19
	Code Freeze																											
													EdgeX F2F in Phoenix															
Developers		Update Documentation, Compose Files and Bug Fixes																										
WG Chairs		Cut Fuji Branches																										
WG Chairs		GitHub Issues: Close / Mark for Geneva																										
DevOps		Create Fuji Jobs For Existing Repos																										
DevOps										Clair Scan of EdgeX Images																		
Release Tsar																							Open Tickets with LF for release on 11/15/19					
Release Tsar																							Finalize Release Notes					

Geneva – DevOps

In

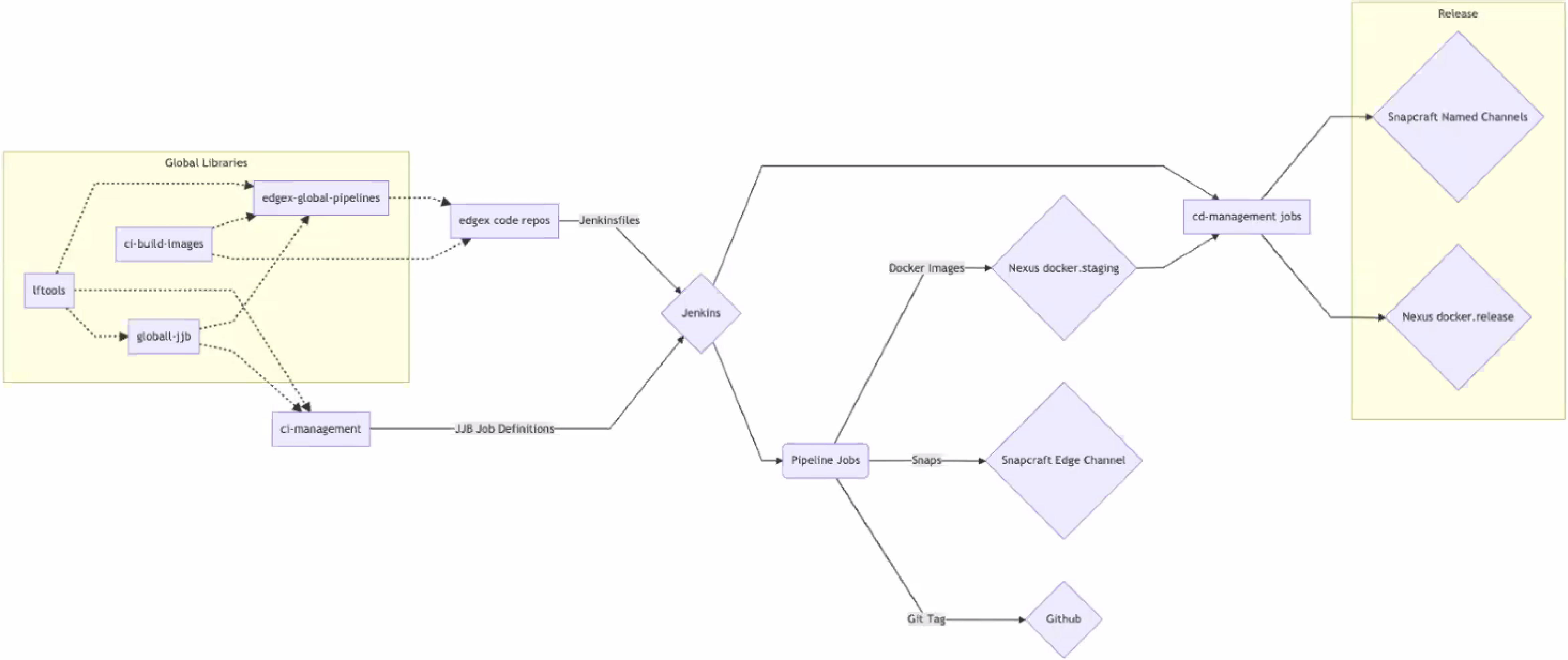
- Full Pipeline transformation for EdgeX services
  - Convert Jenkins JJB Freestyle jobs to Jenkins Pipelines
- Introduce GitHub Org Plugin
- Simplified Jenkinsfile
- Global Libraries to support Jenkins Pipeline transformation
- Add Unit testing to global-libraries (uncommitted) **
- Snyk integration 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 **

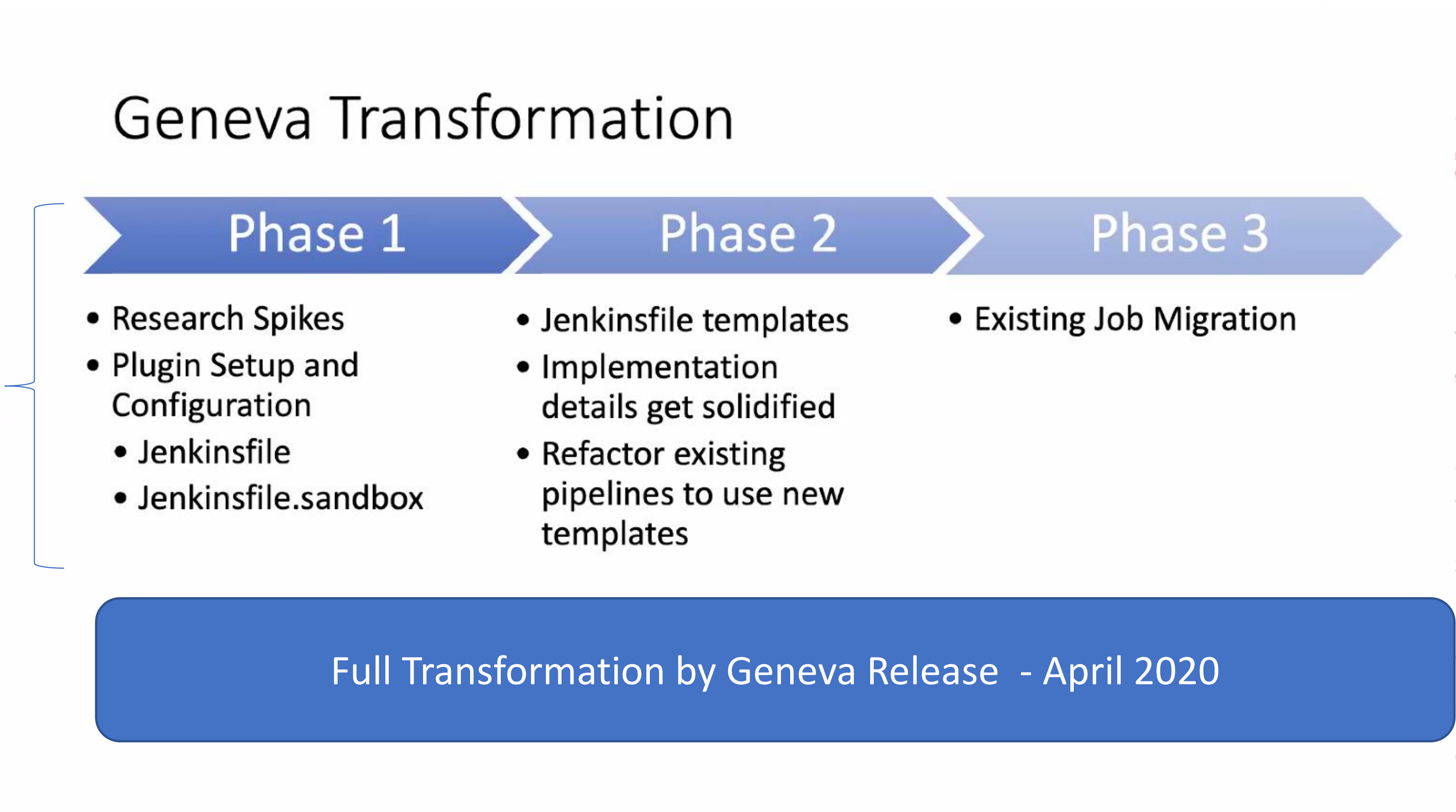
EDGE X FOUNDRY™

Geneva Transformation: Architecture



How long does it take? Is this all Geneva scope?

Phase 1
Work in Progress
Q3 2019



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

EDGE X FOUNDRY™

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)

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

- 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

EDGE X FOUNDRY™

Past / Future Agenda Topics

WW36	
WW37	