LFNETWORKING

Virtual LFN Developer & Testing Forum

June 22 - 25, 2020

Anti-Trust Policy Notice

› Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

› Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at http://www.linuxfoundation.org/antitrustpolicy. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrove of the firm of Gesmer Updegrove LLP, which provides legal counsel to the Linux Foundation.

Requirements subcommittee

Non-functional requirements for Guilin Release

Guilin priorities – SECCOM priorities (Amy)

P1
- [REQ-351, REQ-373] Updates of the languages (java from v8 -> v11 and Python 2.7 -> to 3.x)
- [REQ-323] Updates of directly dependent software components
- [REQ-362] Automated security testing – containers not running as root – SDNC good example
- [REQ-357, REQ-356] Increase the number of CIS Docker Benchmark checks in the Integration healthchecks

P2
- [REQ-361] Secrets management
- [REQ-358] No root access to the DB from main application container. Currently we have some pods (i.e. OOF) that require root access to their mariadb-galera instance for main application to work. This is obviously a security issue. Each application should have its own DB account that allows to access only its own DB.
- [REQ-360] All config files inside the main container should be ReadOnly There are some weird design like in APPC where main container modifies properties provided by the user at runtime. I believe that application configuration should be read only.

P3
- [REQ-349] Increase of code coverage – each project was supposed to propose a % feasible for them and follow the actions to achieve it.
- [REQ-350] CII badging

P4
- [REQ-368] SECCOM initiative (High Priority): service mesh recommendation
- [REQ-375] SECCOM initiative: ONAP MVP
- [REQ-377] User access management
- [REQ-376] Flow management
- [REQ-374] Logs management

Ongoing:
- Fix new OJSIs
- Enable HTTPS on all new external interfaces

Continue hardcoded passwords removal (Krzysztof Opasiak, Sylvain Desbureaux)

ONAP joint effort with SECCOM and OOM

- Continue work related to passwords
  - Started in F release by removing mariadb-galera and postgres passwords
  - Should be continued in G
- Non-Functional Requirements for Guilin:
  1. All new passwords must use common secret template
  2. No hardcoded certificates in docker/OOM repo. All projects must use certInitializer template to retrieve certificate at the deployment time
  3. Project teams be prepared to help (mostly with knowledge on passwords/project) if we manage to touch your component.
  4. Any team that would like to work on their passwords is more than welcomed. Contact OOM Team and we'll do our best to ramp up you quickly.
- More on this in "OOM Status update: consequences on other components"

No root (superuser) access to database from application container (Krzysztof Opasiak, Sylvain Desbureaux)

ONAP joint effort with SECCOM and OOM

- Follow “principle of least privilege” as a best practice
  - Standard security practice
  - Application should not require root access to the DB
  - Separate, short running DB init job may be created to init database
- Non-Functional Requirements for Guilin:
  1. Application container must not access DB as a root
  2. Dedicated user with privileges limited to only those really required should be used
  3. Application may initialized DB using root account from a dedicated job container

- More on this in “OOM Status update: consequences on other components”

Replace NodePorts with ingress controller as a default (Krzysztof Opasiak, Sylvain Desbureaux)

ONAP joint effort with SECCOM and OOM

- NodePorts were always considered “temporary” solution
  - It’s impractical to use them in production environment
  - Support for ingress controller is already implemented in OOM
  - All services are reachable via ingress now we want to make them work
- Non-Functional Requirements for Guilin:
  1. All NodePorts has to be switched off by default
  2. All use cases should use ingress to access ONAP from outside
  3. All projects must work properly with ingress (no hardcoded ports, urls etc)

- More on this in “OOM Status update: consequences on other components”

ONAP container repository (nexus) must not contain upstream images (Krzysztof Opasiak, Catherine Lefevre)

ONAP joint effort with Legal subcommittee and SECCOM

- Hosting means distributing
  - -Docker containers contain software under different licenses
  - -Many of them are copyleft licenses thus distributing them requires to prepare license compliance report
  - -Distributing helm chart that deploy those does not violate the license
  - -Distributing Dockerfile that builds it does not violate the license
  - -Distributing (hosting) ready container image does violate the license unless you do GPL compliance for this container
- Non-Functional Requirements for Guilin:
  1. All external images that are used unmodified from their upstream version cannot be hosted in nexus. They should be downloaded directly from dockerhub/other upstream registry instead
- More on this in “License compliance & how to deal with it?”

ONAP projects must use only approved and verified base images (Krzysztof Opasiak, Catherine Lefevre)

ONAP joint effort with Legal subcommittee and SECCOM

- More different base images, more work for everyone

-ONAP currently uses number of different base images (ubuntu, alpine, centos)

-Most of big and complex images is being used without a valid reason

-Most of them contains copyleft licensed components thus distributing them requires to prepare license compliance report

-We should unify and use single (with exception if required) base image (distro) for all onap components

-TSC should agree to set of acceptable licenses

- Non-Functional Requirements for Guilin:

1.All components should use alpine-based (TBD at M1) base images

- More on this in “License compliance & how to deal with it?”

Documenting ONAP APIs (Andy Mayer, Eric Debeau, Adrian O'Sullivan)

ONAP joint effort with Modeling Subcommittee and Documentation Project.

- Improve ONAP API Documentation
  - Developer Friendly
  - Non-Developer Friendly
  - Easy to Find & Easy to Navigate
  - Common and Uniform Documentation Structure and Approach
  - Provides Information on Using the API (e.g., quick start)
  - Try It For Yourself (TIFY) Examples
- Non-Functional Requirements for Guilin:
  1. All components should place externally facing (i.e. interfaces exposed by the ONAP component to either other ONAP components or components external to ONAP) API definitions (e.g. Swagger) in a common path within their Gerrit/Git
     Suggested Path: <Component>/docs/api/swagger/
  2. Apply ReDoc to Swagger and place HTML in Readthedocs for the release
  3. Apply Minimum (Phase 1+) swagger guidelines
     a. See: Proposed Phase 1+ OpenAPI 2.0 / Swagger Style Guide
     b. Use the common insert for the info section (e.g., license info, contact info, etc): Swagger Insert Sample for Info Section

Documenting ONAP APIs: Additional Information

Related JIRAs under the Documentation project for the API Documentation non-functional requirements:

- Epic: https://jira.onap.org/browse/DOC-608
- User Story: https://jira.onap.org/browse/DOC-609
- User Story: https://jira.onap.org/browse/DOC-610
- User Story: https://jira.onap.org/browse/DOC-611

- **Business Impact** - Enables developers, operators and service providers to use leverage ONAP; Improve integration velocity for API client developers; Ease development handoffs;
- **Business Markets** - All developers, operators and service providers can leverage ONAP APIs
- **Funding/Financial Impacts** - Reduction in development and integration expense from using well defined open Interfaces.
- **Organization Mgmt, Sales Strategies** -There is no additional organizational management or sales strategies for this requirement outside of a service providers "normal" ONAP deployment and its attendant organizational resources from a service provider.

Guilin requirements - integration (Morgan)

REQ-355: ONAP projects dealing with GUI must provide GUI test suites
- Today no regression tests provided for these projects
- Important because GUIs give the first impression for new users
- To be integrated in Milestone criteria
- Impacted projects: Portal, SDC, VID,..

REQ-367: Deploy on demand ONAP through CI per use case
- Today use case projects are using the same lab and sometimes are tripping over one another (one project may need staging versions which break the work for the other projects)
- The idea would be to setup an automated chain to allow a per use case on demand deployment in Windriver/Intel Lab
- join session on Windriver/Intel Lab today 2PM and session on Integration priorities Today 2:30 Pm (https://zoom.us/j/98135653372)
- Potential resources bottleneck

REQ-371: Define Robustness and stability metrics, traffic model and run stability CI chain
- The stability test executed on any ONAP release sofar is limited (72 hours / 1 looping test / kubernetes metrics)
- Additional word is needed to qualify ONAP stability within an operational context
- A long duration CI chain is needed
- join session on Integration priorities Today 2:30 Pm (https://zoom.us/j/98135653372)

REQ-378: Clearly split ONAp code and use case code
- Today when you install ONAP, you install also code (BPMN, Policy, Models,..) dealing with use cases you do not really care
- The pre-provisioning for use cases must be better controlled. It shall be possible at installation to include or not samples in the different components
- An ONAP solution shall be cleaneable and reduced to the end user's needs
- First work could be initiated with SDC, SO, Policy, DCAE
- join session on Integration priorities Today 2:30 Pm (https://zoom.us/j/98135653372)

Guilin requirements – OOM and Gating (Sylvain)

- REQ-356 : ONAP shall increase the number of security tests performed during integration testing
  - New tests will be added to on the gating and daily deployments:
    - Certificate renewal date
    - No hardcoded certificate in the container (tentative)

- REQ-356: ONAP shall increase the number of Docker Benchmark tests
  - New tests will be added to on the gating and daily deployments:
    - Base image from authorized image list
    - Used versions from upstream following SECCOM requirement (

Guilin requirements – OOM and Gating (Sylvain)

- REQ-359: Container rootfs must be mounted readOnly
  - As said for several years (https://www.projectatomic.io/blog/2015/12/making-docker-images-write-only-in-production/), it's important for security purpose to have a read only rootfs
  - OOM new templates will enforce this best practice
  - More on this on "OOM Status update: consequences on other components" session

- REQ-360: Application config should be fully prepared before starting the application container
  - Editing config files with sed from docker entrypoint script often causes a lot of silent failures in OOM deployments.
  - Instead, config should be either provided as a ConfigMap and templated using helm or generated in the init container before the main application container comes up.
  - More on this on "OOM Status update: consequences on other components" session

Guilin requirements – OOM and Gating (Sylvain)

- REQ-362: All containers must run as non-root user
  - After starting with containers from OOM on F release, we want to extend this to all containers that are deployed as a part of OOM.

- REQ-363: ONAP components should be able to run without AAF and MSB
  - AAF is not the only possible security solution for ONAP. In some cases ONAP may be deployed behind a reverse proxy or using service mesh.
  - That's why components should be able to work (even in degradated mode in example using HTTP instead of HTTP or without authentication) without AAF available.
  - The same for MSB.
  - It's not the most cloud native solution for accessing services in kubernetes thus it should be possible to deploy ONAP without it and access services using for example API gateway.
  - More on this on « Service Mesh for RBAC and security PoC » session

Guilin requirements – OOM and Gating (Sylvain)

- REQ-365: Containers must have no more than one main process
  - Docker best practice is to have one main process (java, nginx, gunicorn, ...) per container as it allows a fine grained supervision of this process
  - More on this on “OOM Status update: consequences on other components” session
- REQ-366: Containers must crash properly when a failure occurs
  - Kubernetes best practice mandates that when an issue occurs (no access to Database, REST mandatory call fails, bug in code, ...), the container must crash with exit code different than 0
  - More on this on “OOM Status update: consequences on other components” session

LFNETWORKING
Virtual Developer & Testing Forum
June 22 - 25, 2020

Thank You!

LFNETWORKING