ONAP El Alto Release Management Proposal

El Alto Challenges

- Compressed schedule AND mid-cycle release
- Some pervasive changes will impact many projects (Angular, Java & Python upgrades)
- Checklist intensive release management may be too much reporting overhead for El Alto
- Not all projects have significant defect backlogs
- Clock is ticking on fixing identified security defects with in 60 days (CII Badging)

El Alto Proposed Priorities

- TSC approved El Alto release content
  - “Reduce Internal Debt” (Every 4 ONAP releases)
  - Vulnerability issues
  - Documentation
  - Test Automation & CI/CD pipeline
  - Test Coverage including jS
  - Deployment procedure improvements
  - Refactoring
  - JIRA Backlog Reduction (defects, etc.)

Suggested Priorities

- Groom backlogs (pre-El Alto)
- Security related refactoring (CVEs in sprint 1 to be ready for Early Drop)
- Documentation
- Test Automation
- Upgrade process improvements
- Defect Reduction

El Alto Proposed Release Progress Tracking

Use first 2 weeks to groom backlog and determine the El Alto scope for each project

Examples:

Epic - Security Related Refactoring (top pri. for Early Drop)

User stories - Specific changes planned placed into sprints 1 - 3

Epic - Documentation

User Stories - Set of documentation jiras

El Alto Proposed Early Drop (ED)

- Focus early on security defects to meet the 60 day resolution window as required for CII Badging
- Keep the Dublin stability in the build from start of El Alto through the release (Don't Break the Build)
- Component release at the end of test cycle #2
  - Requires Release Notes, coverage goals, CII Badging review, etc.
- Deliver ED using the Integration Waves defined in Dublin
  - Wave 1 - SDNC, SO, DMaaP, AAI
  - Wave 2 - APPC, Policy, Clamp, OOF, DCAE
  - Wave 3 - The rest

El Alto SoW proposal to ONAP community

1. Work on OJSI tickets to solve penetration tests findings/CVEs – 60 days ticking clock!
2. Update ONAP environment (java, python, kubernetes, docker, ubuntu, alpine…) to commonly agreed versions
3. Focus on known vulnerabilities by **upgrading** libraries to commonly agreed versions – coordination with ONAP Release Manager, stakeholders: OOM, Integration, PTLs
4. Focus on known vulnerabilities by **replacing** libraries with commonly agreed versions and components – coordination with ONAP Release Manager, stakeholders: OOM, Integration, PTLs
5. Update CII Badging answers within release LCM
6. Contribute to ONAP security communication matrix creation

Courtesy of Pawel@SECCOM

El Alto SoW proposal to SECCOM community

1. Support ONAP community in achieving El Alto goals
2. Update Security by Design with: CII Badging gates, https communication update for M3, revamping of the vulnerability review tables
3. Update Oparent.pom file with SECCOM recommended and consulted versions, including dependency management from Maven
4. Create communication matrix based on inputs from PTLs and validate it with scripts with Integration team – coordinate with architecture team
5. Support Integration team with security testing enhancements
6. Consider security features scope and perform comparative analysis of AAF and ISTIO
7. Design CMPv2 implementation – stakeholder: AAF
8. Benchmark Nexus-IQ with Whitesoftware for Dan's project
9. Continue known vulnerabilities management and CII Badging answers reviews

Courtesy of Pawel@SECCOM


ONAP
OPEN NETWORK AUTOMATION PLATFORM