Remediating Known Vulnerabilities in Third Party Packages

In order to improve the security of the ONAP code base, projects are to focus on upgrading the third party packages that are direct dependencies. This change has been implemented in the Frankfurt release. Previous releases required vulnerability analysis in addition to package upgrades. Beginning with the Frankfurt release, the remediation of known vulnerabilities in third party packages will be managed as follows.

- Changes from Remediating Known Vulnerabilities in Third Party Packages in the Beijing through El Alto releases
    - There is no requirement to provide effective/ineffective analysis until there are tools to support the analysis.
    - There is no requirement to create vulnerability review tables.
- ~~M2~~ M0
    - Projects identify the direct dependencies (packages) in each project component.
        - NexusIQ provides a list of all packages used in a component.
        - Maven creates dependency tree that identifies direct dependencies as the "left-most packages".
    - Projects identify the most recent version of the direct dependencies
        - Tools to help choose the upgrade version
            - NexusIQ
            - Maven (https://mvnrepository.com)
    - SECCOM updates oparent.pom to include the most recent version of included packages that are available at M2.
    - SECCOM prioritization of package upgrades
        - Priority 1: outdated packages containing a Critical vulnerability
        - Priority 2: outdated packages containing a Severe vulnerability
        - Priority 3: all other outdated packages
    - Each project opens Jiras ticket(s) to update older package versions in direct dependencies.
        - **Note: There is no requirement to upgrade transitive dependent packages.**
        - ~~There must be a separate Jira for each package to be upgraded in each project repository.~~
        - Required information in Jira ticket:
            - Old and new version numbers
            - Label of "ComponentUpgrade"
            - Fix Version = release under development
    - ~~**Exceptions:** The project must request a TSC exception for each direct dependency that cannot be upgraded by M1.~~
        - ~~Required information in the "ComponentUpgrade" Jira ticket with an exception:~~
            - ~~The reason that the package cannot be upgraded,~~
            - ~~Fix Version = the next release to be developed~~
- M1
    - Package updates must be complete complete
    - **Outdated packages that are not updated require TSC exceptions:** The project must request a TSC exception for each direct dependency that cannot be upgraded by M1.
        - Required information in the "ComponentUpgrade" Jira ticket with an exception:
            - The reason that the package cannot be upgraded,
            - Fix Version = the next release to be developed
- ~~M4~~
    - Project will close each Jira ticket that has been completed.
    - SECCOM will create a report of the status of all "ComponentUpgrade" Jiras for the release.
    - ~~Open tickets will require a TSC exception~~
        - ~~Jiras with M4 exceptions must contain the same information documented above~~
- Readthedocs
    - All projects will list all CVEs (CVE number only) associated with third party packages in the readthedocs in the Third Party Vulnerabilities section.
        - Vulnerabilities are listed in the NexusIQ reports for each project repository scanned

The CLAMP team will investigate writing a script to automatically generate project-level Jira tickets for all direct dependencies.

Frankfurt Release: The CLAMP team wrote a script that generates user stories for each outdated direct dependency in a project and links them to an epic for the project. Example epic: CLAMP-601. Example user story: CLAMP-602. Each user story identifies and outdated package and the newest version.

> **Proposal:** The CLAMP team will run the script for all projects during the week of 10 February, creating user stories and epics for each project. SECCOM will measure progress for REQ-263 using the automatically generated tickets.

The script is an open source project which can be found here. The CLAMP team welcomes contributions.