Portal Platform Security/Vulnerability Threats

This template is intended to be used to document the outcome of the impact analysis related to the known vulnerability reported by Nexus-IQ (CLM tab in Jenkins). Nexus-IQ can identify the known vulnerabilities contained in the components use by onap components.

This table will be presented to TSC at Code Freeze milestone (M4) to the TSC.

It is recommended to **first update to the latest version** of the third party components available. In case the latest third party components still reports some vulnerabilities, you must provide an impact analysis as illustrated in the example below.

In the case where you have nested third party components (a third party component embedding another third party component) and there is **NO CVE** number for the upstream third party component (meaning the third party component you are embedding), it is recommended to open a vulnerability issue on the upstream third party component.

The following table is addressing 2 different scenarios:

- Confirmation of a vulnerability including an action
- False Positive

The information related to Repository, Group, Artifact, Version and Problem Code are extracted from the CLM report (see the below screenshot)

Repository	Group	Impact Analysis	Action
portal	com.fasterxml.jackson.core	False positive. Analysis: This vulnerability is not exposed from the portal's code, because 1. The portal does not pass any untrusted data for deserialization, as there is XSS/XSRF validation enabled in the portal's backend code. 2. and the default typing (ObjectMapper.setDefaultTyping()) is not called as we use concrete java types. 3. and we use Spring Security 4.2.3 as recommended in the nexus-iq report. Spring version 4.2.3 will take care of this. Comments from Nexus-IQ: Spring Security has provided their own fix for this vulnerability (CVE-2017-4995). If this component is being used as part of Spring Security, then you are not vulnerable if you are running Spring Security 4.2.3.RELEASE or greater for 4.x or Spring Security 5.0.0.M2 or greater for 5.x.	Not vulnerable in ONAP
	~~javax.servlet~~	~~No clear instruction on what to upgrade to~~	Vulnerability removed as per nexus-iq
~~Portal-SDK~~	~~commons-httpclient~~	~~The recommendation is to use org.apache.httocomponents. But we are not directly using the said package/class. It comes as a dependency.~~	Vulnerability removed as per nexus-iq
Portal	moments	All available versions of moment.js are vulnerable. Upgrade is not an option. Analysis: Not vulnerable as all our date fields are reformatted and validated before being submitted. See below CVE 185 information: The moment package is vulnerable to Regular Expression Denial of Service (ReDoS). The monthsShortRegex(),monthsRegex(),weekdaysRegex(),weekdaysShortRegex(), and weekdaysMinRegex() functions in the moment.js, moment-with-locales.js, and regex.js files use a vulnerable regular expression while parsing the date input. A remote attacker can exploit this vulnerability by crafting a date input containing a very long sequence of repetitive characters which, when parsed, consumes available CPU resources and results in Denial Of Service.	Not vulnerable in ONAP
Portal, Portal-SDK	angular	Analysis: Cannot upgrade angular as this will require changes on all the Portal pages. From our analysis the vulnerability cannot be exploited because the portal application follows the below design recommendations provided by nexus-iq report.	Not vulnerable in ONAP

Portal	commons-beanutils	All available versions of common-beanutils are vulnerable. Upgrade is not an option. Analysis: The portal code do not use classloader so it is not vulnerable in ONAP. CVE CWE: 20 Description from CVE Apache Commons BeanUtils, as distributed in lib/commons-beanutils-1.8.0.jar in Apache Struts 1.x through 1.3.10 and in other products requiring commons-beanutils through 1.9.2, does not suppress the class property, which allows remote attackers to "manipulate" the ClassLoader and execute arbitrary code via the class parameter, as demonstrated by the passing of this parameter to the getClass method of the ActionForm object in Struts 1.	Not vulnerable in ONAP
Portal-SDK	org.apache.poi	Analysis: Not vulnerable as we do not use POI to read documents. We use only to generate XLS from our own data. CVE CWE:399: Apache POI in versions prior to release 3.17 are vulnerable to Denial of Service Attacks: 1) Infinite Loops while parsing crafted WMF, EMF, MSG and macros (POI bugs 61338 and 61294), and 2) Out of Memory Exceptions while parsing crafted DOC, PPT and XLS (POI bugs 52372 and 61295).	Not vulnerable in ONAP
~~Portal, Portal-SDK~~	~~com.codehaus.jackson~~	~~Need Exception.~~	Vulnerability removed as per nexus-iq

CLM Report