Analyzing and securing your application dependencies

Red Hat Developers Documentation Team

2018-12-11 13:07:31 UTC

Table of Contents

Using data analytics to develop your application 2
1. Using analytics in the development stage 3
2. Using analytics at the deployment stage 4

This guide contains instructions on using CodeReady Toolchain analytics engine to identify dependencies with CVEs and License issues, take corrective action, and augment your development stack based on the provided insights.

Using data analytics to develop your application

After you create or import a codebase in CodeReady Toolchain the Analytics engine analyzes your codebase and identifies dependencies with security vulnerabilities and license issues. It also provides insights on other dependencies that can add value to your stack.

Chapter 1. Using analytics in the development stage

CodeReady Toolchain provides an integrated development environment in the form of an Eclipse Che workspace to develop your codebase.

CodeReady Toolchain analytics engine analyzes your stack and its dependencies at the development stage within your Che workspace. It flags dependencies with security vulnerabilities and suggests secure, alternate versions to replace them while you develop your application codebase. You can use this analysis to develop a secure codebase with appropriate dependencies.

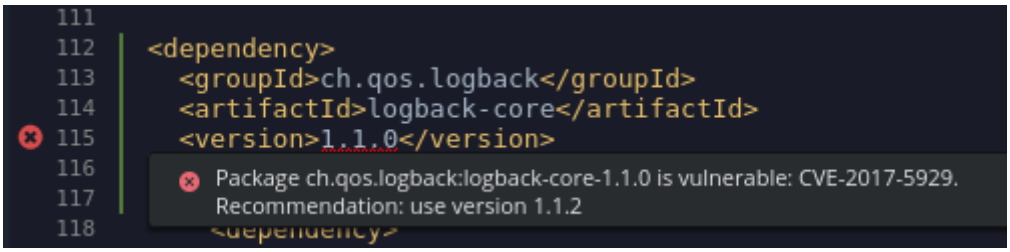
Prerequisite

1. Create a Che workspace to develop your stack.
2. Run your application in the Che workspace.

Procedure

You can access inputs from CodeReady Toolchain analytics within the Che IDE as follows:

1. In your Che workspace, open the manifest file of your project, for example, `pom.xml` for a Maven Stack, `package.json` for NPM, or `requirements.txt` for Python.
2. Make modifications to your code. CodeReady Toolchain analyzes the stack, flags the dependency if it has any security vulnerabilities, and suggests an alternate secure version.
3. If you see an error icon ( ) move the mouse pointer over the icon to see the Common Vulnerabilities and Exposures (CVE) for the flagged dependency and the suggested alternate version.



4. Update the dependencies to the suggested version.

Chapter 2. Using analytics at the deployment stage

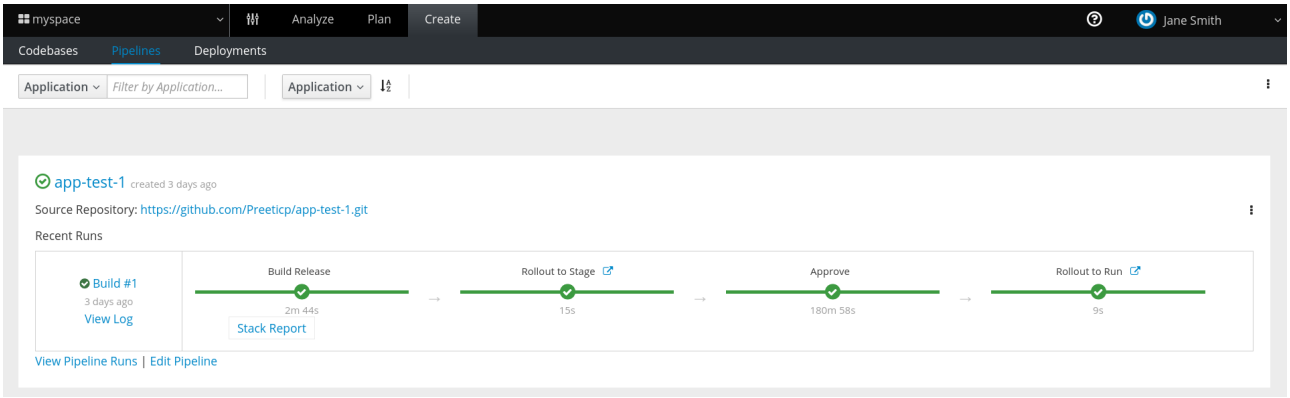
When you create a new quickstart project, a new build is executed. CodeReady Toolchain analytics is triggered during the **Build Release** stage of the build pipeline. It analyzes your stack and its dependencies and provides a detailed report on the security issues and license issues affecting your dependencies along with insights on alternate and additional dependencies suitable for your stack. Use the stack report to make informed decisions about the open source dependencies in your stack.

Prerequisites

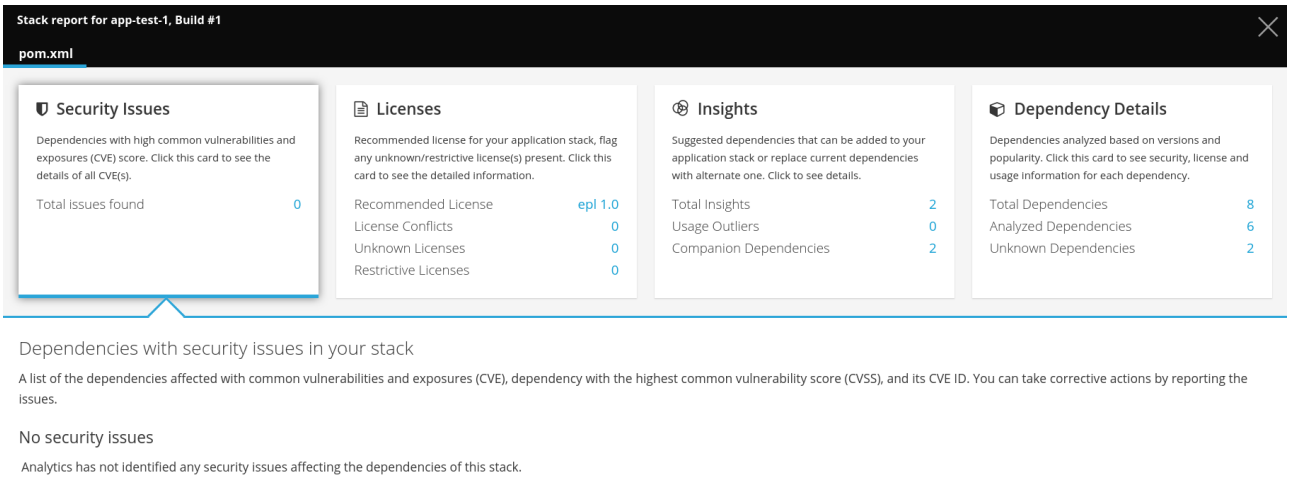
Create an application or import an existing codebase to CodeReady Toolchain.

Procedure

1. Navigate to **Create › Pipelines** to view the pipeline builds for your project.
2. In the **Pipelines** view, under the **Build Release** stage, click **[Stack Report]** to see the analysis report for your entire stack.



The report displays a summary of four key aspects relevant to your stack in the form of cards: **Security Issues**, **Licenses**, **Insights**, and **Dependency Details**.



3. Click each of the cards to see a detailed analysis report for your stack and its dependencies:

Security Issues

This card highlights the number of security issues in your stack, the highest Common Vulnerability Scoring System (CVSS) score, and the number of dependencies with this high score.

Click the **Security Issues** card to see details on:

- Dependencies with security issues
- The number of Common Vulnerabilities and Exposures (CVEs) found in each of your dependencies
- The highest Common Vulnerability Scoring System (CVSS) score in your dependency and its CVE ID.

A CVSS score highlighted in:

- Red indicates a severe vulnerability, with a score in the range of 7 - 10.
- Orange indicates a moderate vulnerability, with a score in the range of 4 - 6.9.

Licenses

This card suggests an appropriate stack level license and flags conflicting, unknown, and restrictive licenses (licenses that are not commonly used in similar stacks or that do not work well with the stack's representative license) affecting your stack.

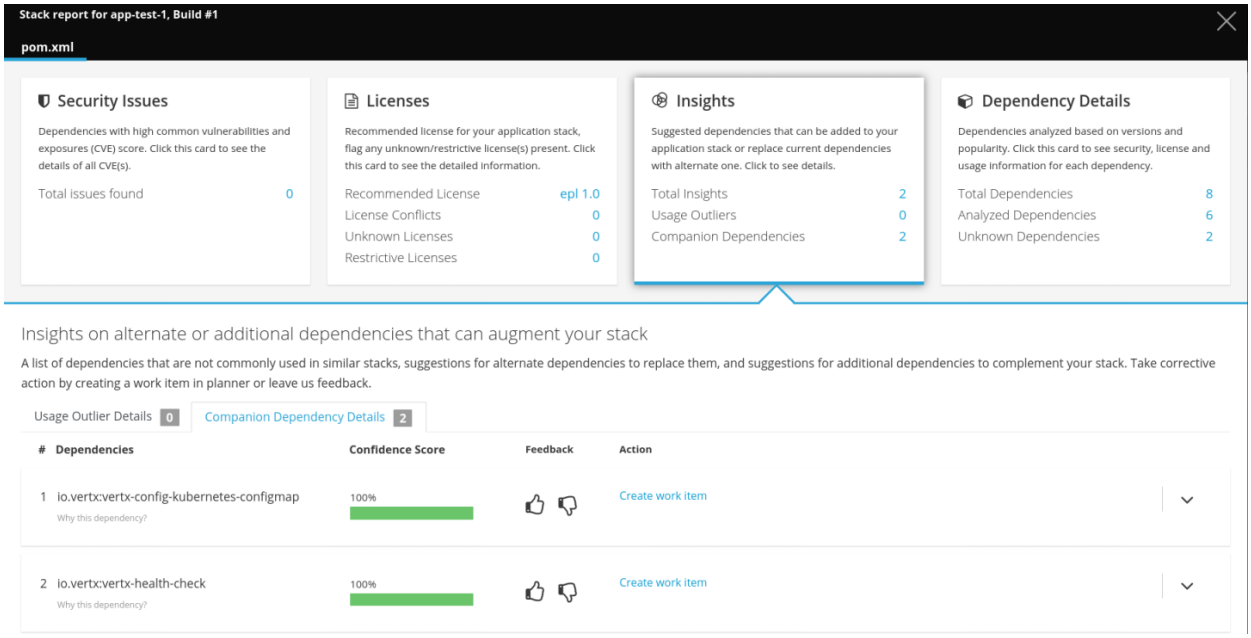
Click the **Licenses** card to see the following detailed information:

- The **Conflicting license(s) details** tab is displayed by default. It lists dependencies that conflict with licenses of other dependency or with the stack level license. It highlights the licenses which are affected in the dependency, and suggests alternate dependencies that go well with your stack, and avoid such conflicts.
- Click the **Unknown license(s) details** tab to see the list of dependencies with licenses unknown to CodeReady Toolchain. It highlights the affected or unknown license, and suggests alternate dependencies to replace such dependencies.

Insights

Based on the analysis of other similar stacks, this card identifies usage outliers (dependencies not commonly used in similar stacks and that rarely go well together) in your stack and also highlights the number of companion (additional) dependencies that could augment your stack.

Click the **Insights** card to see the following insights:



- The **Usage outliers details** tab is displayed by default. It identifies and lists dependencies in your stack that are not commonly used in similar stacks or that do not work well with other dependencies in the stack. It suggests alternate dependencies, suitable to your stack, to replace them. The **Confidence score** depicts the confidence of CodeReady Toolchain analytics on the suitability of the alternate dependency to your stack.
- Click the **Companion component details** tab to see a list of additional dependencies that you can add to your stack to enhance it. Based on the confidence score, you can decide on the suitability of the dependency to your stack and add it. You can also provide your feedback on the suggested dependencies.

Dependency Details

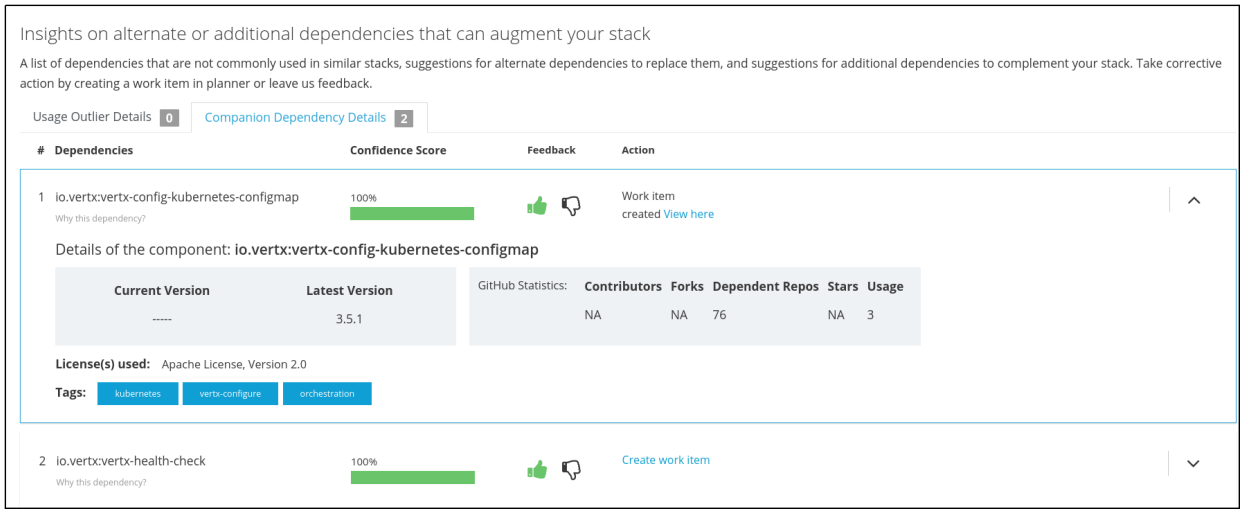
This card lists the number of dependencies analyzed by CodeReady Toolchain and those unknown to it.

Click the **Dependency Details** card to see details on:

- The **Analyzed dependency details** tab is displayed by default. It lists details of all the dependencies analyzed by CodeReady Toolchain and the **Components check** section highlights security, usage, and license issues in them. It suggests alternate dependencies to replace dependencies with usage and license issues.
- The **Unknown dependency details** tab lists dependencies unknown to CodeReady Toolchain analytics.

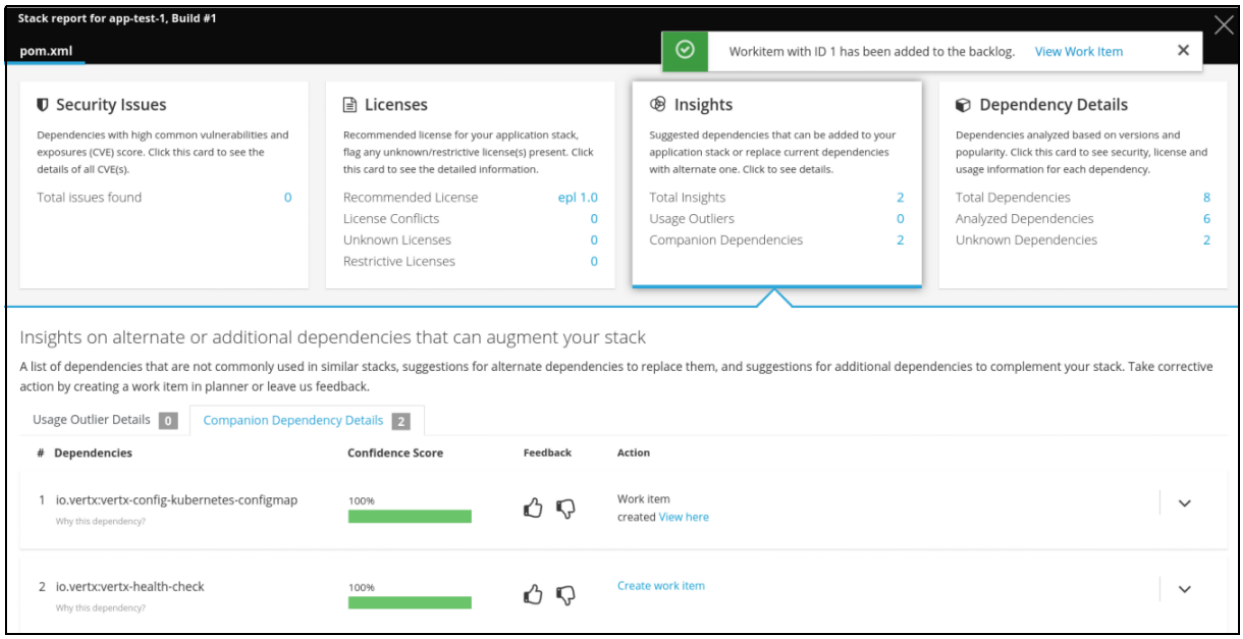
4. Expand the arrow adjacent to the dependency to see the following detailed information about the existing or the suggested companion dependency:

- The current and latest available version of the dependency
- GitHub statistics relevant to it that help assess its popularity
- Licenses used by the dependency
- Tags associated with the dependency



In the case of usage outliers, details of the suggested replacement dependency are displayed along with those of the existing dependency. These statistics help you compare the existing dependency with the alternate dependency and make a smart choice for your stack.

5. To act on the analytics and insights provided by the report:
   - In the **Security Issues** detailed view, click **Report an issue** to report the security vulnerability as an issue in the CodeReady Toolchain planner. This ensures all your team members are aware about the security issue and can take the necessary follow-up action.
   - In the **Insights** detailed view, click **Create work item** to create auto-populated issues in CodeReady Toolchain planner for adding the suggested alternate or companion dependency.



You can now act upon the relevant input provided by the stack report and enhance your development project.