CONTRAST SECURITY

CONTRAST OSS HELPS DEVOPS MANAGE AND TRIAGE HIDDEN OSS LIBRARY RISK

SOLUTION BRIEF

EXECUTIVE OVERVIEW

The adage, "Security teams don't know what they don't know," rings true when it comes to open-source software (OSS) vulnerabilities. Legacy software composition analysis (SCA) tools only provide a point-in-time assessment of open-source components and cannot continuously assess application security (AppSec) throughout the development life cycle. Lack of insight into all the different dependent OSS libraries that get pulled into the application during continuous integration/continuous deployment (CI/CD) processes creates enormous visibility gaps in the application layer. Security and development teams often have no insight if libraries are being used when the application is run. This creates major backlogs as DevOps teams cannot efficiently prioritize which vulnerabilities need to be immediately addressed.

As part of the Contrast DevOps-Native AppSec Platform, Contrast OSS helps organizations prioritize critical vulnerabilities by tracking the libraries that actually get used by applications during runtime operation. It also provides development and security teams with comprehensive visibility of all OSS components to better understand the depth of risk that library dependencies can produce.

> ***Many people download open-source libraries, assuming they are safe—only to discover they're infested with malware.[1]***

OSS LIBRARY DEPENDENCIES CREATE VISIBILITY PROBLEMS

Software today is often built from as much as 90% open-source code—including hundreds of discrete libraries in a single application.[2] Developers routinely use open-source libraries (such as Apache) to introduce functionality to their applications at speed. What many developers do not know, however, is that for the top-level library to deliver on its functionality, it must call on directly dependent libraries. These libraries, in turn, may be linked to transitive dependent libraries—creating dependencies of dependencies.

Because these dependent libraries may include vulnerabilities, this structure creates layers of unaccounted risk. AppSec teams typically have no insight into which of these libraries actually gets used when the application is running. This lack of visibility creates confusion when it comes to prioritizing vulnerability remediation—especially since any vulnerable libraries that get used when the application is running should garner the most immediate attention for remediation.

> ***The National Institute of Standards and Technology (NIST) sponsors the National Vulnerability Database (NVD)—a public repository for information on software vulnerabilities, including those in open-source software.[3]***

Security teams need to know about potentially vulnerable libraries in order to mitigate risk before shipping applications to production. Development teams need insight into which libraries (whether they be top-level or dependent) carry vulnerabilities so that they can better focus their remediation efforts. The ultimate goal is mitigation of software risk earlier in the software development life cycle (SDLC). This ensures that developers spend less time fixing security defects and more time shipping code. It costs six times more to fix a bug found during implementation than to fix one identified during design; 15 times more if it is identified in testing; and 100 times more during regular maintenance once the code is in production.[4]

CONTRAST OSS ENABLES DEVOPS TO FOCUS ON CRITICAL LIBRARY VULNERABILITIES

Solving the key problems of OSS means prioritizing vulnerable components that are actually used by the application for remediation effort by DevOps. At the same time, it also translates into deprioritizing work on those components that are not in use. Accurately determining usage of OSS libraries requires directly observing and measuring the behavior of the application runtime. By assessing the application while it is running, teams can observe which top-level libraries and which dependencies are accessed in order to properly prioritize remediation efforts.

As an instrumentation tool embedded within the application itself, **Contrast OSS** can automatically perform software composition analysis (SCA) during the application runtime. It discovers all active open-source components and reports an exact bill of materials to Contrast TeamServer, which provides centralized management and reporting for policy-based control. There is no need to run separate assessments with different tools. Because of its continuous assessment of a running application, Contrast OSS is more accurate than SCA tools relying solely on inspecting a "point-in-time" project configuration file.

> ***Static application security testing (SAST) that guesses if a component might be used by the application is plagued with false positives and false negatives. These inaccuracies create noise that interferes with effective and automatic prioritization of vulnerabilities, while increasing an application's risk posture.[5]***

Additionally, Contrast OSS reports on the extent to which each library is used. This occurs by detailing the total number of available library classes and the total called at runtime. With this information, security teams can prioritize remediation of libraries, ensuring the most critical libraries (those that introduce the greatest risk to the application) are prioritized first.

Contrast OSS also offers a means for developers to assess their applications at the earliest stages of the SDLC. Leveraging instrumentation, Contrast OSS detects if a particular library poses a risk to the running application. If a library is both vulnerable and called at runtime, then it should be prioritized for remediation.

But in order to remediate, the security team needs to know how the library got there. This is where Contrast's command-line interface (CLI) tool comes into play. If the vulnerable library is a top-level dependency, that top-level library needs to be updated. If the dependency is transitive, then the CLI's dependency tree helps show which library needs to be updated.

> ***The Open Web Application Security Project (OWASP) is a nonprofit foundation that works to improve the security of software. OWASP notes that using old versions of open-source components with known vulnerabilities has been one of the most critical web application security risks in recent years.[6]***

EXPEDITING WORKFLOWS, CONSERVING RESOURCES

Unlike outdated toolsets that assess open-source risks at a fixed moment in time, Contrast OSS continuously evaluates open-source libraries within actual running applications. This internal runtime visibility detects which open-source libraries are called in runtime, if they are vulnerable, and if they expose the organization to unnecessary security risks or legal problems due to open-source licensing complications.

As part of Contrast's instrumentation-based AppSec platform, Contrast OSS delivers a more accurate understanding of critical OSS libraries to help expedite CI/CD pipelines by prioritizing the vulnerabilities that actually matter to the running application. Focusing the remediation efforts of DevOps teams saves time and money while simultaneously improving the security of applications.

1 Gilad David Maayan, "How to Make Your CSO Happy with Your Open Source Components," CPO Magazine, August 28, 2019.

2 Frank Nagle and Jenny Hoffman, "The Hidden Vulnerabilities of Open Source Software," Harvard Business School, February 24, 2020.

3 "National Vulnerability Database," NIST, accessed April 15, 2020.

4 Mukesh Soni, "Defect Prevention: Reducing Costs and Enhancing Quality," iSixSigma, accessed April 16, 2020.

5 Augusto Barros, "From my Gartner Blog – Considering Remediation Approaches For Vulnerability Prioritization," Security Boulevard, May 2, 2019.

6 Kirk Jackson, "Introduction to the OWASP Top Ten," OWASP, February 9, 2020.

CONTRAST SECURITY
240 3rd Street
Los Altos, CA 94022
888.371.1333

Contrast Security is the world's leading provider of security technology that enables software applications to protect themselves against cyberattacks, heralding the new era of self-protecting software. Contrast's patented deep security instrumentation is the breakthrough technology that enables highly accurate assessment and always-on protection of an entire application portfolio, without disruptive scanning or expensive security experts. Only Contrast has sensors that work actively inside applications to uncover vulnerabilities, prevent data breaches, and secure the entire enterprise from development, to operations, to production.