{deepfactor}

PARTNER BRIEF

Deepfactor + Synopsys Black Duck

Deepfactor Developer Security enables engineering teams to quickly discover and resolve security vulnerabilities, software supply chain risks, and compliance violations by observing running applications during development and testing. Deepfactor integrates with Synopsys Black Duck to reduce alert volume and improve remediation time by providing developers with contextual runtime information and access to Black Duck Security Advisories (BDSA).

// Deepfactor Integrates with Synopsys Black Duck to Help Developers Prioritize Cloud Native Supply Chain Security Risks

With engineering teams embracing cloud native development and rapidly adopting open source software, releases are increasing in both frequency and complexity. In order to understand and address application risks, development teams can use Software Composition Analysis (SCA) to discover vulnerable libraries and dependencies. However, augmenting SCA results with dynamic, contextual analysis of the running application can help developers navigate the results:

- > Save time by identifying and prioritizing vulnerabilities discovered in active code by observing the application during runtime
- > Help developers identify and triage vulnerable and insecure code by providing contextual, application-aware insights such as system calls and stack traces
- > Extend developer visibility into security risks spanning application code, container images, web/API interfaces and compliance

// The Integration

Deepfactor integrates with Synopsys Black Duck, which replaces vulnerability information from the National Vulnerability Database (NVD) with enhanced Black Duck Security Advisories (BDSA) that are researched and analyzed by the Synopsys Cybersecurity Research Center (CyRC) for completeness and accuracy.

Deepfactor utilizes a language-agnostic library to observe every thread, process, container, and pod of cloud native applications without requiring agents or privileged kernel code. With this information, Deepfactor creates a Dynamic Bill of Materials (DBOM), which includes usage information for dependencies and OS packages. By default, Deepfactor compares the results with data from the NVD to provide developers with valuable information on known and common vulnerabilities. The vulnerabilities are then mapped to cloud security standards and contextual runtime information, such as stack traces and method tracing, to help engineering teams pinpoint vulnerable code and prioritize remediation.

The Solution

{deepfactor} + SYNOPSYS® BLACK DUCK

The Benefits

- > Reduces alert volume to help developers prioritize and remediate application vulnerabilities
- > Identifies exploitable code by observing which packages and libraries are loaded when the application is running
- > Provides developers with an integrated solution that delivers application-aware insights spanning application code, dependencies, container images, web/API interfaces and compliance

The Requirements

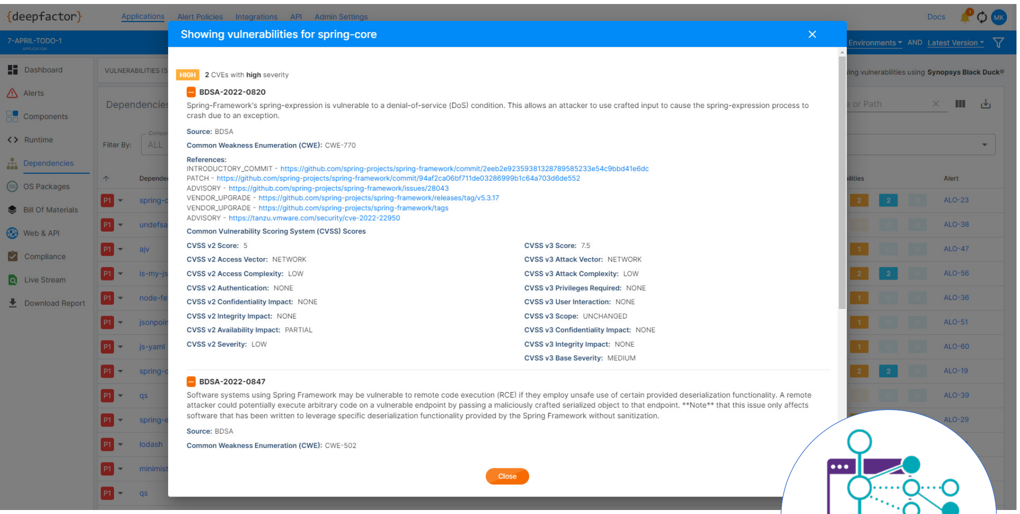
Synopsys Black Duck

- > Product License
- > Hostname / IP Address
- > API Access Token

// The Advantages

Help your engineering team manage the security, operational, and license compliance risks that come from the use of open source and third-party code. With Synopsys Black Duck and Deepfactor Developer Security customers can:

- Prioritize and enhance SCA alerts results based on the behavior of the running application to accelerate resolution of vulnerabilities and reduce the impact of security on development
- Provide contextual and actionable security insights to developers directly within the CI/CD pipeline
- Identify insecure code and behavior violations in first and 3rd party code that can't be found with static scanning
- Reduce the risk of data breaches, software supply chain attacks, and compliance violations by identifying security risks earlier in the software development lifecycle



Showing vulnerabilities for spring-core

HIGH 2 CVEs with **high** severity

BDSA-2022-0820

Spring-Framework's spring-expression is vulnerable to a denial-of-service (DoS) condition. This allows an attacker to use crafted input to cause the spring-expression process to crash due to an exception.

Source: BDSA

Common Weakness Enumeration (CWE): CWE-770

References:
INTRODUCTORY_COMMIT - https://github.com/spring-projects/spring-framework/commit/2eeb2e92359381328789585233e54c9bbd41e6dc
PATCH - https://github.com/spring-projects/spring-framework/commit/94af2ca06bf711de03286999b1c64a703d6de552
ADVISORY - https://github.com/spring-projects/spring-framework/issues/28043
VENDOR_UPGRADE - https://github.com/spring-projects/spring-framework/releases/tag/v5.3.17
VENDOR_UPGRADE - https://github.com/spring-projects/spring-framework/tags
ADVISORY - https://tanzu.vmware.com/security/cve-2022-22950

Common Vulnerability Scoring System (CVSS) Scores

CVSS v2 Score: 5
CVSS v2 Access Vector: NETWORK
CVSS v2 Access Complexity: LOW
CVSS v2 Authentication: NONE
CVSS v2 Confidentiality Impact: NONE
CVSS v2 Integrity Impact: NONE
CVSS v2 Availability Impact: PARTIAL
CVSS v2 Severity: LOW

CVSS v3 Score: 7.5
CVSS v3 Attack Vector: NETWORK
CVSS v3 Attack Complexity: LOW
CVSS v3 Privileges Required: NONE
CVSS v3 User Interaction: NONE
CVSS v3 Scope: UNCHANGED
CVSS v3 Confidentiality Impact: NONE
CVSS v3 Integrity Impact: NONE
CVSS v3 Base Severity: MEDIUM

BDSA-2022-0847

Software systems using Spring Framework may be vulnerable to remote code execution (RCE) if they employ unsafe use of certain provided deserialization functionality. A remote attacker could potentially execute arbitrary code on a vulnerable endpoint by passing a maliciously crafted serialized object to that endpoint. **Note** that this issue only affects software that has been written to leverage specific deserialization functionality provided by the Spring Framework without sanitization.

Source: BDSA

Common Weakness Enumeration (CWE): CWE-502

Deepfactor showing vulnerabilities using **Synopsys Black Duck**

SYNOPSYS® BLACK DUCK

// Next Steps

Pricing available upon request—sales@deepfactor.io

For more information on this integration, please visit our documentation.

About Deepfactor

Deepfactor is a developer security platform that enables engineering teams to quickly discover and resolve security vulnerabilities, software supply chain risks, and compliance violations early in development and testing. Requiring no code changes, the Deepfactor runtime observability technology seamlessly plugs into cloud native architectures, enabling developers to identify, prioritize, and remediate application risks.

About Synopsys

Synopsys offers the most comprehensive solution for building integrity—security and quality—into your SDLC and software supply chain. We've united leading testing technologies, automated analysis, and experts to create a robust portfolio of products and services. This portfolio enables companies to develop customized programs for detecting and remediating defects and vulnerabilities early in the development process, minimizing business risk and maximizing engineering productivity. We don't stop when the test is over. As a recognized leader in application security testing, we offer onboarding and deployment assistance, remediation guidance, and training solutions that empower you to optimize your investment.

{deepfactor}

Deepfactor is a developer security platform that enables engineering teams to quickly discover and resolve security vulnerabilities, supply chain risks, and compliance violations early in development and testing. For more information, follow Deepfactor on **Twitter** or **LinkedIn** or **contact us**.