CASE STUDY

Identifying and Addressing Known Vulnerabilities in ICS Software

Industrial control systems (ICS) support much of the world’s critical infrastructure and manufacturing capacity, making software reliability and security extremely important in this space.

This case study focuses on an arbitrary ICS-focused software development kit from the internet which Synopsys staff downloaded and analyzed. Synopsys discovered the presence of hundreds of known vulnerabilities in third-party components that could have been eliminated prior to release through the use of software composition analysis.

Abstract

Industrial control systems (ICS) support much of the world’s critical infrastructure and manufacturing capacity, making software reliability and security extremely important in this space.

This case study focuses on an arbitrary ICS-focused software development kit from the internet which Synopsys staff downloaded and analyzed. Synopsys discovered the presence of hundreds of known vulnerabilities in third-party components that could have been eliminated prior to release through the use of software composition analysis.

Solution evaluation

Software composition analysis is critical due to the fact that an estimated 70% to 90% of software applications today use third-party libraries. Users of third party components rarely allocate resources to ensure the components they are consuming are secure because they falsely assume that security and quality testing are upstream responsibilities. By applying software composition analysis techniques, you and your organization can verify that third-party components within software are secure and thus avoid absorbing unnecessary security risks.

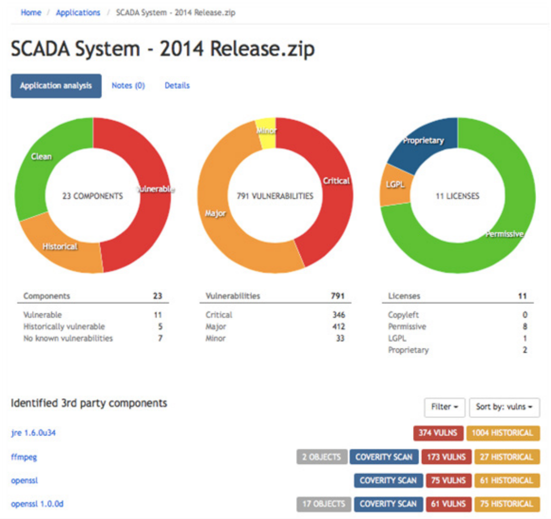
Synopsys Software Composition Analysis (Protecode Supply Chain™) relieves the burden of software component testing by providing a comprehensive bill of materials containing third-party components, their location, and their version in binary packages. Using the US NIST National Vulnerabilities Database (NIST NVD) and various other databases, Synopsys Software Composition Analysis (SCA) Supply Chain analysis engine enumerates known vulnerabilities within each third-party component. For every known vulnerability found, Synopsys SCA Supply Chain provides its Common Vulnerabilities and Exposures (CVE) number, where details of the vulnerability and a criticality score are provided.

Discovery phase

The engineering and support staff at Synopsys regularly test our SCA Supply Chain by acquiring binary packages and uploading the contents into its cloud services. The results of the scan are analyzed to not only determine the effectiveness of our SCA Supply Chain, but to also assess the results and relay critical findings to the developers of the analyzed package(s).

In 2014, a Synopsys engineer downloaded a SCADA (supervisory control and data acquisition) software package from the vendor’s developer website. The website advertised that there were over 20,000 licensed users globally, and listed some of their marquee customers, who were distributed among multiple critical infrastructure sectors such as airports and water management.

Upon scanning the downloaded software package, it was discovered that over 700 known vulnerabilities affected the product:



SCADA System - 2014 Release.zip

Components	23
Vulnerable	11
Historically vulnerable	5
No known vulnerabilities	7

Vulnerabilities	791
Critical	346
Major	412
Minor	33

Licenses	11
Copyleft	0
Permissive	8
LGPL	1
Proprietary	2

Identified 3rd party components

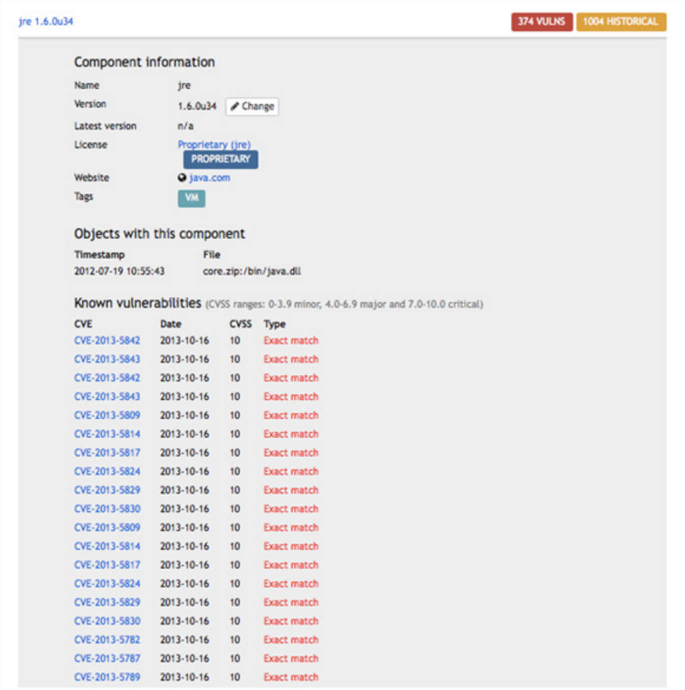
jre 1.6.0u34			374 VULNS	1004 HISTORICAL
ffmpeg	2 OBJECTS	COVERITY SCAN	173 VULNS	27 HISTORICAL
openssl		COVERITY SCAN	75 VULNS	61 HISTORICAL
openssl 1.0.0d	17 OBJECTS	COVERITY SCAN	61 VULNS	75 HISTORICAL

Of these vulnerabilities, over 300 were deemed critical, mainly through the scoring system used in the US NIST National Vulnerability Database (https://nvd.nist.gov).

The NIST NVD primarily provides the Synopsys SCA Supply Chain with the CVEs. Sponsored by the US Federal Government, MITRE Corporation (https://cve.mitre.org) compiles CVEs and articulates the details of the vulnerability and assigns a criticality score between 0 and 10. A score between 7.5 and 10 is deemed "critical" by the scoring system and means that the vulnerability is remotely executable with no authentication required.

The Java package (jre 1.6.0) within the SCADA system contained over 300 known vulnerabilities and of these vulnerabilities, over 150 scored between 7.5 and 10.

The package was further analyzed and the vulnerabilities were graphed over time, starting with the oldest known component to the most recent. The results were quite startling:



jre 1.6.0u34 374 VULNS 1004 HISTORICAL

Component information

Name	jre
Version	1.6.0u34
Latest version	n/a
License	Proprietary (jre) PROPRIETARY
Website	java.com
Tags	VM

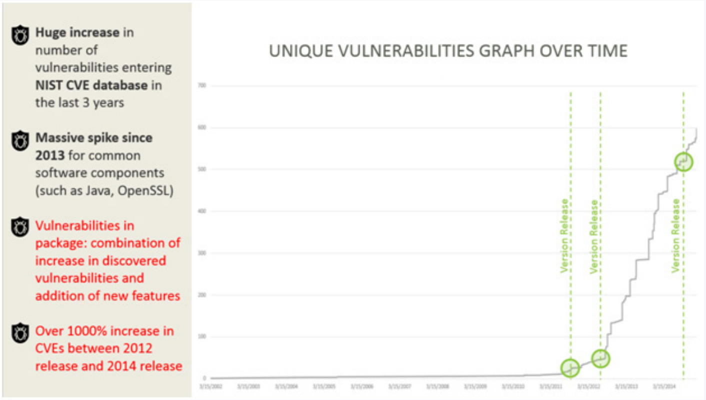
Objects with this component

Timestamp	File
2012-07-19 10:55:43	core.zip:/bin/java.dll

Known vulnerabilities (CVSS ranges: 0-3.9 minor, 4.0-6.9 major and 7.0-10.0 critical)

CVE	Date	CVSS	Type
CVE-2013-5842	2013-10-16	10	Exact match
CVE-2013-5843	2013-10-16	10	Exact match
CVE-2013-5842	2013-10-16	10	Exact match
CVE-2013-5843	2013-10-16	10	Exact match
CVE-2013-5809	2013-10-16	10	Exact match
CVE-2013-5814	2013-10-16	10	Exact match
CVE-2013-5817	2013-10-16	10	Exact match
CVE-2013-5824	2013-10-16	10	Exact match
CVE-2013-5829	2013-10-16	10	Exact match
CVE-2013-5830	2013-10-16	10	Exact match
CVE-2013-5809	2013-10-16	10	Exact match
CVE-2013-5814	2013-10-16	10	Exact match
CVE-2013-5817	2013-10-16	10	Exact match
CVE-2013-5824	2013-10-16	10	Exact match
CVE-2013-5829	2013-10-16	10	Exact match
CVE-2013-5830	2013-10-16	10	Exact match
CVE-2013-5782	2013-10-16	10	Exact match
CVE-2013-5787	2013-10-16	10	Exact match
CVE-2013-5789	2013-10-16	10	Exact match

The vertical axis indicates the number of vulnerabilities (CVEs) affecting the SCADA package and the horizontal axis represents the timeline. As shown by the graph, the number of vulnerabilities affecting the system took a massive upturn around 2012.



- **Huge increase** in number of vulnerabilities entering **NIST CVE database** in the last 3 years
- **Massive spike since 2013** for common software components (such as Java, OpenSSL)
- Vulnerabilities in package: combination of increase in discovered vulnerabilities and addition of new features
- Over 1000% increase in CVEs between 2012 release and 2014 release

The vendor website provided three release dates between 2012 and 2014. These dates correlate with the sudden sizeable increase in known vulnerabilities affecting the SCADA system, clearly indicating (by our analysis) that the use of third-party components during product enhancement drove the vulnerability count up.

Deployment and benefits realized

The software manufacturer was contacted and presented with the results of the analysis. The initial reaction from the software vendor was one of shock and when asked how this situation came to be, they informed us that the changes they had implemented in the software coincided with the increase in cybersecurity vulnerabilities. The reason they had not addressed these issues is their testing plan did not include testing to enumerate the third-party libraries contained in the software.

The SCADA system vendor addressed the discovered issues and within two months brought the known vulnerability count down to a total of 40, a significant decrease. A large number of the vulnerabilities were addressed by simply updating the Java package to the latest version. The remaining 40 vulnerabilities were the result of one of the software components within the SCADA system having internal dependencies on vulnerable open source libraries which had to be addressed by the component creator.

> Within two months the vendor brought the known vulnerability count down to a total of 40, a significant decrease.

It is important to note that once the vendor was provided with the information needed to assess the vulnerabilities in their software, they were able to act quickly to fix the issues. However, this does not necessarily mean that end users could update their SCADA systems, since many of these are critical and cannot be taken offline to make such changes. Nonetheless, the vendor was able to inform customers about vulnerabilities that affect their software, giving customers the opportunity to implement mitigating procedures while awaiting the installation of software updates and upgrades.

Conclusion

As modern society grows increasingly dependent on technology to run our critical infrastructures, security becomes a more critical aspect of control system software. Using Synopsys SCA Supply Chain throughout the development life cycle will help ICS manufacturers and end users identify and mitigate vulnerabilities from control system software, ensuring overall safety in cyber-physical processes.

Explore how to secure your control system software with the Synopsys Software Composition Analysis tool.

Learn more

THE SYNOPSYS DIFFERENCE

Synopsys offers the most comprehensive solution for integrating security and quality into your SDLC and supply chain. Whether you're well-versed in software security or just starting out, we provide the tools you need to ensure the integrity of the applications that power your business. Our holistic approach to software security combines best-in-breed products, industry-leading experts, and a broad portfolio of managed and professional services that work together to improve the accuracy of findings, speed up the delivery of results, and provide solutions for addressing unique application security challenges. We don't stop when the test is over. Our experts also provide remediation guidance, program design services, and training that empower you to build and maintain secure software.

For more information go to www.synopsys.com/software

SYNOPSYS®
185 Berry Street, Suite 6500
San Francisco, CA 94107 USA

U.S. Sales: (800) 873-8193
International Sales: +1 (415) 321-5237
Email: software-integrity-sales@synopsys.com