

# Nexus Repository를 활용한 위험한 오픈소스 라이브러리 유입 차단 및 관리

2023.09.05

| [최경철\(kchoi@opentext.com\)](mailto:kchoi@opentext.com)

# *Application Security*

오픈소스 라이브러리 이해

소나타입 플랫폼

- Lifecycle
- Nexus Repository
- Repository Firewall

# 오픈소스 라이브러리 이해

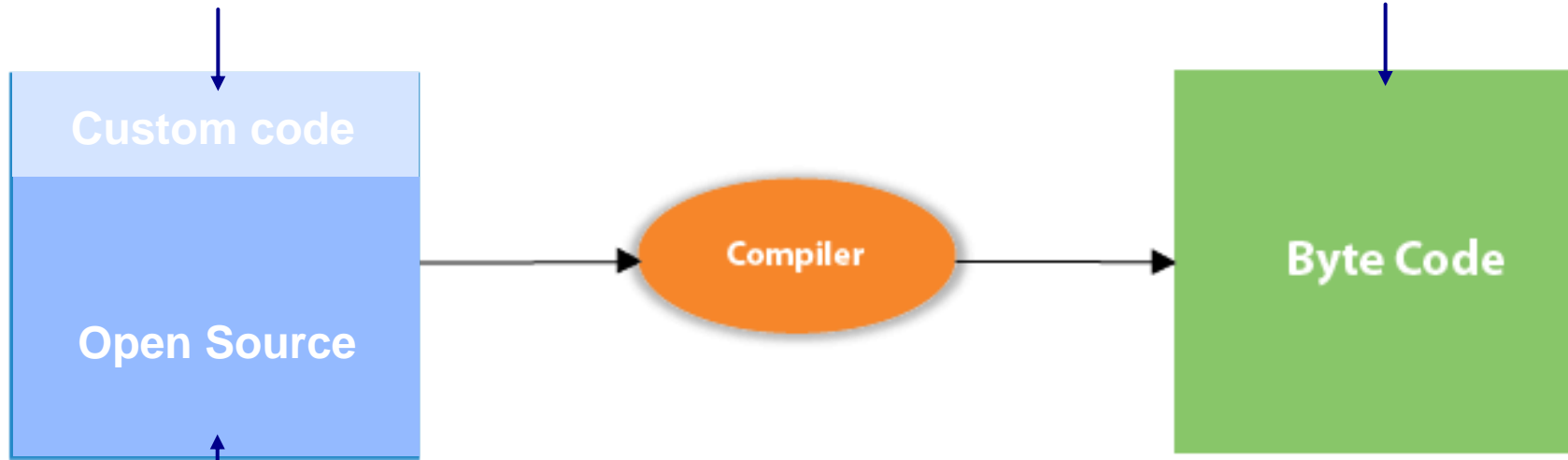
# 애플리케이션 보안 테스트 - SAST, SCA, DAST

SAST(Static Application Security Testing)

DAST(Dynamic Application Security Testing)

“소프트웨어 보안약점 점검”

“웹 취약점 점검”



SCA(Software composition analysis )

“오픈소스 라이브러리 취약점 점검”

# 패키지 매니저

- ✓ 패키지(라이브러리 등)를 관리(추가, 수정, 삭제)하는 작업을 자동화 및 관리하기 위한 도구

Language	Package Manager
PHP	Composer
Java	Maven
Ruby	RubyGems
Python	pip
Web frontend	Bower
Node.js & JavaScript	NPM

```
m pom.xml (BookStore)
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.example.maven</groupId>
  <artifactId>BookStore</artifactId>
  <packaging>pom</packaging>
  <version>1.0-SNAPSHOT</version>
  <modules...>
  <profiles>
    <profile...>
      <profile>
        <id>productionServer</id>
        <properties>
          <database.url>
            jdbc:postgresql://host/database
          </database.url>
        </properties>
        <dependencies>
          <dependency>
            <groupId>org.postgresql</groupId>
            <artifactId>postgresql</artifactId>
            <version>9.4-1206-jdbc4</version>
          </dependency>
        </dependencies>
      </profile>
```

# 패키지 매니저 - 의존성

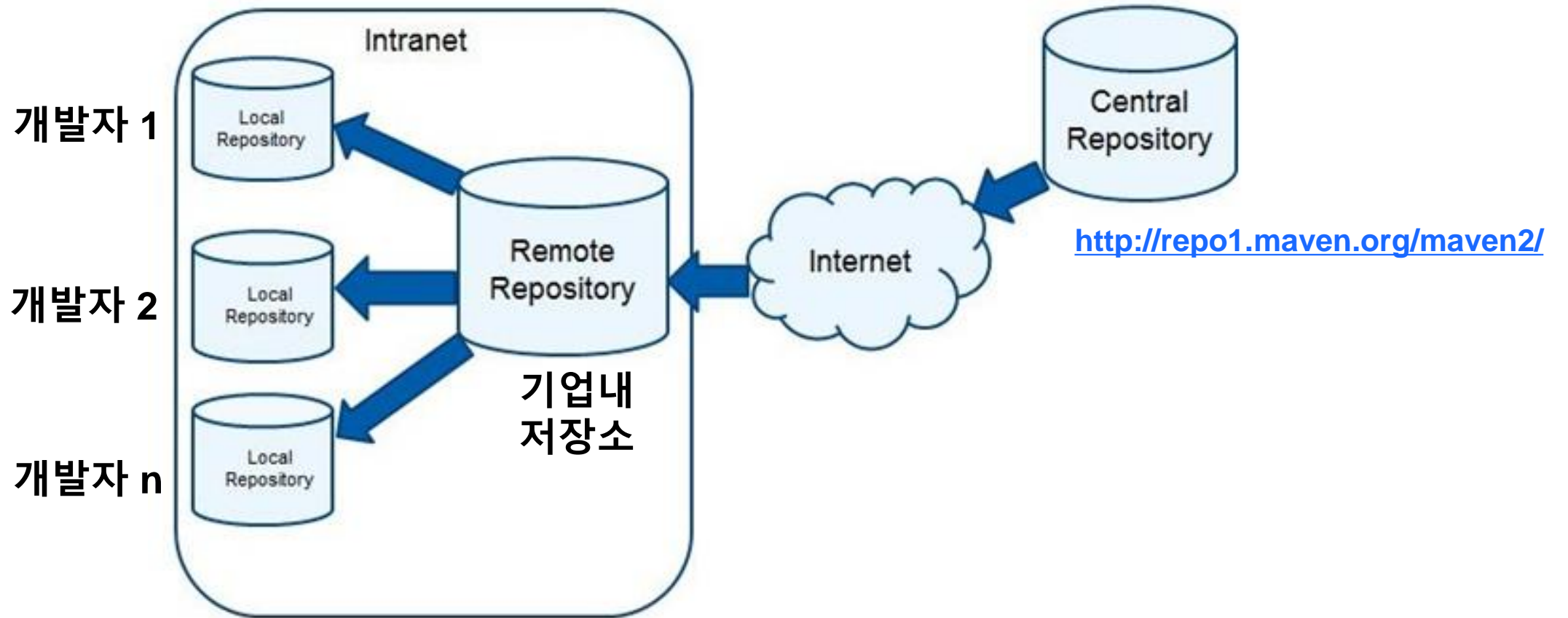
- ✓ pom.xml 에 Junit 라이브러리 지정시, **정의하지 않은 의존성 라이브러리(hamcrest-core-1.3.jar) 자동 다운로드**

```
1 <project xmlns="http://maven.apache.org/POM,
2   <modelVersion>4.0.0</modelVersion>
3   <groupId>com.codebind</groupId>
4   <artifactId>maven-demo</artifactId>
5   <version>0.0.1-SNAPSHOT</version>
6
7   <dependencies>
8   <dependency>
9     <groupId>junit</groupId>
10    <artifactId>junit</artifactId>
11    <version>4.12</version>
12  </dependency>
13 </dependencies>
14 </project>
```

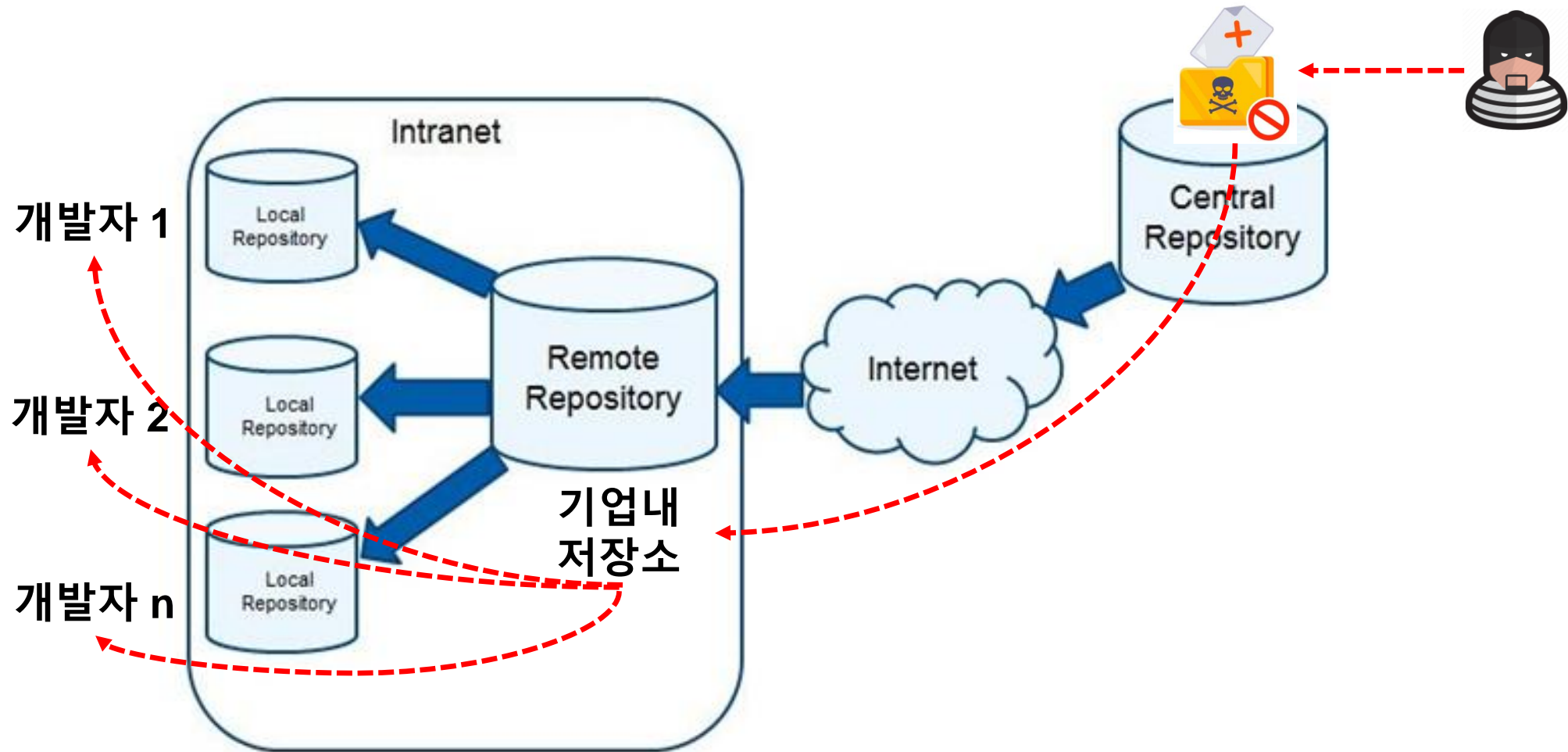
✓ Indirect dependency

✓ Direct dependency

# 기업내 오픈소스 라이브러리 관리(1)



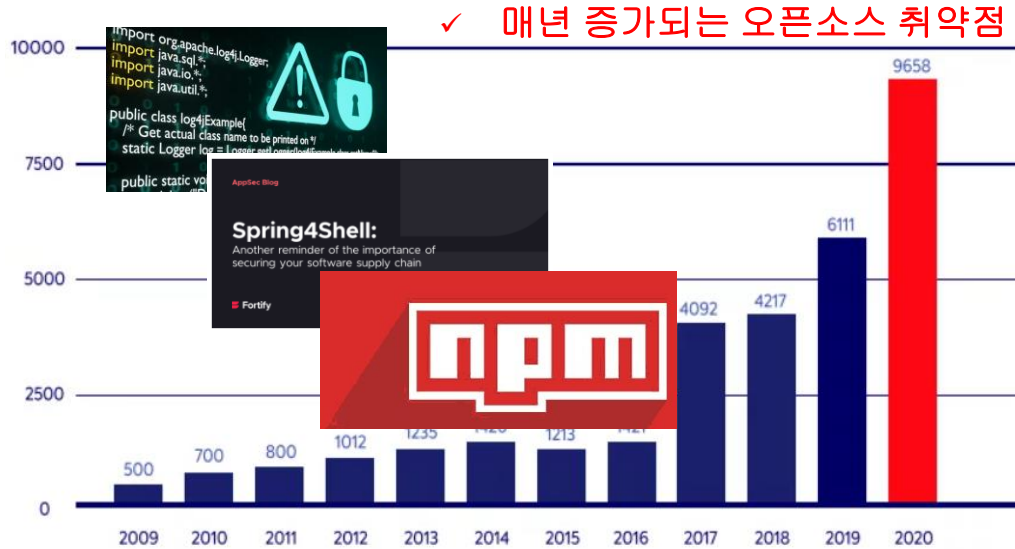
# 기업내 오픈소스 라이브러리 관리(2)





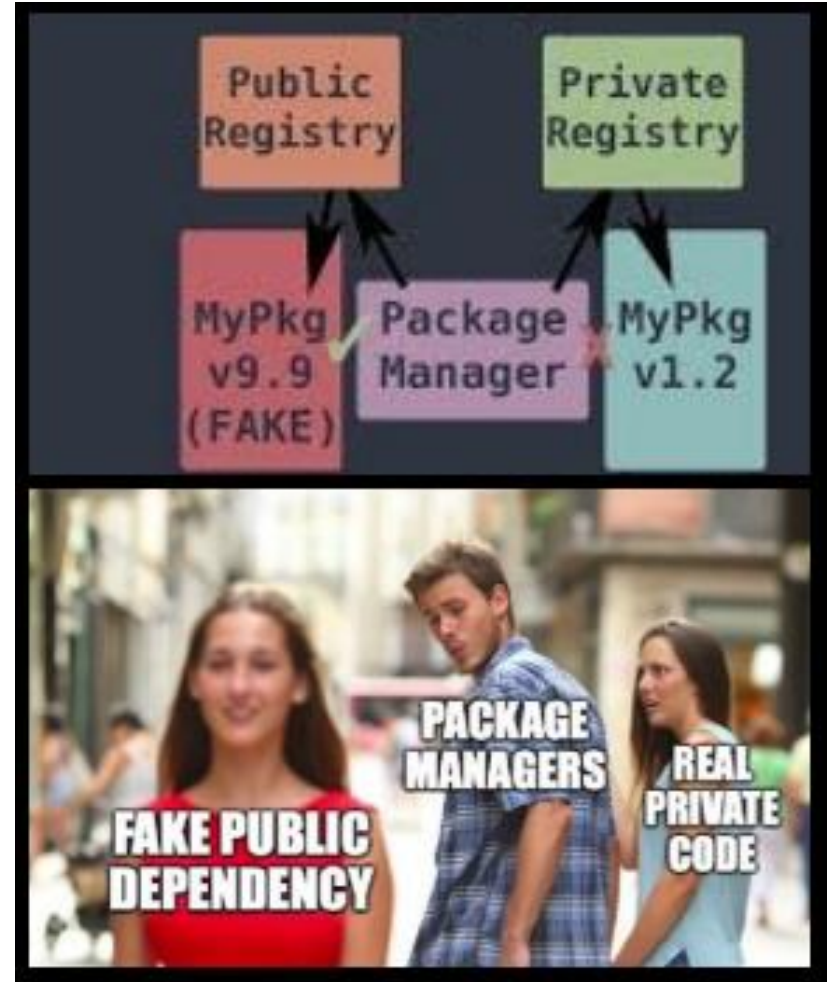
# 오픈소스 라이브러리 취약점 사례

Open Source Vulnerabilities per Year: 2009-2020



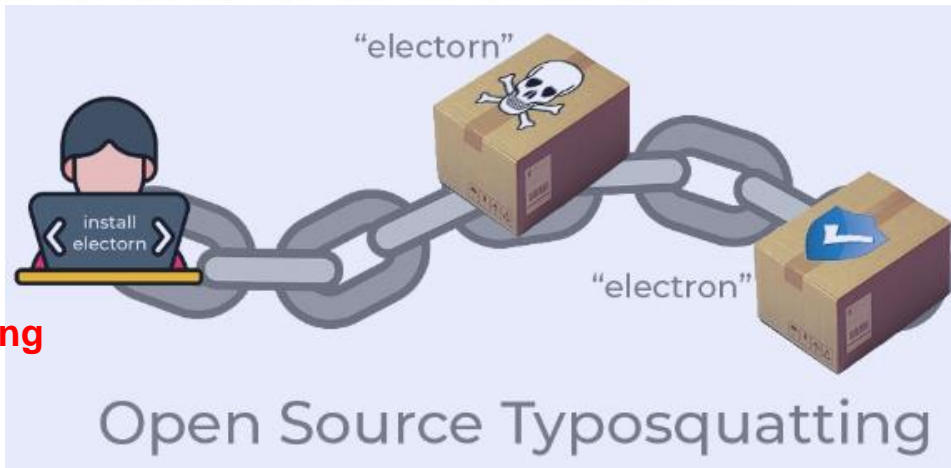
✓ 매년 증가되는 오픈소스 취약점

✓ Dependency confusion  
(의존성 혼동)



[https://openchain-project.github.io/OpenChain-KWG/meeting/12th/OpenSourceVulnerability\\_20211220.pdf](https://openchain-project.github.io/OpenChain-KWG/meeting/12th/OpenSourceVulnerability_20211220.pdf)

✓ Typesquatting  
(오타)



<https://www.activestate.com/resources/quick-reads/how-open-source-typoquatting-attacks-work/>

# SBOM(Software Bill of Materials)

✓ BOM(자재명세서)이란?



<https://www.arenasolutions.com/resources/glossary/bill-of-materials/>

# SBOM(Software Bill of Materials)

✓ SBOM이란? 취약점 식별 및 관리를 위한 식별정보

## Elements

		% Daily Value*
<b>Supplier Name</b>	The name of an entity that creates, defines, and identifies components.	%
<b>Component Name</b>	Designation assigned to a unit of software defined by the original supplier.	
<b>Version of the Component</b>	Identifies a specific version of the component.	
<b>Other Unique Identifiers</b>	Other identifiers relevant to the component.	
<b>Dependency Relationship</b>	Characterizes the relationship between components.	
<b>Author of SBOM Data</b>	The name of the entity that creates the SBOM data for this component.	
<b>Timestamp</b>	Record of the date and time of the SBOM data assembly.	%

- 제공자 이름
- 컴포넌트 이름
- 컴포넌트 버전
- 해쉬정보
- 컴포넌트 의존성
- 타임스탬프

<https://soos.io/sbom-101-what-is-an-sbom-why-are-they-important>



Administration Priorities

MAY 12, 2021

## Executive Order on Improving the Nation's Cybersecurity

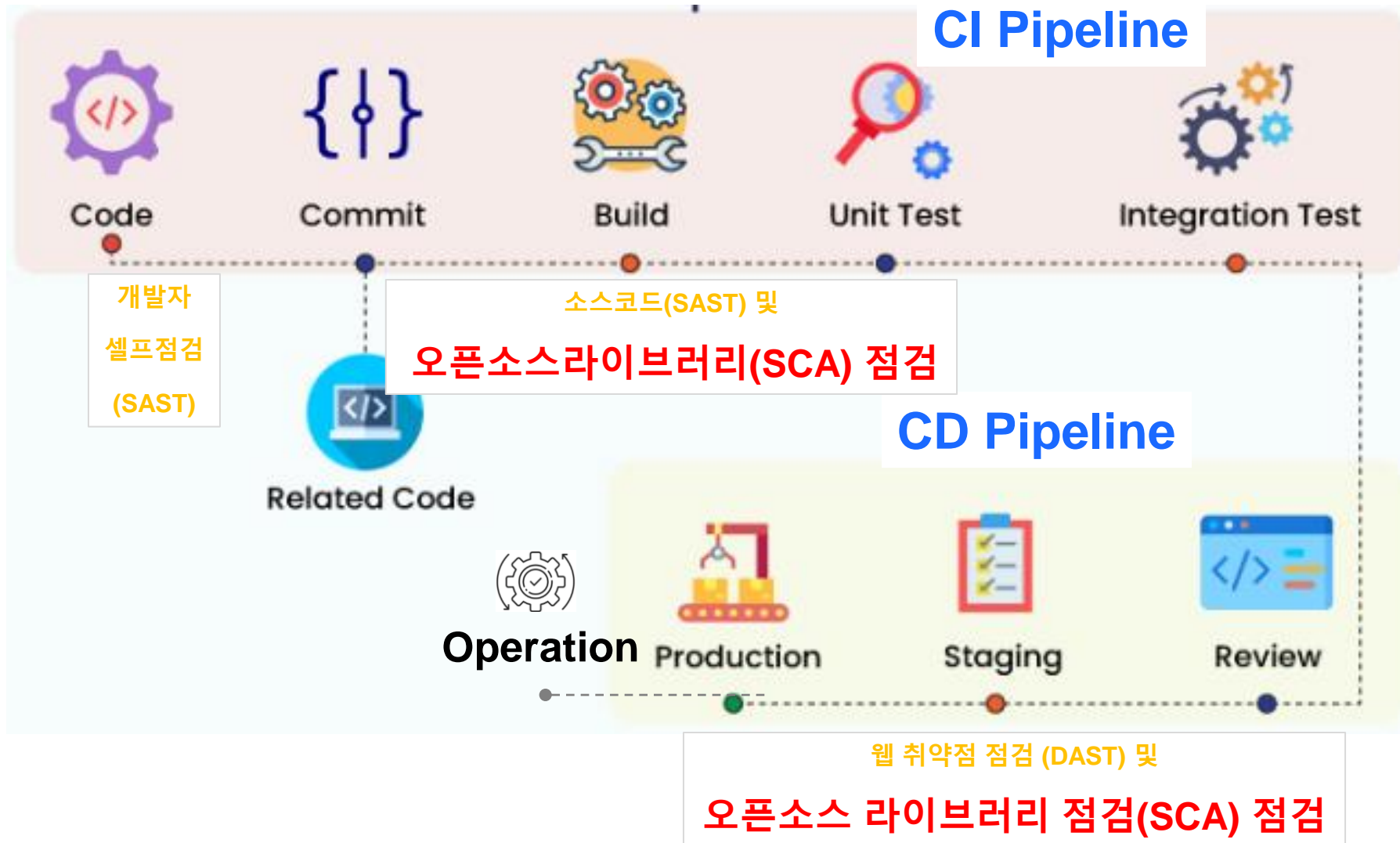
BRIEFING ROOM PRESIDENTIAL ACTIONS

By the authority vested in me as President by the Constitution and the laws of the United States of America, it is hereby ordered as follows:

**Section 1. Policy.** The United States faces persistent and increasingly sophisticated malicious cyber campaigns that threaten the public sector, the private sector, and ultimately the American people's security and privacy. The Federal Government must improve its efforts to identify, deter, protect against, detect, and respond to these actions and actors. The Federal Government must also carefully examine what occurred during any major cyber incident and apply lessons learned. But cybersecurity requires more than government action. Protecting our Nation from malicious cyber actors requires the Federal Government to partner with the private sector. The private sector must adapt to the continuously changing threat environment, ensure its products are built and operate securely, and partner with the

미국대통령 행정명령 – 연방정부 계약기업, SBOM제출의무화

# 오픈소스 라이브러리 점검방식



# Sonatype 플랫폼

# 소나타입 제품 플랫폼

-  Sonatype Repository Firewall  
Block malicious open source at the door
-  Sonatype Nexus Repository  
Build fast with centralized components
-  Sonatype Lifecycle  
Control open source risk across your SDLC



내부  
오픈소스저장소

**Nexus Repository**  
(오픈소스 라이브러리 저장소)



WebInspect

**Firewall**

**Repository Firewall**  
(위험한 라이브러리 유입 차단)



Fortify

**Lifecycle**  
(SDLC단계의 오픈소스 라이브러리 위험요소 확인 및 빌드통제)

← 인터넷 →

← 기업내부망 →

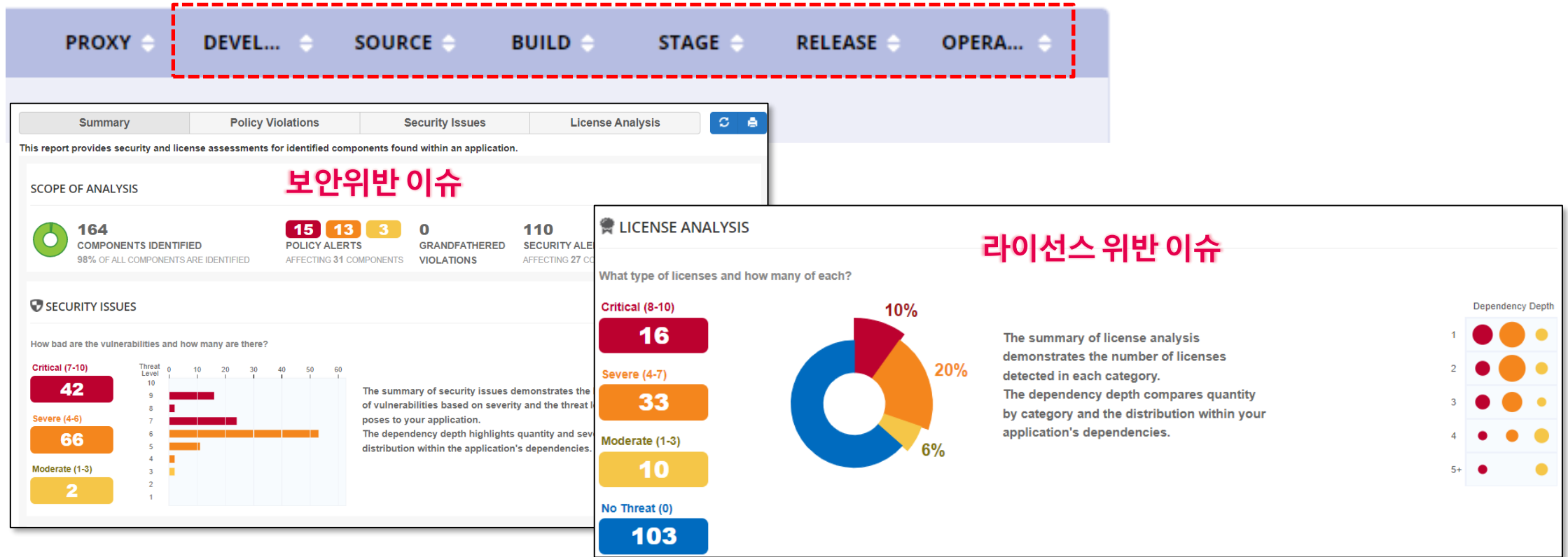
The background features several glowing blue lines of varying thickness and curvature, creating a sense of motion and depth against a solid black background. The lines are most prominent on the right side, where they appear to curve and overlap.

# 라이프사이클

( Sonatype Lifecycle)

# 소나타입 제품 플랫폼 – Lifecycle(Stage개념)

- ✓ Lifecycle은 SDLC단계에서 발생가능한 “**CVE취약점**” 및 “**라이선스위반사항**”을 탐지하고, 통제할 수 있는 수단제공



- ✓ Policy Violations 메뉴 : CVE 취약점 , Component 상태 및 노후상태정보(5년이상 오래되거나 사용인기도 낮음)
- ✓ Security 메뉴 : CVE 취약점
- ✓ Legal 메뉴 : 라이선스 유형(위반사항 포함)




# 소나타입 제품 플랫폼 – Lifecycle(Policy개념)

✓ Lifecycle은 Stages별(Proxy제외, Develop ~ Operate 단계) 정책적용

NAME	PROXY	DEVEL...	SOURCE	BUILD	STAGE	RELEASE	OPERA...
Local to Root Organization							
10 Security-Namespace Conflict	Fail	—	—	—	—	—	—
10 Security-Malicious	Fail	Fail	Fail	Fail	Fail	Fail	Fail
10 Security-Critical	—	—	—	—	—	—	—
10 License-Banned	—	—	—	—	—	—	—
9 Security-High	—	—	—	—	—	—	—
9 License-None	—	—	—	—	—	—	—
9 Integrity-Rating	Fail	—	—	—	—	—	—

✓ Stages 별 정책여부 결정(No action, Warn, Fail)  
(예) Security-Malicious 로 판단되는 라이브러리의 경우, CI/CD단계에서 Fail 처리

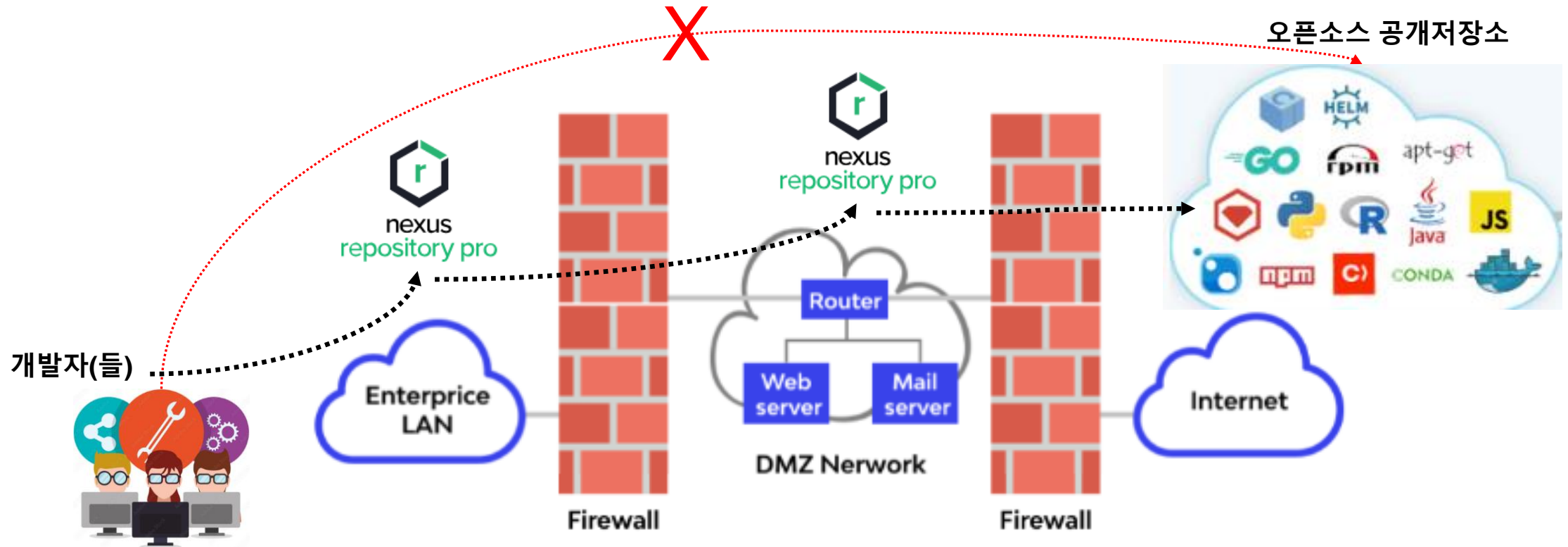


# 넥서스 리포지토리

( Sonatype Nexus Repository )

# 소나타입 제품 플랫폼 – Nexus Repository Pro

- ✓ 기업내 업무효율 및 보안을 위해 사용하는 라이브러리 Repository
- ✓ 공공/기업 80%이상이 Nexus Repository Community버전을 사용
- ✓ 인터넷 접속제한이 있는 내부망
- ✓ 공용라이브러리의 버전관리 및 팀간 공유/배포 어려움을 해소



# 소나타입 제품 플랫폼 – Nexus Repository Pro

- ✓ Repositories(Nexus Pro) – 기업내부 저장소(Repository Firewall연동시, 기업내부의 저장소 취약점 점검)

## Repositories

Configuration Policies Namespace Confusion Protection Access

### Configuration

REPOSITORY	REPOSITORY MANAGER
01-test	FD6DA94C-2E8F1C24-21A6DE92-6E0B4436-77...
02-test	FD6DA94C-2E8F1C24-21A6DE92-6E0B4436-77...
03-test	FD6DA94C-2E8F1C24-21A6DE92-6E0B4436-77...
04-test	FD6DA94C-2E8F1C24-21A6DE92-6E0B4436-77...
05-test	FD6DA94C-2E8F1C24-21A6DE92-6E0B4436-77...

## 15-test Repository Results

19 1 20 VIOLATIONS Affecting 1 component 5 COMPONENTS 100% of all components identified 0 QUARANTINED components

THREAT	POLICY	QUARANTINED	COMPONENT
	<input type="text" value="policy name"/>		<input type="text" value="component name"/>
● 10	Security-Critical		com.thoughtworks.xstream : xstream : 1.4.5
● 10	Security-Critical		com.thoughtworks.xstream : xstream : 1.4.5
● 10	Security-Critical		com.thoughtworks.xstream : xstream : 1.4.5

● Enabled

● Enabled



# 리포지토리 Firewall

( Sonatype Repository Firewall )

# 소나타입 제품 플랫폼 – Repository Firewall

## Sonatype Repository Firewall vs 보안솔루션 비교

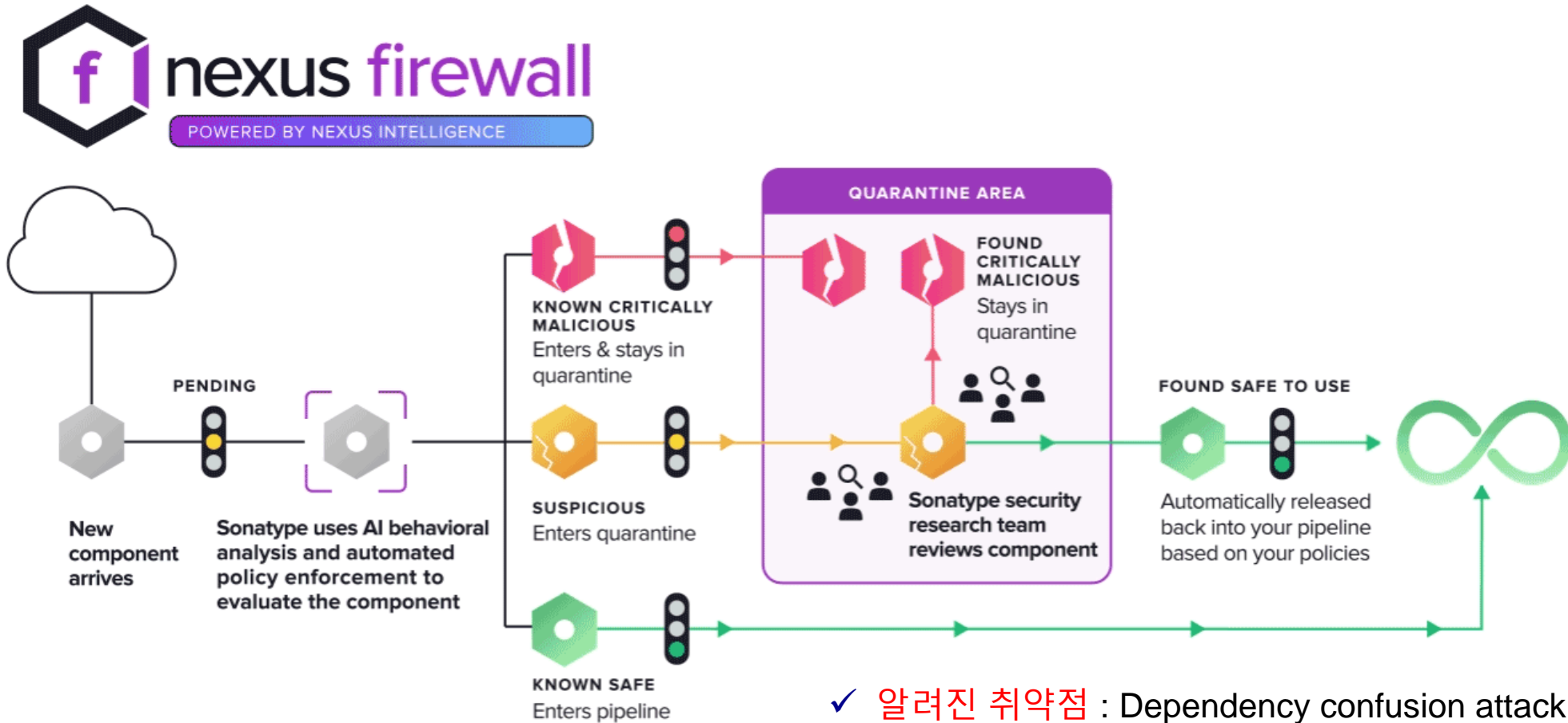
구분	주요 기능	오픈소스 라이브러리 보안기능
네트워크 방화벽	<ul style="list-style-type: none"> <li>특정포트 및 IP 차단</li> </ul>	-
IPS	<ul style="list-style-type: none"> <li>인터넷 웜, 악성코드 및 해킹 등과 같은 유해 트래픽 차단</li> </ul>	-
웹 방화벽	<ul style="list-style-type: none"> <li>80, 443포트로 유입되는 웹 해킹 패턴 차단</li> </ul>	-
<b>Repository Firewall</b>	<ul style="list-style-type: none"> <li>위험 및 의심스러운 오픈소스 라이브러리 다운로드 격리</li> </ul>	허용 혹은 격리

(예) 지극히 정상적인 라이브러리 다운로드 트래픽(공격패턴 등이 전혀없다. !!)

<https://www.test.com/repository/demo-mvn-proxy/com/thoughtworks/qdox/qdox/2.0-M8/qdox-2.0-M8.jar>

# 소나타입 제품 플랫폼 – Repository Firewall

- ✓ 라이브러리 저장소 방화벽은 외부에서 다운로드하는 위험/의심스러운 라이브러리를 격리(유입방지)



- ✓ 알려진 취약점 : Dependency confusion attack, Malware 등
- ✓ 의심스런 취약점 : Pending/Suspicious integrity rating

# 소나타입 제품 플랫폼 – Repository Firewall

✓ proxy타입의 저장소(외부인터넷과 연결된)에 보호정책제공( 악의적인 라이브러리 유입 “**격리/허용**” )

✓ Repository Firewall 정책

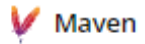
NAME	PROXY	DEVEL...	SOURCE	BUILD	STAGE	RELEASE	OPERA...
Local to Root Organization							
10 Security-Namespace Conflict	Fail	—	—	—	—	—	—
10 Security-Malicious	Fail	Fail	Fail	Fail	Fail	Fail	Fail
10 Security-Critical	—	—	—	—	—	—	—
10 License-Banned	—	—	—	—	—	—	—
9 Security-High	—	—	—	—	—	—	—
9 License-None	—	—	—	—	—	—	—
9 Integrity-Rating	Fail	—	—	—	—	—	—
8 License-Copyleft	—	—	—	—	—	—	—
7 Security-Medium	—	—	—	—	—	—	—



# 소나타입 제품 플랫폼 – Repository Firewall

- ✓ 라이브러리 저장소 방화벽은 현재 보호되고 있는 라이브러리, 격리(차단된 라이브러리)정보를 상세하게 제공

## com.thoughtworks.xstream : xstream : 1.4.5



Overview

Policy Violations

Security

Legal

Labels

### Policy Violations

THREAT	POLICY/ACTION	CONSTRAINT NAME	CONDITION
● 10	<b>custom - security critical</b> ❗ Proxy Failing	custom - security critical	Found security vulnerability CVE-2013-7285 with severity >= 9 (severity = 9.8)
● 10	<b>custom - security critical</b> ❗ Proxy Failing	custom - security critical	Found security vulnerability CVE-2021-21342 with severity >= 9 (severity = 9.1)
● 10	<b>custom - security critical</b> ❗ Proxy Failing	custom - security critical	Found security vulnerability CVE-2021-21344 with severity >= 9 (severity = 9.8)

The image features the OpenText logo in a bold, white, sans-serif font, centered horizontally. The logo is set against a dark blue background with several glowing, curved lines in a lighter blue color that sweep across the frame from the top right towards the bottom left. The lines have a soft, ethereal glow and vary in thickness, creating a sense of motion and depth.

**opentext**™