



SYNK OSS TOOL

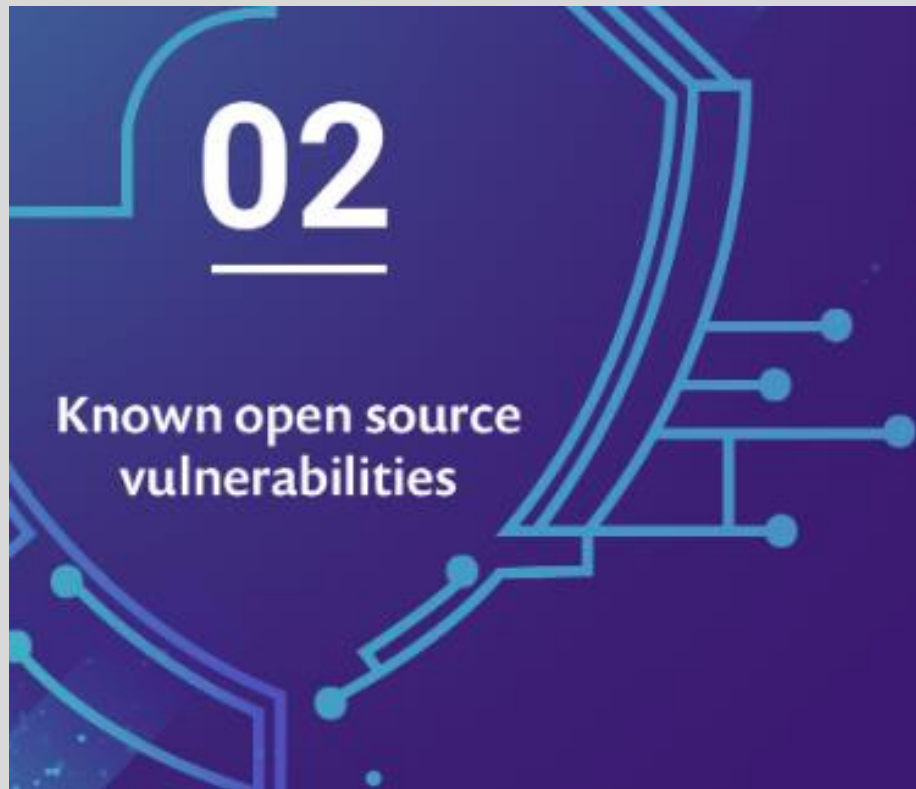
DevOps Training

@COPYRIGHT OF WWW.CLOUDBEARERS.COM

SYNK?



Known Open Source Vulnerabilities



02

Known open source vulnerabilities

A vulnerability is a vulnerability, whether known or not. The key difference between the two is the likelihood of an attacker to be aware of this vulnerability, and try to exploit it. Therefore, the better known the vulnerability is, the more urgent it is to deal with it.

A known vulnerability might have a CVE ID associated with it as part of a responsible disclosure, or it might just be disclosed on the internet or stored in open databases. These are all types of known vulnerabilities that you should prioritize eliminating as they have a higher chance of being attacked in production. After these, vulnerabilities that are captured in closed vulnerability databases or even shared in the dark web should be considered.

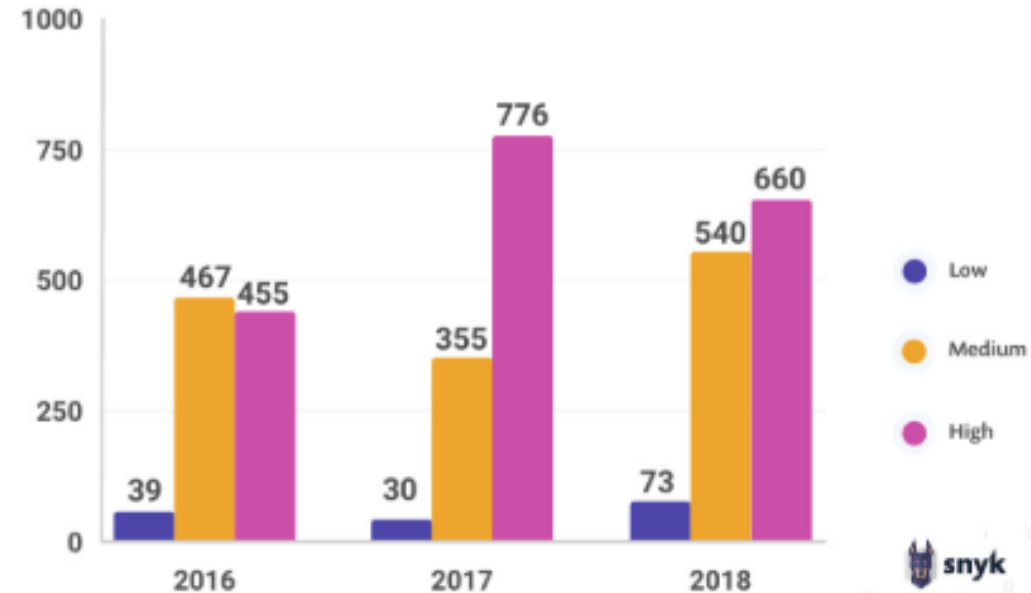
SEVERITY TRENDS

Trends in severity

When we look at vulnerability severity for application libraries disclosed over the last three years across all language ecosystems, 2018 shows a smaller number of high vulnerabilities as compared to the previous year.

However an interesting insight for both 2017 and 2018 is that there were more high severity vulnerabilities than medium or low vulnerabilities as compared to 2016.

Vulnerability severities by year



What Synk provides ?



Developer-first security

Giving developers a security tool they use and love.



Automated remediation

Powerful fix advice and automation that enables security at scale and speed.



Leading vulnerability database

Hand-curated, enriched and first to publish vulnerability content.

CVE and NVD

CVE and NVD Are Two Separate Programs

The [CVE List](#) was launched by [MITRE](#) as a community effort in 1999, and the [U.S. National Vulnerability Database \(NVD\)](#) was launched by the [National Institute of Standards and Technology \(NIST\)](#) in 2005.

- **CVE** - A [list](#) of entries—each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities. CVE Entries are used in numerous [cybersecurity products and services](#) from around the world, including NVD.
- **NVD** - A vulnerability database built upon and fully synchronized with the CVE List so that any updates to CVE appear immediately in NVD.
- **Relationship** – The CVE List *feeds* NVD, which then builds upon the information included in CVE Entries to provide enhanced information for each entry such as fix information, severity scores, and impact ratings. As part of its enhanced information, NVD also provides advanced searching features such as by OS; by vendor name, product name, and/or version number; and by vulnerability type, severity, related exploit range, and impact.

While separate, both CVE and NVD are sponsored by the [U.S. Department of Homeland Security \(DHS\) Cybersecurity and Infrastructure Security Agency \(CISA\)](#), and both are available to the public and free to use.

Language Support

Languages



Snyk for
Swift and
Objective-C



Snyk for
Ruby



Snyk for
Java



Snyk for
Scala



Snyk for
Python



Snyk for
Golang



Snyk for
.NET



Snyk for
PHP



Snyk for
Node.js (NPM
& Yarn)

Installation

CLI - Installation

Snyk's CLI helps you find and fix known vulnerabilities in your dependencies, both ad hoc and as part of your CI (Build) system.

The Snyk CLI requires you to [authenticate](#) with your account before using it. It supports Node.js, Ruby, Python, Java, Scala, Go, PHP and .NET.

Installation

Snyk is installed in one of two methods, either as an npm module, or via a Snyk created Docker container.

Before you begin, ensure:

- npm is installed on the same machine or you use our Snyk Docker deployment.
- To run Snyk on Alpine Linux, first install libstdc++. See [this doc](#) for more help.

Installation

Installation via npm

Run these commands to install it for local use:

```
npm install -g snyk
```

Once installed, you need to authenticate with your Snyk account:

```
snyk auth
```

To test your installation change directory into a folder containing a supported package manifest file (package.json, pom.xml, composer.lock, etc.) and run:

```
cd /my/project/  
snyk test
```

Synk Wizard

CLI - Wizard

Wizard

Snyk's `wizard` walks you through finding and fixing the known vulnerabilities in your project. Note that the wizard is currently **only available for Node.js projects**.

It leverages the separate `test`, `protect` and `monitor` actions, supported by an interactive workflow. To run the wizard, navigate to your project folder and run `snyk wizard` like so:

```
cd ~/projects/myproj/snyk wizard
```

DEMO

DEMO TIME

it's
Q & A
TIME!



THANK YOU!

training@laksans.com