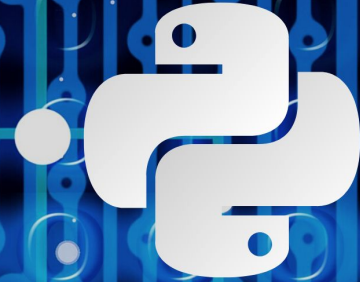


ActiveState

DevOps & SLSA

Best Practices for Software
Supply Chain Security



Introductions

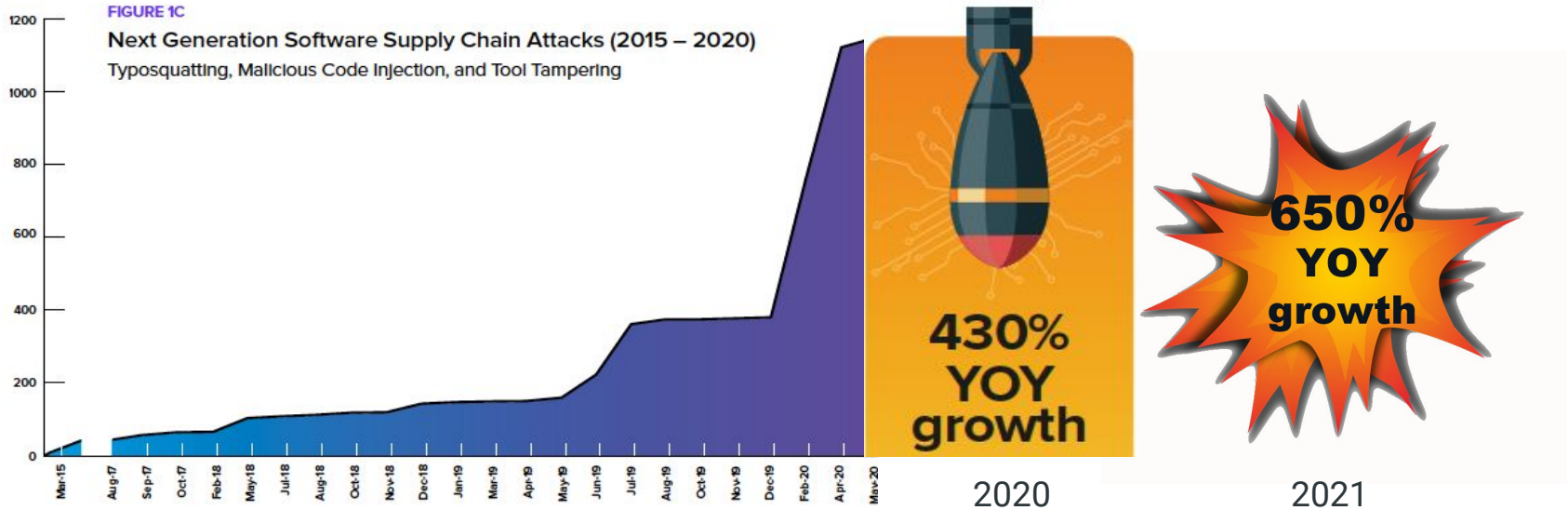


Shaun Lowry
Development Team Lead

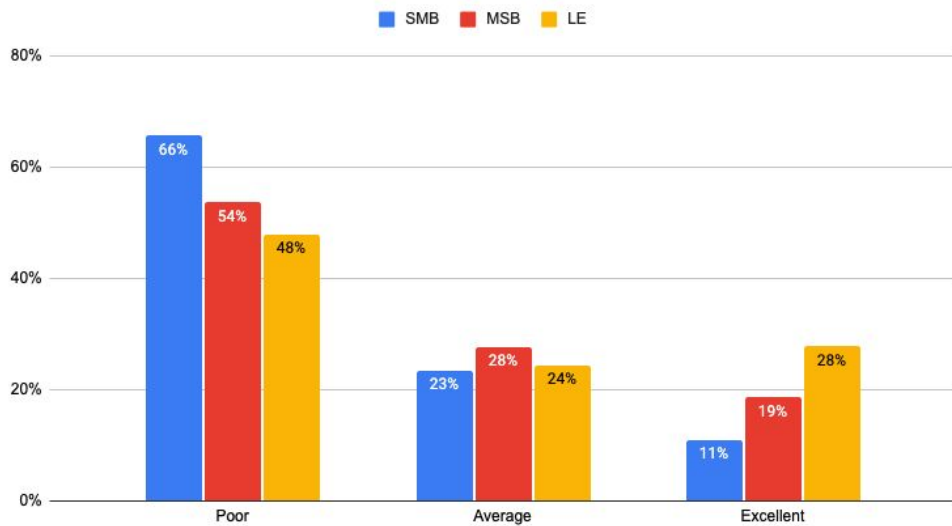


Dana Crane
Product Marketing Manager

Growing Supply Chain Threat



State of Supply Chain Security



Supply Chain Security Maturity by Org Size

Address “Import” security issues like:

- Typosquatting
- Dependency confusion
- Author impersonation

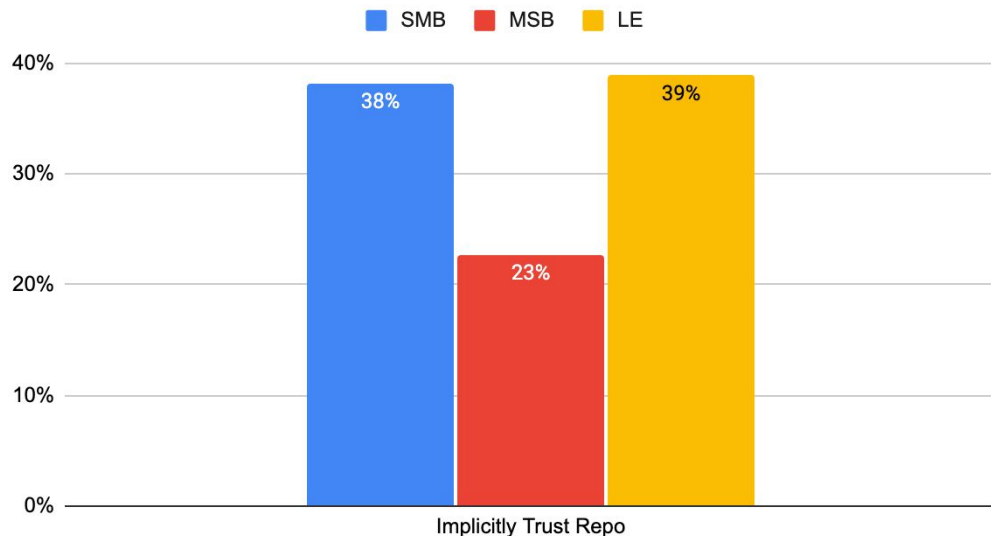
Address “Build” security issues like:

- Malicious build/install scripts
- Dynamic packages that include remote resources

Address “Consumption” security issues, like:

- Using signed and verified packages

State of Supply Chain Security

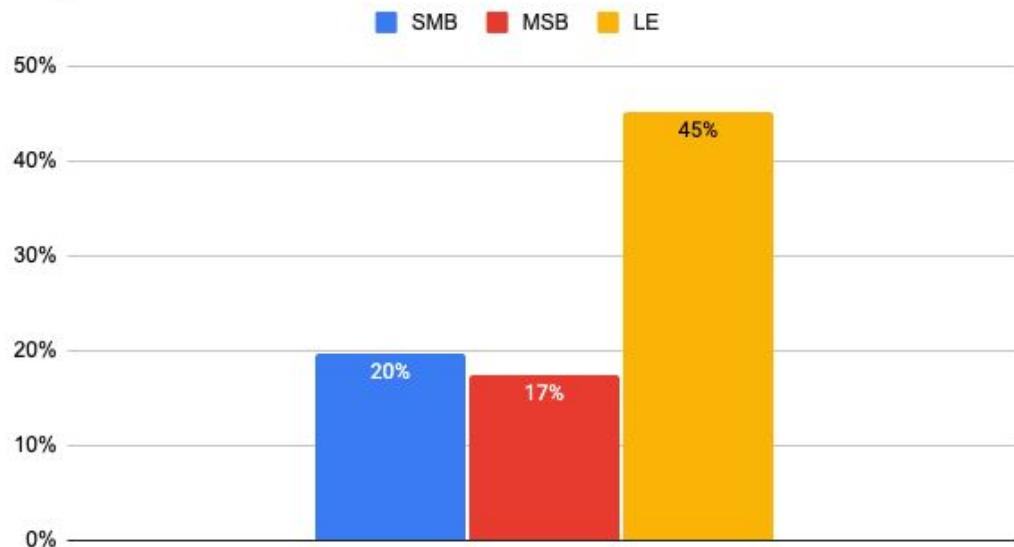


Public Repository Trust by Org Size

This is despite the fact that:

- Public repos contain hundreds of thousands of packages created by tens of thousands of authors and maintainers, all of whom must be trusted.
- Public repos contain pre-built, but unsigned packages.

State of Supply Chain Security



Reproducible Builds by Org Size

The issue is twofold:

- Open source software is typically built as a one-off task on a per-project basis for the operating system(s) used by the team.
- Open source dependencies, once added to the codebase, are rarely updated/maintained*.

*Veracode State of Software Security v12

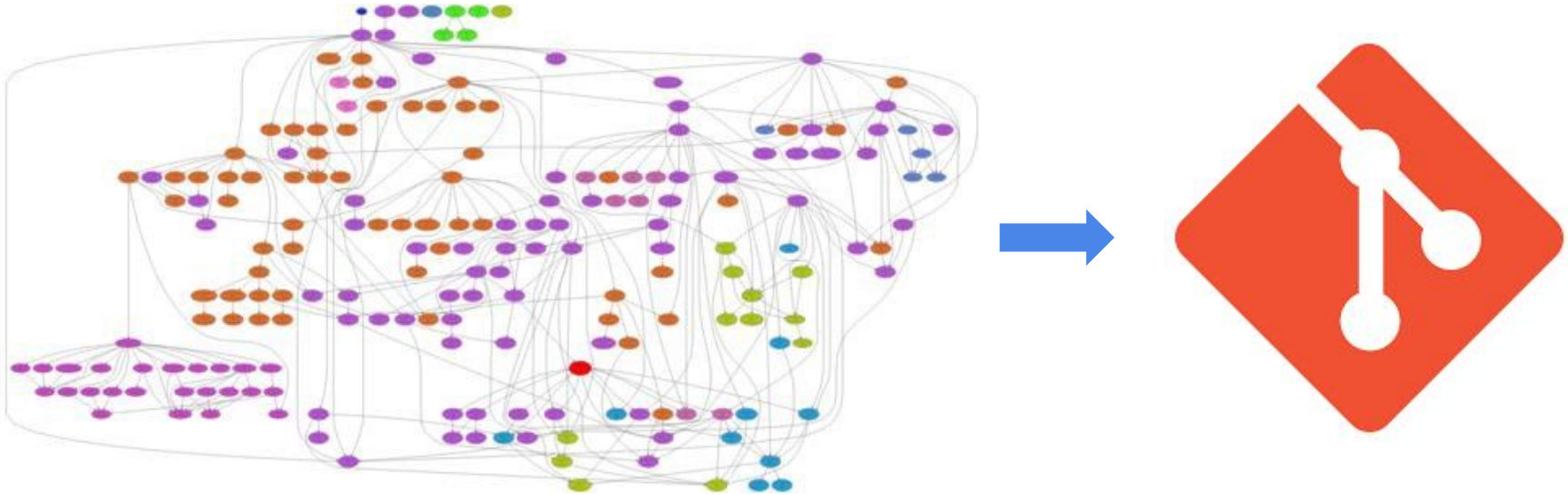
ActiveState

Poll: How would you rate your supply chain security?

- **Poor** - we import pre-built dependencies and implicitly trust the vendor/ public repository.
- **Average** - we build dependencies from source code, but our build system is not explicitly secured/ designed to create reproducible builds.
- **Excellent** - we build everything from source in a reproducible way, and frequently audit our build infrastructure for security holes.

ActiveState

Dependency Vending



Dependency Vending is a Dependency Management strategy that involves checking all your dependencies into your Code Repository

Pros and Cons of Self-Vendoring

Pros:

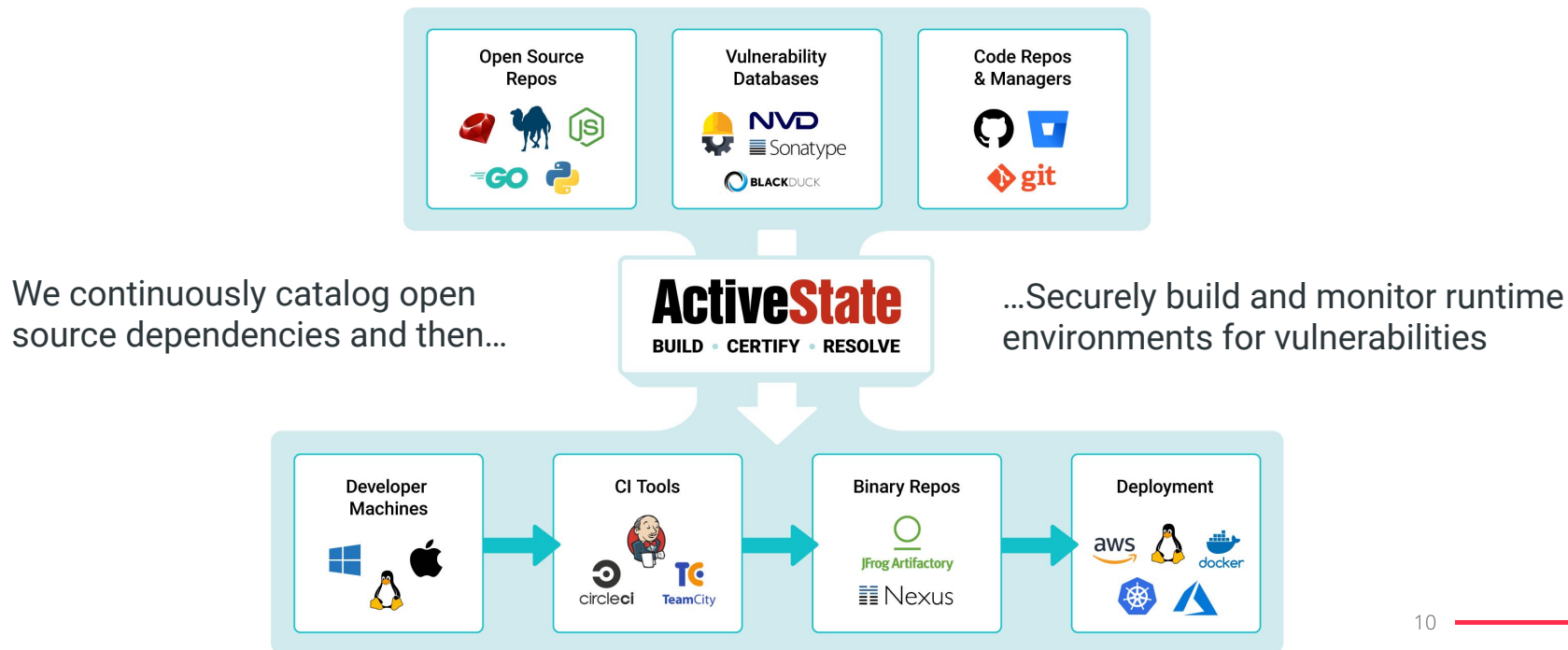
- Avoid Dependency Conflicts
- Avoid Breaking the Build
- Consistent Deployments
- Faster Fixes

Cons:

- Need to Build Everything Yourself
- Outdated Dependencies
- Invisible Vulnerabilities
- Source Code/Repository Clutter

ActiveState

ActiveState Platform



Dependency Vendors Comparison

	Prebuilt?	Secure?	Up to Date?	Time & Resources
Public Repo	Yes	No	Yes	Low
Trusted Vendor	Yes	Yes	No	Low
Build It Yourself	No	Yes	Yes	High
ActiveState Platform	No	Yes	Yes	Low

SLSA & Secure Build Service

SLSA - Addressing Supply Chain Security

- **SLSA** - Supply Chain Levels for Software Artifacts (<https://slsa.dev/>).
- **OSSF Initiative** - Operates under OSSF umbrella
- **Industry Backed** - Google, ChainGuard, Linux Foundation
- **Multiple Levels** - Levels 0 through 4 providing increasing assurance.

Artifacts and Attestations

- **Artifact** - Any digital asset which forms part of a software supply chain. Source code, build scripts, installable binaries.
- **Attestation** - A statement about the provenance of an Artifact. Who created it, how, when and with what.
- **In-toto ITE-6** - The recommended format for attestations.
- **Non-Falsifiable** - SLSA levels 3 and above

ActiveState

Secure Build Service

- Tamper-proof system creates reproducible builds of secure artifacts

Build	Scripted	✓
	Build Service	✓
	Ephemeral Environment	✓
	Isolated	✓
	Parameterless	✓
	Hermetic	✓
	Reproducible	✓

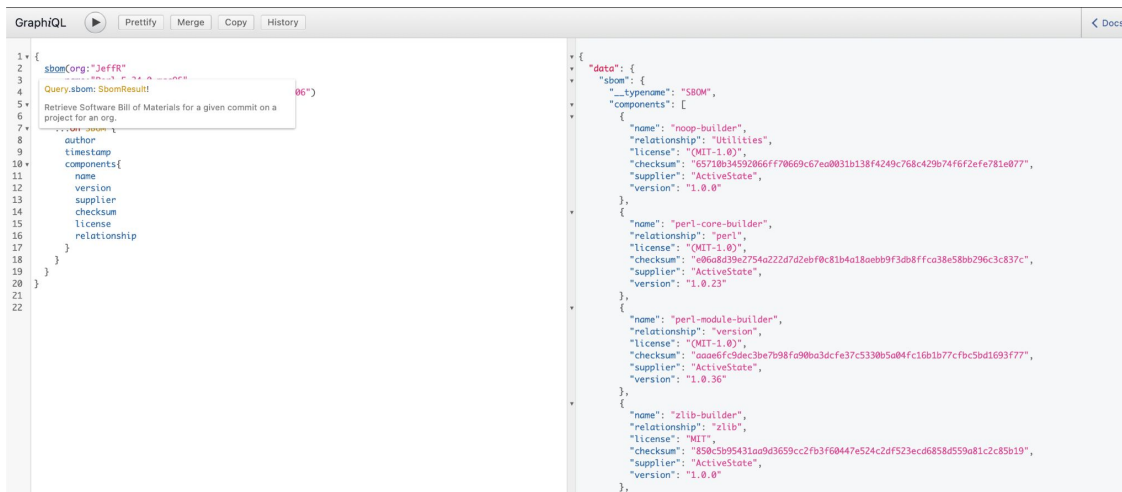
ActiveState

Poll: How SLSA compliant is your build system?

- Uses only **Scripted Builds**
- Is a **Dedicated Build Service**
- Employs **Ephemeral, Isolated Build Steps**
- Creates only **Parameterless Builds**
- Employs **Hermetically Sealed Environments**
- Generates **Reproducible Builds**

Automatically Generate SBOMs

- Software Bill Of Materials (SBOM) for each of your runtime environments



The screenshot shows the GraphQL Playground interface. On the left, a query is defined to retrieve SBOM data for a specific commit. On the right, the resulting JSON data is displayed, showing a list of software components with their names, relationships, licenses, checksums, suppliers, and versions.

```
1 {
2   sbom(org: "JeffFR"
3     ...
4     Query: sbom: SBOMResult!
5     ...
6     Retrieve Software Bill of Materials for a given commit on a
7     project for an org.
8   ) {
9     author
10    timestamp
11    components {
12      name
13      version
14      supplier
15      checksum
16      license
17      relationship
18    }
19  }
20 }
21
22
```

```
{
  "data": {
    "sbom": {
      "__typename": "SBOM",
      "components": [
        {
          "name": "noop-builder",
          "relationship": "Utilities",
          "license": "MIT-1.0",
          "checksum": "65710b34592866ff70669c67ea0031b138f4249c768c429b74f6f2e7e781e077",
          "supplier": "ActiveState",
          "version": "1.0.0"
        },
        {
          "name": "perl-core-builder",
          "relationship": "perl",
          "license": "MIT-1.0",
          "checksum": "e96a8f99e2754a222d7d2ebf0c81b4a18aebb9f3db8ffca38e58b296c3c837c",
          "supplier": "ActiveState",
          "version": "1.0.23"
        },
        {
          "name": "perl-module-builder",
          "relationship": "version",
          "license": "MIT-1.0",
          "checksum": "aaadfc94c3be7b98fa90ba3dcfe37c5330b5a04fc16b1b7c7fbc5bd1e93f77",
          "supplier": "ActiveState",
          "version": "1.0.36"
        },
        {
          "name": "zlib-builder",
          "relationship": "zlib",
          "license": "MIT",
          "checksum": "890c5b95431a9d3659cc2fb3f60447e524c2df523ecd6858d59a81c2c85b19",
          "supplier": "ActiveState",
          "version": "1.0.0"
        }
      ]
    }
  }
}
```

Automatically Generate SLSA Attestations

- Every build step logged and signed

```
{
  "type": "https://in-toto.io/Statement/v0.1",
  "invocation": {
    "configSource": {
      "digest": {
        "sha256": "938ba2b070a89358811646677115d4785ed091c71211f75876d72fe6b38f"
      },
      "entryPoint": "build",
      "url": "s3://platform-sources/shared/938ba2b070a89358811646677115d4785ed091c71211f75876d72fe6b38f/openssl-1.1.1b.tar.gz"
    },
    "environment": {
      "env": {}
    },
    "parameters": {}
  },
  "materials": [
    {
      "digest": {
        "sha256": "805f1e306e8798fc86031c3215a9096d221e9f30b3311fa0658d11e6c6f2"
      },
      "url": "asinage-docker://docker-registry.activestate.build/activestate/centos-8-builder:2.0.13"
    }
  ],
  "predicate": {
    "buildConfig": {
      "steps": [
        {
          "command": "build",
          "parameters": {}
        }
      ]
    },
    "buildType": "https://activestate.com/platform_builder/v0.1",
    "builder": {
      "id": "https://activestate.com/builder/openssl-builder1.0.8q"
    },
    "metadata": {
      "buildFinishedOn": "2022-07-05T23:03:25.896701Z",
      "buildInvocationId": "Builder openssl-builder 1.0.8 building shared openssl 1.1.1b.0.15 for artifact 3577feef-b2c-9015-abc9-86cc93a5e69",
      "buildStartedOn": "2022-07-05T23:03:56.106701Z",
      "completeness": {
        "environment": true,
        "materials": true,
        "parameters": true
      }
    },
    "reproducible": true
  },
  "predicateType": "https://slsa.dev/provenance/v0.2",
  "subject": {
    "digest": {
      "sha256": "804d34eac6239f8a71ca764c763c28e621b6a58a5c827ecde8e99fe4d"
    },
    "url": "s3://as-builds/p8856/shared/openssl/1.1.1b.0.15/2/3577feef-b2c-9015-abc9-86cc93a5e69/artifact.tar.gz"
  }
}
```

Cost-Effective Supply Chain Security

- Produce a catalog of dependencies
- Securely built
- Reproducible runtime environments across all operating systems
- SBOMs and verifiable artifact attestations
- Can be integrated into existing build systems to enhance dependency security

Platform Demo

Demo: Create a Runtime Environment

You have unsaved changes. Save your changes to update your project.

Importing packages from Python project file 43/100 [Save](#) [Cancel](#)

Language [Change Version](#)

Python 3.9.12
● 1 CVE >

Platforms [Change](#)

Linux Glibc 2.28 ⓘ
x86 · 64-bit

Mac ⓘ
x86 · 64-bit

Windows 10 ⓘ
x86 · 64-bit

Vulnerabilities ● 2 Critical ● 1 High

Requested Packages [Add Packages](#) [Import from File](#)

Python 3 Packages [Vulnerabilities \(CVEs\)](#) Licenses

+ flask	Auto (2.1.2) ▾	● 0 CVEs	Cancel
+ numpy	Auto (1.22.1) ▾	● 0 CVEs	Cancel
+ pillow	Auto (9.1.0) ▾	● 0 CVEs	Cancel

Dependencies 38
Automatically added to support requested packages & platforms.

26 Changes Only show changes

ActiveState

Q&A

Next Steps

Schedule a demo with our product experts:

<https://www.activestate.com/get-demo/>

Learn more about Supply Chain Security:

<https://www.activestate.com/solutions/slsa>

Try the ActiveState Platform for free:

<https://platform.activestate.com/>

ActiveState

Webinar Feedback

Take our quick survey!

<https://www.surveymonkey.com/r/devops-slsa-webinar>