2024 IN OPEN SOURCE MALWARE

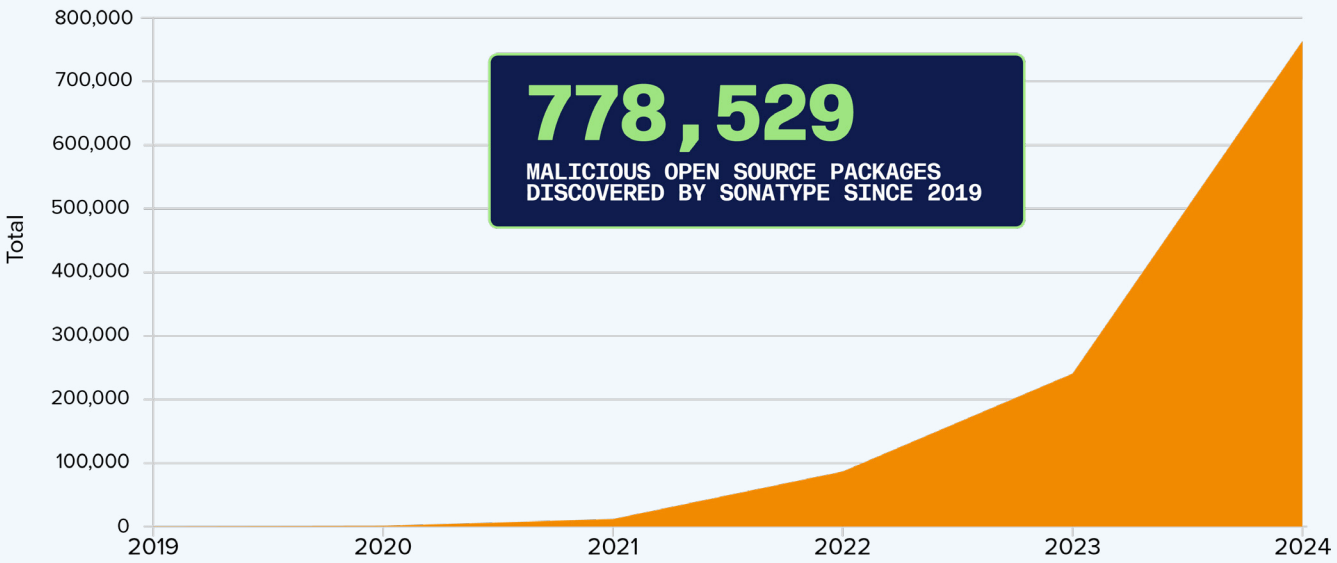
2024 in Open Source Malware

The proliferation of open source malware, or malicious open source packages, continues to accelerate, posing unprecedented risks in the form of software supply chain attacks. Unlike vulnerabilities, which are accidental coding errors, **open source malware is intentionally crafted to infiltrate and exploit software supply chains.** In fact, Sonatype estimates* 50% of unprotected repositories already have cached open source malware.

In the 2024 State of the Software Supply Chain, Sonatype reported a staggering 156% year-over-year increase in malicious packages identified over the past year. To date, **Sonatype has identified 778,529 pieces of open source malware** since it started tracking in 2019, an increase of over 70,000 since Sonatype's annual report was published in October.

FIGURE 1

Open Source Malware Over the Years



*Estimated from anonymous analysis of 100k + binary repositories between Jan-May 2024

What is Open Source Malware?

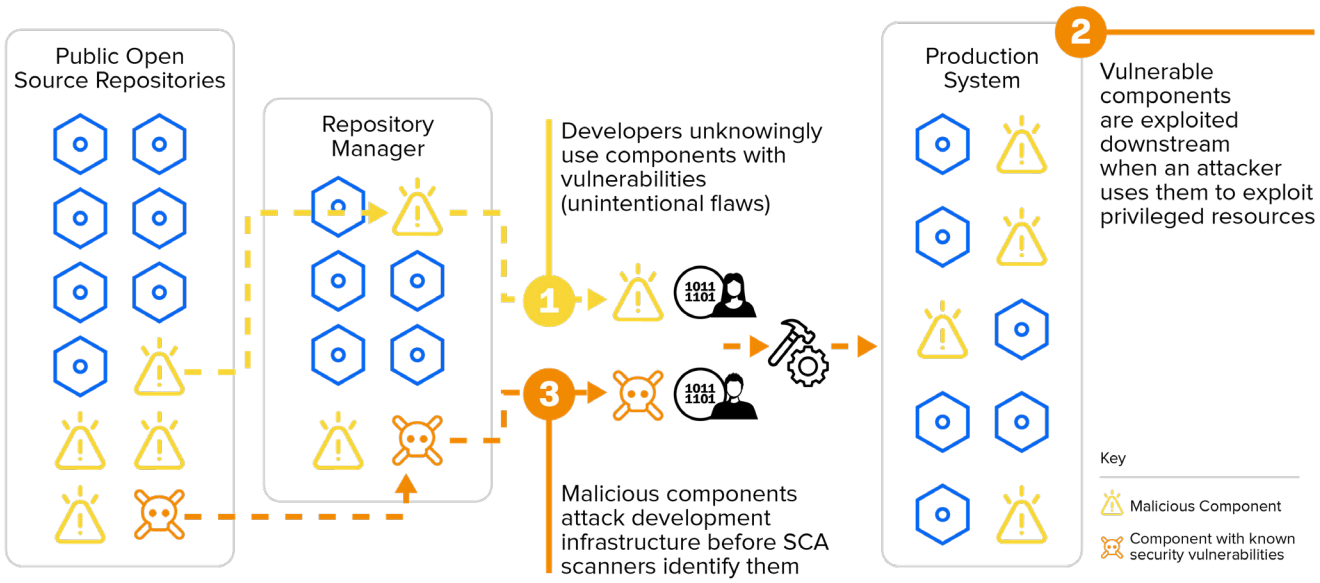
While traditional malware often spreads through email attachments, malicious websites, or compromised devices, open source malware disguises itself as legitimate open source software (OSS) components to infiltrate the storage locations of code and other development assets. This unique distribution method — compromised open source repositories — exploits gaps in dependency management tooling and development build pipelines, bypassing conventional security mechanisms in order to attack software developers directly.

THREE DISTINCT CHARACTERISTICS OF OPEN SOURCE MALWARE

INTENTIONAL INSERTION
Open source malware is purposefully uploaded to open source repositories for malicious purposes.

TARGETING DEVELOPERS
It is designed to trick developers into running the malicious component during dependency install.

DIFFICULT TO DETECT
Today's endpoint security tools do not detect these types of signatures.

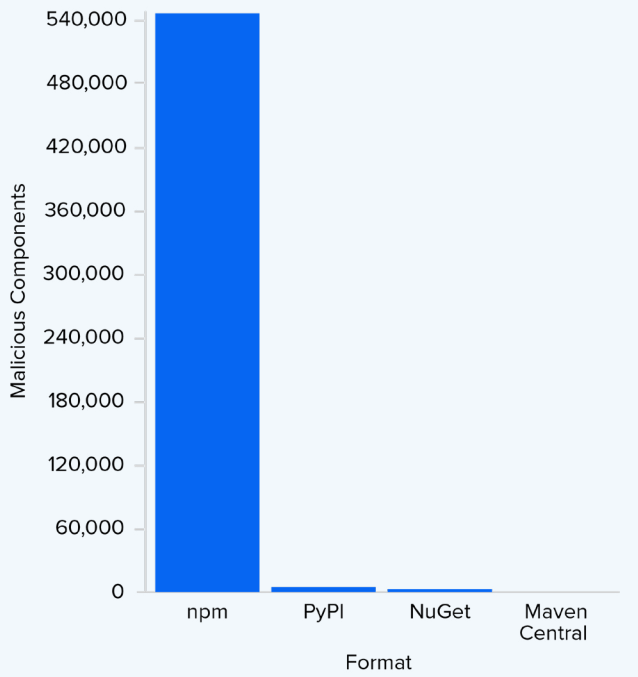


How does open source malware get in?

Open source malware thrives in ecosystems with low barriers to entry, no author identity verification, high consumption rates, and diverse user bases. As ecosystems like npm and PyPI process trillions of open source package requests annually, they create fertile ground for malicious actors seeking to infiltrate the software supply chain. Popular packages within these ecosystems are targeted, mimicked, modified, and repackaged to include malware, sometimes through maintainer account takeovers. Attackers will often mimic legitimate package names and publish higher versions to trick build systems into installing malicious versions. These CI/CD pipelines then become unwitting consumers of malicious code when tainted components are introduced.

FIGURE 2

Open Source Malware by Format



Low barriers of entry to ecosystems

The world's largest registry for JavaScript packages, npm, exemplifies the risk contained in public repositories, representing 98.5% of the malicious packages Sonatype has identified in the past year. Its open publishing model allows anyone to create and distribute packages, whether they are good-faith contributors or malicious actors. This accessibility, combined with minimal vetting processes, makes npm particularly vulnerable.

- **Ease of publishing:** Developers can publish packages with minimal verification, enabling attackers to upload malicious components quickly and at scale.
- **Spam and volume:** In recent years, npm has faced surges of spam packages — some exploiting protocols like Tea.xyz to monetize downloads, others embedding malware for more insidious purposes.
- **High demand:** With over 4.5 trillion requests expected in 2024 — a 70% year-over-year growth — npm's scale makes it an attractive target for attackers seeking maximum impact.

Beyond npm, other ecosystems like PyPI, which represented just over 1% of the open source malware Sonatype observed, also face challenges from dependency confusion, typosquatting, and spam. PyPI's rapid growth, driven by the rise of AI and cloud applications, has introduced complexities in managing both legitimate and malicious packages. As with npm, its open publishing model is a double-edged sword: fostering innovation while increasing the attack surface for malicious actors.

Shadow downloads bypassing repository managers

There is a large number of malicious packages that are bypassing repository managers and are being downloaded directly to a developer's machine or shared build infrastructure. **Shadow downloads are third-party or open source components retrieved from a public repository in a way that bypasses an artifact repository manager.** This practice introduces unvetted and unobservable dependencies into projects, bypassing established governance, review, and security processes.

With millions of open source packages downloaded daily, a substantial portion bypass repository managers, which means open source malware has an easy way into development. While precise numbers vary by organization, recent insights indicate a surprising percentage in production environments originated from shadow downloads, escaping security review entirely.

The below chart looks at the known downloads retrieved from public repositories in a way that bypasses caching repository protections to enter software development. In November, the number of shadow downloads reached more than 63 billion, a 32.8% increase over the same month of last year. Each of those components is bypassing security checks prior to application vulnerability scanning — this is important because malware is designed to trigger on download.

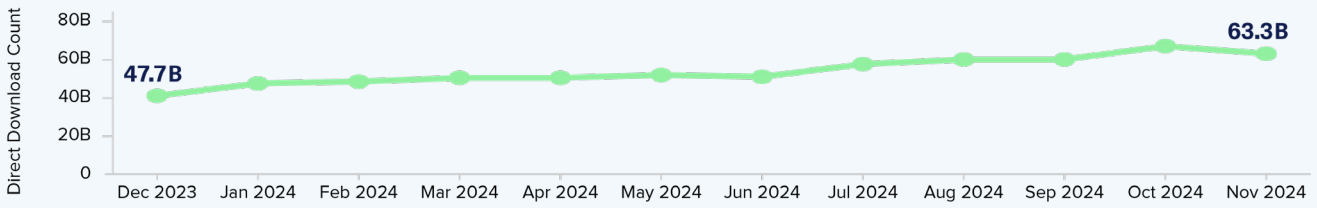
Shadow downloads undermine software supply chain security in several ways:

- **Lack of visibility:** Dependencies introduced via shadow downloads often go unnoticed, making it difficult to manage updates.
- **Increased malware risk:** Direct downloads from public repositories expose systems to unvetted components, increasing the likelihood of introducing malicious packages, such as those associated with dependency confusion or typosquatting.
- **Governance gaps:** By bypassing repository managers, organizations lose the ability to enforce policies, such as release integrity checks or vulnerability scans.

Without centralized oversight, malicious components can and will evade traditional defenses and embed themselves into the software development lifecycle (SDLC), where traditional endpoint security products will not catch them and before application scanning will detect them. This represents a massive gap in enterprise security, and attackers can easily exploit gaps in governance to deliver malicious payloads. By the time a shadow-downloaded package is identified as malicious, it has already infiltrated the SDLC.

FIGURE 3

Growth in Shadow Downloads



Source: Maven Central

Open Source Malware in 2024

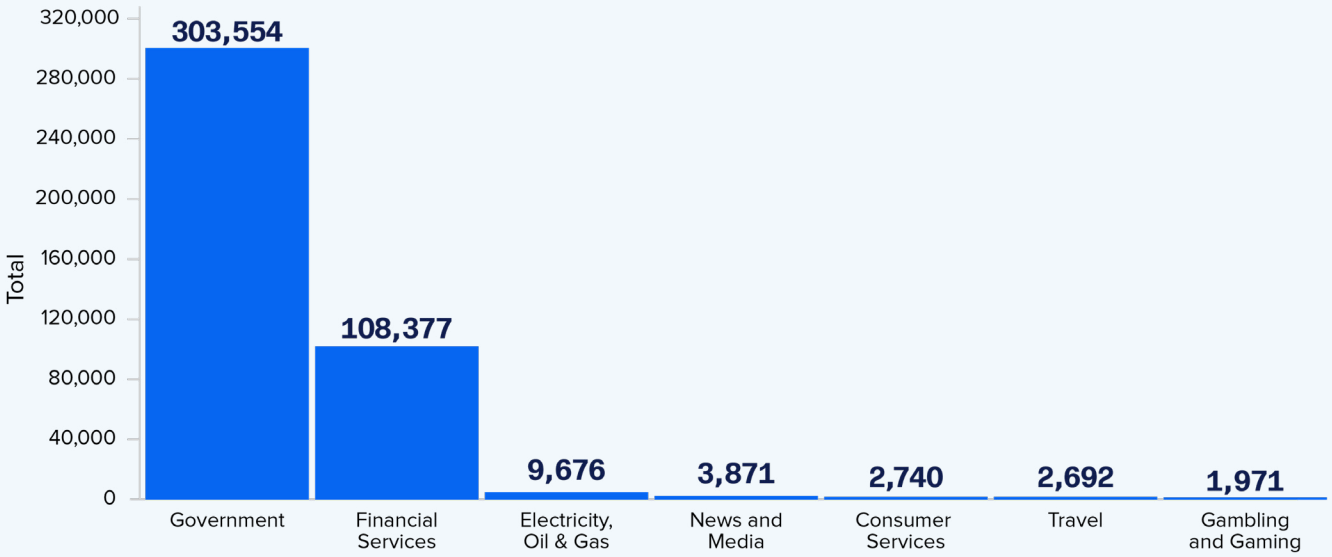
Open source malware manifests in various forms, each targeting specific vulnerabilities within software supply chains. The scale and creativity of open source malware attacks reveal the sophistication of modern threat actors.

In 2024, open source malware has increasingly targeted a wide range of industries (see Figure 4). Government organizations saw the highest number of attempted malware attacks, making up 67.31% of the total malware attacks blocked by Sonatype this year. Other heavily impacted industries include financial services (24.03%) and electricity, oil and gas (2.15%).

Potentially Unwanted Applications (PUAs) represent the bulk of open source malware activity (see Figure 5), nearly half of the total open source malware uncovered (64.75%). This category includes components that incorporate unintended functionalities, such as protestware or data collection. They pose significant risks by bypassing proper vetting processes and introducing unpredictability into software systems. While a PUA may not seem outright malicious, they could contain spyware, adware, or tracking components that would compromise the security and privacy of end users.

FIGURE 4

Malware Attacks Blocked by Industry



Security holdings packages (24.21%) are components that have been flagged by ecosystem maintainers, often for malicious activity or to lock down End of Life (EOL) components, and replaced with a clean placeholder package to draw attention to consumers. Some repositories, including npm, do not disclose reasoning behind using security holdings packages. Although they cannot be consumed, if your organization has already pulled one, it should be removed.

Data exfiltration malware (7.86%) is designed to harvest sensitive information from infected systems. Examples include extracting environment variables, personally identifiable information (PII), authentication tokens, and API keys, which are then sent to external servers for exploitation.

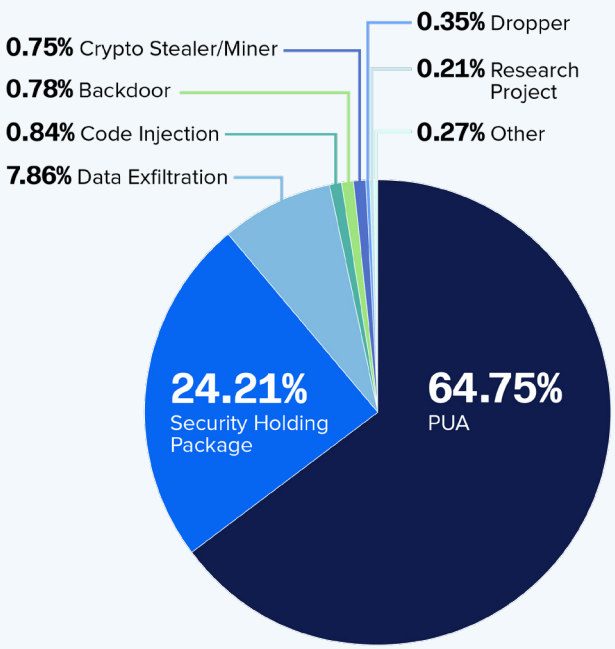
Code injection malware (.84%) contains harmful code that, if consumed, allows threat actors to execute unauthorized commands or gain access to sensitive data within an application.

Open source malware designed to introduce backdoors (.78%) install hidden access points that allow attackers to wait for the opportune moment to exploit that will have the most impact. This is similar to the impacts of the software supply chain attack against **SolarWinds' Orion software**, which allowed nation-state actors access to federal agency networks.

Crypto stealers (.75%) extract cryptocurrency or hijack computational resources for unauthorized mining. While crypto stealers represent just a small portion of the malware Sonatype has observed, we saw with **Lottie Player**, for example, that just one piece of open source malware can lead to $723,000 in cryptocurrency lost to threat actors.

FIGURE 5

Open Source Malware By Type



Notable Malicious Packages in 2024

APRIL 16

Tea.yaml

Threat actors exploited the “Tea” protocol by flooding npm and PyPI with Potentially Unwanted Applications (PUAs). These malicious packages overwhelmed repositories, undermining trust and usability.

MAY 29

Pytoileur

A malicious package, pytoileur, was identified on PyPI concealing code that installed trojanized Windows binaries. These binaries enabled surveillance, established persistence on compromised systems, and facilitated cryptocurrency theft.

JUNE 3

LUMMA

Threat actors leveraged the LUMMA malware using namespace confusion, packaging it as open source packages targeting developer environments.

JULY 26

Travis.yml

Researchers uncovered malware embedded in a travis.yml attempting to be passed off as a sample CI/CD build configuration file. The malware deployed a malicious macOS binary disguised as “Safari updates,” leveraging trusted build systems to distribute its payload.

AUGUST 7

Solana-Py

The solana-py package on PyPI exemplified a sophisticated typosquatting attack. Borrowing code from the legitimate project, this malicious package covertly extracted user secrets, making it an effective tool for targeted credential theft.

OCTOBER 31

Lottie Player

The JavaScript library Lottie Player was compromised. Attackers released three malicious versions, leading to significant financial losses, including one reported phishing incident resulting in over $723,000 stolen from a user.

Strengthen Defenses Against Open Source Malware

The rise of open source malware represents a fundamental challenge to the integrity of modern software development. As dependency chains grow increasingly complex and repositories process trillions of requests annually, the threat landscape will continue to evolve.

Predictions for 2025 suggest a sharp escalation in AI-driven open source malware attacks, proliferation of crypto-stealers, and an expanding focus on unmaintained open source projects. The attempted XZ Utils takeover and backdoor discovered earlier this year is, in all likelihood, just the tip of the iceberg with additional sophisticated campaigns awaiting discovery. The value of open source as an attack vector has been realized, so there is an urgent need to increase protections.

Improving ecosystem governance

Ensuring proper governance of dependencies at the ecosystem level is an important step to reducing exposure to harmful components. Ecosystem managers can do this by requiring packages be digitally signed with cryptographic signatures that verify the packages' authenticity and source.

Implementing vetting processes for contributors that requires a thorough onboarding process can help reduce the risk of malicious submissions. On an ongoing basis, dependency monitoring is crucial to ensure compromised components are identified and removed, ultimately reducing exposure for downstream users.

Stopping open source malware before development

The key to mitigating the threat of open source malware lies in blocking it before it enters the development environment. Allowing malicious components to infiltrate build pipelines dramatically increases the difficulty of detection and removal. By this stage, malware is already embedded in the SDLC, often bypassing traditional security measures.

To counter this, organizations must block malicious packages before they enter the repository and eliminate shadow downloads — the use of unauthorized or unvetted components outside the scope of managed repositories.

Methodology

Sonatype examined a broad set of open source package consumption data and proprietary data, including shadow downloads, which are downloaded directly from package managers and bypass repository manager protections, malicious packages blocked by Sonatype Firewall, dependency update patterns for more than 1.5 trillion requests from Maven Central and thousands of open source projects, and the assessment of hundreds of thousands of enterprise applications. The report also analyzed malicious packages observed in the Java (Maven Central), JavaScript (npm), Python (PyPI), and .NET (NuGet) ecosystems.

About Sonatype

Sonatype is the software supply chain security company. We provide the world's best end-to-end software supply chain security solution, by combining the only proactive protection against malicious open source, the only enterprise grade SBOM management and the leading open source dependency management platform. This empowers enterprises to create and maintain secure, quality, and innovative software at scale. As founders of Nexus Repository and stewards of Maven Central, the world's largest repository of Java open-source software, we are software pioneers and our open source expertise is unmatched. We empower innovation with an unparalleled commitment to build faster, safer software and harness AI and data intelligence to mitigate risk, maximize efficiencies, and drive powerful software development. More than 2,000 organizations, including 70% of the Fortune 100 and 15 million software developers, rely on Sonatype to optimize their software supply chains. To learn more about Sonatype, please visit www.sonatype.com.