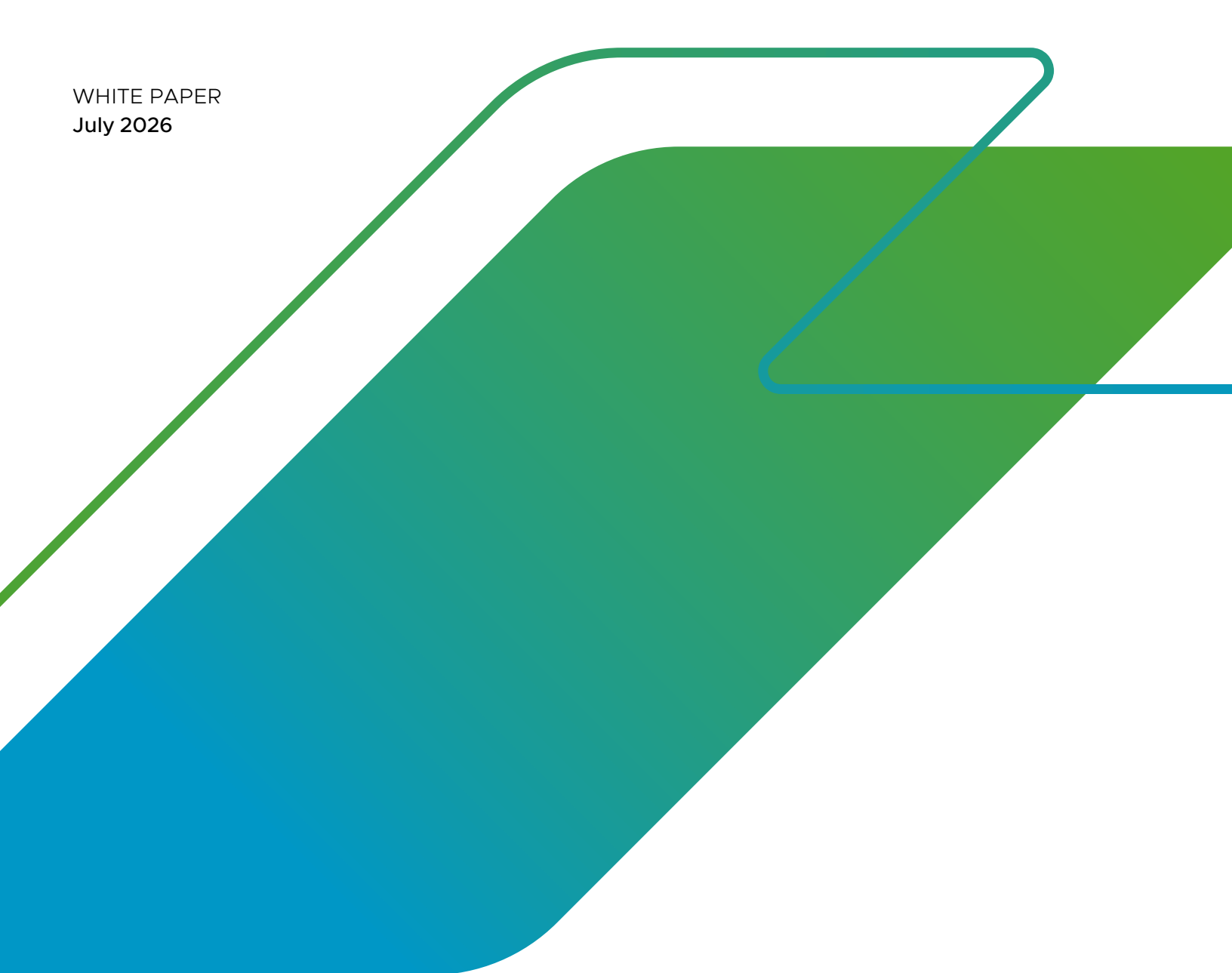


WHITE PAPER
July 2026

An abstract graphic consisting of several overlapping, rounded rectangular shapes in shades of green and blue, creating a layered, geometric effect on the left side of the page.

The Fallacy of Third-Party Support

The Case for First-Party Enterprise
Support in an Era of Accelerating Threats

Table of contents

Table of contents 2

Executive summary 3

The high stakes of unsecured software 4

Options for enterprise support 5

 First-party support 5

 Third-party support 5

Risks of third-party support. 6

 Inability to validate vulnerability fixes 6

 Reactive vs. proactive 6

 Missing architectural knowledge 6

 Risk of side effects 7

 Compliance and audit 7

 Limited scope and total cost of ownership 7

 Limited interaction with the core team 7

Advantages of first-party support 8

Conclusion 8

Executive summary

Enterprises today stand at the foot of a mountain of technical debt and needed software upgrades. As the software industry ships more versions faster and faster, and with more bugs and vulnerabilities, followed by subsequent patches to address them all, the typical enterprise just cannot keep up.

However, the risks of not climbing that mountain are just as daunting. The number of new vulnerabilities reported and the average cost of a security breach have reached record highs. Meanwhile, the time it takes to exploit a vulnerability has dropped to a record low. These trends mean the risks of not patching software are growing—all while the gap in organizational preparedness compared to accelerating threats is not showing signs of closing. On top of that, AI-generated code is increasing the volume of software, and organizations will be facing an exponential increase in their mountain of tech debt tomorrow.

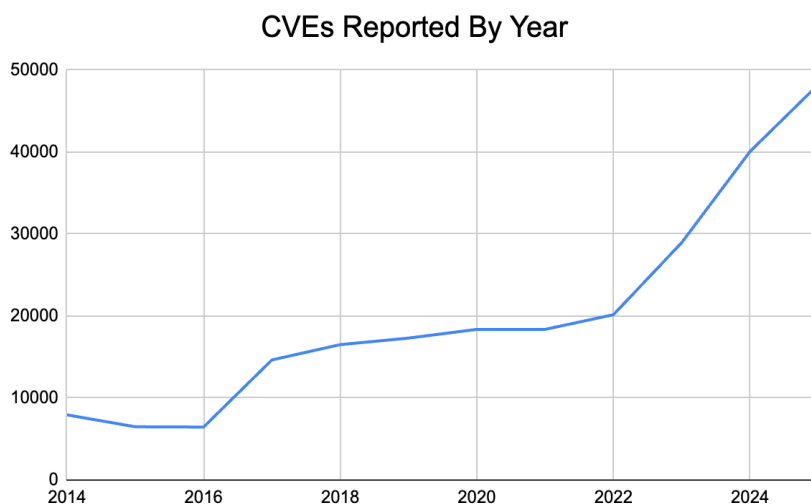
Open source software adds an additional layer of complexity to that equation. Instead of obtaining terms and conditions on how patching and support timelines work up front as part of a purchase from a trusted company, the open source software that enterprises rely on may be maintained by just a single person (such as the [XZ](#) project) with little to no commitment to maintenance service levels.

Working with a vendor that does not have a direct connection to the project they claim to support carries a clear risk profile. Given that only a tiny fraction of vulnerabilities have a public proof of concept, a vendor without access to the original security reporter is limited in their ability to validate the fix.

This document highlights critical challenges that modern enterprises face when choosing open source support vendors and describes why [Tanzu Spring](#), [Broadcom's first-party support](#) offering, provides the most comprehensive enterprise support option. [Tanzu Spring](#) is the only first-party enterprise support plan that covers the entire critical Java stack: Spring, Apache Tomcat, and the JDK. Organizations that rely on Tanzu Spring can channel their energy into innovating for their businesses, instead of draining their energy on risk management and remediation.

The high stakes of unsecured software

The production environments where enterprise software runs are increasingly hostile places. MITRE reported that 2025 was a record year for reported CVEs, totaling over 48,000 new vulnerabilities reported.¹ This represented a year-over-year increase of over 20% in reported CVEs, with no signs of slowing down.



As the number of reported vulnerabilities continues to rise, the cost of a breach is following suit. IBM's [Cost of a Data Breach Report 2025](#)² showed that, while the global cost of a data breach fell in 2025 to \$4.4 million, the average cost of a data breach in the US hit a new record high at \$10.22 million, representing a more than 9% increase over the previous year. When broken down by industry, healthcare and finance had the largest data breach costs, with healthcare having the highest growth in costs (over 31% increase in cost per breach YoY).

The risks of a data breach to an enterprise are rising from both increased opportunities for attackers and to an enterprise's cost per breach. However, the threat of a breach is not the sole reason for enterprises to obtain enterprise support. Compliance regulations—such as PCI-DSS, HIPAA, DORA, FIPS, and others—all require access to patches and regular application of those patches to software, with the potential for severe penalties and business loss for non-compliance. First-party software support is preferable to third-party support because only [first-party support like that available with Tanzu Spring](#) can guarantee that patches are considered “vendor supplied” for audit purposes. First-party support ensures immediate compliance and removes the need for internal supply chain teams to manually vet third-party fixes.

Most large organizations have decades' worth of applications and systems in place. Those are the apps that helped them achieve their longevity and current success. Figures vary, but anecdotally, executives estimate that they spend

¹ <https://www.cve.org/about/Metrics>

² “[Cost of a Data Breach Report 2025](#),” IBM.

anywhere between 50% to 80% of enterprise IT budget on “maintenance” that deals with old software. Old software can be prone to security issues as much as newer software can. Sometimes, these older applications can require even more attention because newer software often has more attention from the business.

No matter the size of your application portfolio, you still need to address the fundamental question: “How do we keep our enterprise safe?”

Options for enterprise support

While access to patches and upgrades has always been a critical part of enterprise maintenance, the inconsistent nature of where to get patches—specifically for open source offerings—has made it much more difficult than one would expect. Today, there are two options for enterprise support: first-party support and third-party support.

First-party support

First-party support is delivered by the entity that develops (or acquired) the open source project and/or employs its core committers. The same organization that writes the code, defines the roadmap, and releases the official patches is the one providing the support.

[Tanzu Spring](#) is the industry’s leading example of first-party enterprise support, because the Tanzu team is the same organization that maintains the Spring portfolio as well as employs the top committers for Apache Tomcat.

First-party support providers have unique advantages because of where they sit in the software lifecycle. They have direct access to design intent, historical context, and unreleased fixes. Owning the original code base allows first-party support teams like the Tanzu Spring team to drive the roadmaps and support cycles. Because first-party support providers have access to the security researchers and organizations that report vulnerabilities, they can be proactive in releasing patches. Customers who use first-party support typically are covered for both legacy applications and new development.

Third-party support

Third-party enterprise support is delivered by an independent vendor that has no ongoing development role in the upstream project; expertise is built through use, community participation, and analysis rather than through writing the code itself. Third-party support does not employ core committers or control official roadmaps or releases.

Third-party support vendors typically provide patches or extended support for older, end-of-life (EOL) versions. However, they do not typically provide support for up-to-date versions. One of the more negative effects of this is that the patches provided by third-party vendors may or may not be merged into the upstream project. This can lead to divergence from the official project over time, which can, in turn, lead to difficulties in upgrading to future official releases. Divergent code may also fail to meet certification standards because the code

has “forked” from the official project source code.

While typically characterized as “neutral” or “independent” in the sense that they are not the original author or steward of the project, third-party support vendors are also distinct from resellers or system integrators, who may offer first-party support via contracts with the original upstream vendor.

Risks of third-party support

Third-party support offers extended periods for receiving software patches at a price point that can be attractive. However, there are risks associated with this approach.

Inability to validate vulnerability fixes

The largest risk with applying third-party fixes is that organizations have no proof that the vulnerability was actually fixed. According to a [Tenable report](#), only 2.6% of vulnerabilities reported between January 1, 2024, and September 20, 2025, had public proofs of concept published.³ Without access to the original reporter, third-party patch providers cannot validate the fix, and they have no public proof of concept (PoC) available for independent validation. In addition, developing exploit code for every vulnerability patch is time consuming and impractical for support vendors. Because of these constraints, third-party support vendors’ ability to ensure that a patch fully addresses a vulnerability is severely limited.

Reactive vs. proactive

Third-party vendors are always in reactive mode. In contrast, first-party support vendors can be proactive in providing customer patches. Third-party support vendors can begin to work on a patch for a CVE only once it is made public—after bad actors already know about it. Having a time gap before exploitation of vulnerabilities is critical for production environments. A [report from Mandiant](#) revealed that the time to exploit a new vulnerability is currently less than one day,⁴ so any head-start advantage can be extremely valuable.

Since the complexity of each vulnerability is different, the time from announcement to a released patch can vary with third-party vendors. This not only impacts the reactive aspect of the patch cycle for the vendor, but also reduces predictability for customers about when to expect a patch as well.

Missing architectural knowledge

Application code is not developed in a vacuum. There are reasons for each line to exist the way it does. Some of those reasons are obvious. Others are not. Understanding the intent of a software algorithm beyond just reading the source code has material impacts on how to patch bugs and vulnerabilities. This understanding is centralized within the team that originally wrote the software. For example, the team providing [Tanzu Spring support](#) is composed of the same engineers who have designed the Spring Framework since its founding over 20

³ “[Why Early Visibility Matters: Risk Lurks in the Vulnerability Disclosure Gaps](#),” Lucas Tamagna-Darr, Tenable, October 2025.

⁴ “[Understanding the New Negative Time-to-Exploit](#),” Melvin Lammerts, October 2025.

years ago, as well as contributors to Apache Tomcat and OpenJDK. This means Tanzu Spring support has a level of architectural knowledge that is simply unavailable to any third-party vendor.

In addition, when applying fixes and upgrades, the original authors have an understanding of where things are headed, preventing conflicts between fixes and the future roadmap.

Risk of side effects

After regular open source support ends and enterprise support begins, access to tests and other internal documentation is typically cut off to the public (RHEL vs. CentOS is an example of this). While having the patches addresses one risk, this can open enterprises up to new risks of newly introduced bugs or other unexpected side effects.

Compliance and audit

While compliance and audit requirements that depend on “vendor-provided” patches are naturally satisfied with first-party support, supply chain and compliance teams need to assess whether third-party-patched builds meet their policies. Risks of unexpected transitive dependency changes or inclusion of additional software via techniques like shading require additional review to determine alignment and compliance with policies.

Limited scope and total cost of ownership

Third-party support typically provides patches and fixes for old, out-of-support software. This means that newer versions of the software may not be covered comprehensively by third-party support offerings. While they may cover older versions of the software, third-party support lacks the ability to influence the scope and cadence of newer releases, which may result in higher total cost of ownership to a customer or gaps in capabilities and coverage that require organizations to engage first-party vendors anyway.

Limited interaction with the core team

Obtaining enterprise support is not just about access to patches. It is also about having a relationship between customer and vendor that allows for frank, confidential discussions about needs and questions. While third-party vendors may be able to provide some level of patches, they cannot offer insights into product roadmaps or future expectations, nor can they offer support from the project authors.

This limited availability leaves a gap that customers may try to fill using public means, such as opening GitHub Issues against public repositories. In doing so, customers may be introducing risks by inadvertently broadcasting security weaknesses and proprietary usage of business-critical software to the public and potential competitors.

Advantages of first-party support

Obtaining support from the original software vendor is the most secure and reliable way to derisk the lifecycle of open source software. Benefits of first-party software support include:

- **Thorough validation of vulnerability fixes by those who report them** – As the sole committers, the Tanzu Spring team has regular communication with the security researchers who report and validate security issues. First-party support from [Tanzu Spring](#) is the only way to get security fixes that are validated by those who found them in the first place.
- **Proactive patches and scheduling of releases to enable integration into existing priorities** – First-party providers like [Tanzu Spring](#) are able to schedule predictable releases for patches to allow you to better manage deployment of those critical updates into your environments.
- **Long-term alignment with the future of the product** – Maintainers of the original project keep roadmaps and long-term plans in mind as part of their decision-making process while implementing fixes, resulting in the best alignment between enterprise-supported versions and newer capabilities.
- **Guaranteed satisfaction of compliance requirements for “vendor-supplied” patches** – Patches released via first-party developers go through the same testing and release processes that the original software did, eliminating any concerns about supply chain security or fix compliance or coverage.

Conclusion

Open source powers the modern enterprise. Support for that footprint is an important investment and is critical for security, compliance, and safer day-to-day use. Understanding the benefits of first-party support and risks of third-party support is important. [Tanzu Spring](#) is the only industry offering that provides comprehensive, first-party support for your complete Java stack. The application framework bits from the Spring portfolio, through Apache Tomcat, and even the JDK itself all have first-party support through Tanzu. Tanzu also [offers tooling](#) to address a number of enterprise-specific needs, like compliance regulations and security add-ons. Tanzu Spring support integrates with Tanzu Application Advisor, which provides comprehensive upgrade tooling.

Tanzu Spring gives you both first-party support and tooling to help you patch and upgrade your Java stack. This means enterprises get full coverage and peace of mind that they'll be able to keep their software up to date, patched, and compliant.

