Checkmarx

REPORT

2024 State of Software Supply Chain Security

Beyond SBOM: What's Next?

2024 State of

Software Supply Chain Security

Beyond SBOM: What's Next?

High-profile attacks affecting thousands of organizations have prompted stakeholders, ranging from national governments to corporate boards, to take a close look at the risk residing in the software supply chain. Attacks are frequent and malicious actors are weaponizing the open source components that make up a large percentage of applications.

While awareness is important, it is what organizations decide to do that will make the difference in managing software supply chain risk. To understand the current state of software supply chain security (SSCS) we surveyed 900 AppSec professionals in US, Europe and APAC based organizations across a wide range of industries.

The findings show an increased sense of awareness with more than half of respondents acknowledging that SSCS is a top or significant area of focus. However, only 7% have already purchased and implemented an SSCS-specific product.

Thanks to US Executive Order 14028, SBOMs have emerged as a common starting point for the SSCS journey. However, more than half of the organizations that request SBOMs from third parties say they are not using them effectively.

If SBOMs are the start, where should organizations focus next? Right now, there's no clear or strong consensus. Most companies are doing something, but few are succeeding.

In this report

- ↘ See the results of our research.
- ↘ Understand the current state of SSCS.
- ↘ Learn how you start to build a future-proof SSCS program.

SBOMs: Stuck on the Starting Block?

Key Findings

Every organization has been victim of a software supply chain attack

100% of respondents said they were aware of a software supply chain attack against their organization at some point. Of these, 18% said they were attacked within the last year and 63% within two years.

Open source software is a big area of focus

With much of the recent focus on SSCS being on vulnerabilities or malicious code in open source software, respondents estimated that, on average, **about 56% of their applications comprised open source software.**

SSCS is a major area of concern

With a significant percentage of their applications being open source, **75% of respondents said they were either very concerned (39%) or concerned (36%)** about software supply chain security.

As a result, organizations are increasingly prioritizing SSCS

57% of respondents said that SSCS was a top or significant area of focus vs. other areas of security. 85% reported that they were actively using, purchasing, or planning to use a solution.

However, doing SSCS well remains a challenge

Respondents reported big gaps between deploying a solution and using it effectively. **50% said they already request SBOMs from their software vendors**, but less than half of those reported confidence that they knew how to use SBOMs effectively if needed.

Everybody's Talking About SSCS

The first step towards fixing a problem is knowing it exists, and there's no question that awareness around SSCS has soared in recent years. The 2020 SolarWinds attack was a wake-up call demonstarting how a single breach in one organization could compromise thousands of customers and partners.

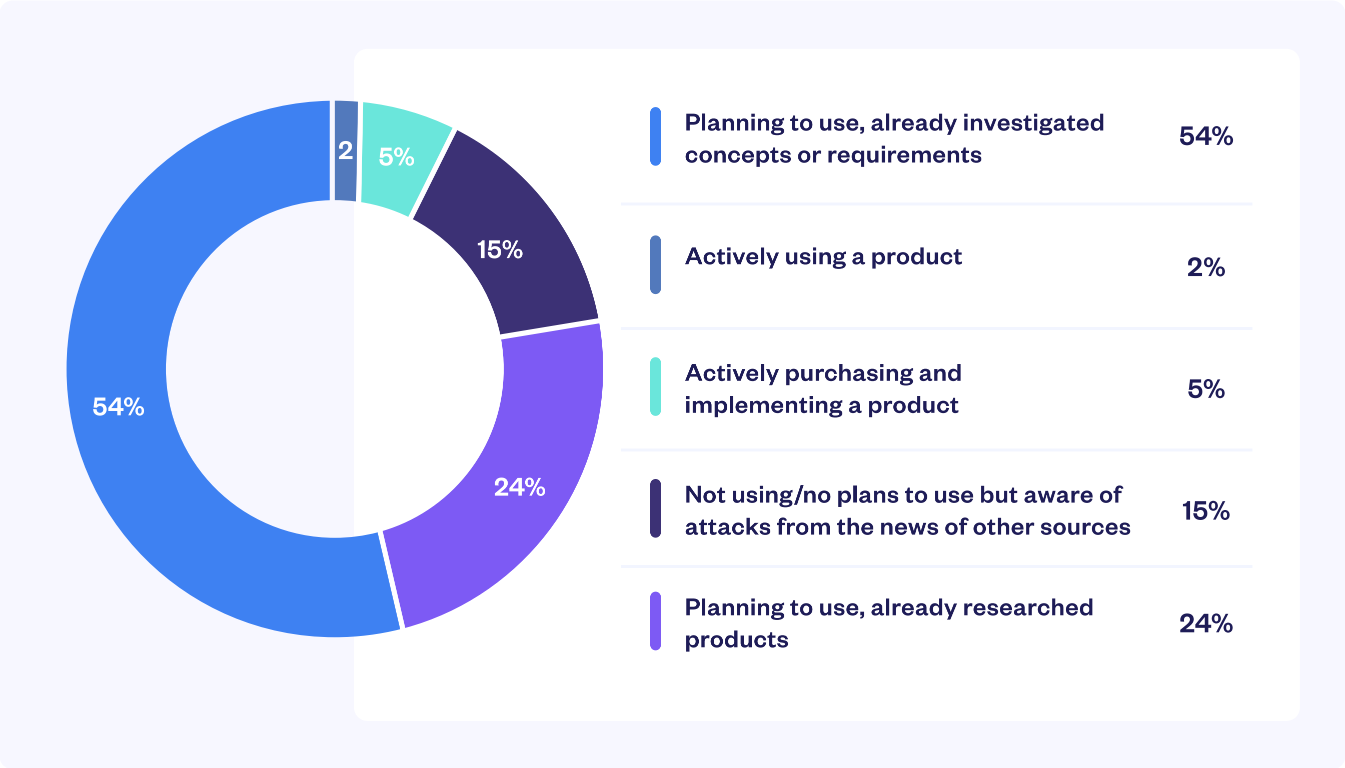
SolarWinds may have been an early high-profile incident, but SSCS attacks haven't slowed down. In December 2023 alone, Checkmarx researchers exposed several major SSCS attacks.

These involved North Korean and Russian threat groups exploiting critical vulnerabilities, poisoning open source and private packages, and stealing cryptocurrency from Ledger Connect Kit users' wallets after compromising a former employee's npmjs account and releasing compromised versions.

These attacks have guaranteed that SSCS stays top of mind. Everyone we surveyed had some level of knowledge around SSCS (Figure 1), and most are trying to act on it.

FIGURE 01

What is your level of knowledge on software supply chain security? (SSCS)



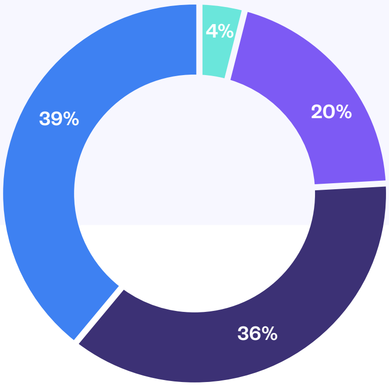
The urgency to act also comes from experience. as 18% of respondents report that they have experienced a SSCS attack in the past year, while 63% experienced one in the past two years.

These figures result from the cycle of zero-day exploit publications. As soon as an exploit is published, attackers update their attack scripts and start scanning public web applications to see what's vulnerable.

As a result, if your applications are exploitable, you will be attacked. This is causing natural concern that SSCS is not as effective as it should be (Figure 2).

FIGURE 02

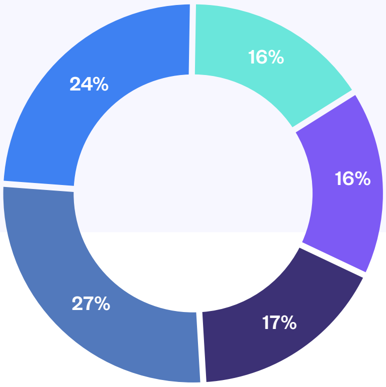
How much are you concerned about SSCS in your organization?



Very concerned	39%
Concerned	36%
Neutral	20%
Unconcerned	4%

FIGURE 03

What is the primary reason you care about or are searching for SSCS?



It's necessary to maintain competitiveness in our market	24%
It's a board-level directive	16%
It's required for compliance	16%
I see a lot of news about attacks	17%
I'm investigating additional risks beyond what we consider today	27%

With awareness comes the realization that SSCS is also strategic. Various factors are prompting organizations to seek SSCS solutions, including board directives, compliance requirements, and competitive pressures (Figure 3).

The most common reason for searching for SSCS solutions is "investigating additional risks beyond what we consider today".

The result is that there is growing focus on SSCS, with eight of ten organizations saying it has the same or greater priority than last year, and 58% saying it is a significant or top area of focus.

SCA as a Foundation for SSCS

There's clearly a strong drive to address SSCS, however most organizations don't know where to start.

Many organizations have a good foundation for building an SSCS program—even if they don't know it—because they are using, or plan to use, software composition analysis (SCA) tools to identify open source vulnerabilities and license risks

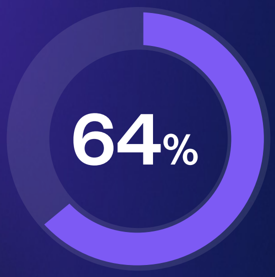
Open source and third-party risk management is critical to controlling application security risk due to the proliferation of open source software (OSS) in application development. Open source software is used in more than 56% of applications on average, according to surveyed organizations.

Attackers have weaponized open source libraries by injecting malicious code into frequently used packages that are subsequently built into applications.

SCA vendors have responded to this threat by adding new checks and protections, such as:

- Checking the reputation of contributors to open source libraries
- Running code to identify malicious behavior
- Looking for instances of dependency confusion, typosquatting, chain and star-jacking.

Checkmarx scans millions of OSS packages every month and has detected more than 385,000 malicious packages to date. A robust SCA solution, featuring what was described above, is a good start to an SSCS program.



of organizations
are already using or plan to use SCA tools.

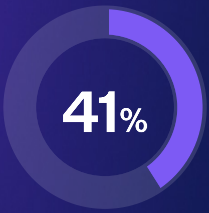
SBOMs: Stuck on the Starting Block?

If SCA represents the (often unknown) start to SSCS, the first conscious effort is typically focused on the Software Bill of Materials (SBOMs).

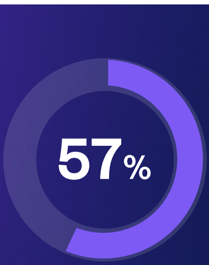
SBOMs are the headline feature and most tangible element of Executive Order 14028. They should contain an accurate list of all open source software ingredients found in a software-based product, creating transparency and visibility so users of the product can act if a vulnerability emerges in a product they are using. At least, that's the theory.

SBOMs are only mandatory for vendors selling software products to the US federal government, but the government's enormous buying power has created a seismic shift that means they have become a de facto standard. As a result, for many organizations today SBOMs and SSCS are one and the same.

Like all standards, however, their value depends on how they are applied. Evidence suggests that there is a lot of room for improvement in how SBOMs are operationalized to support SSCS.



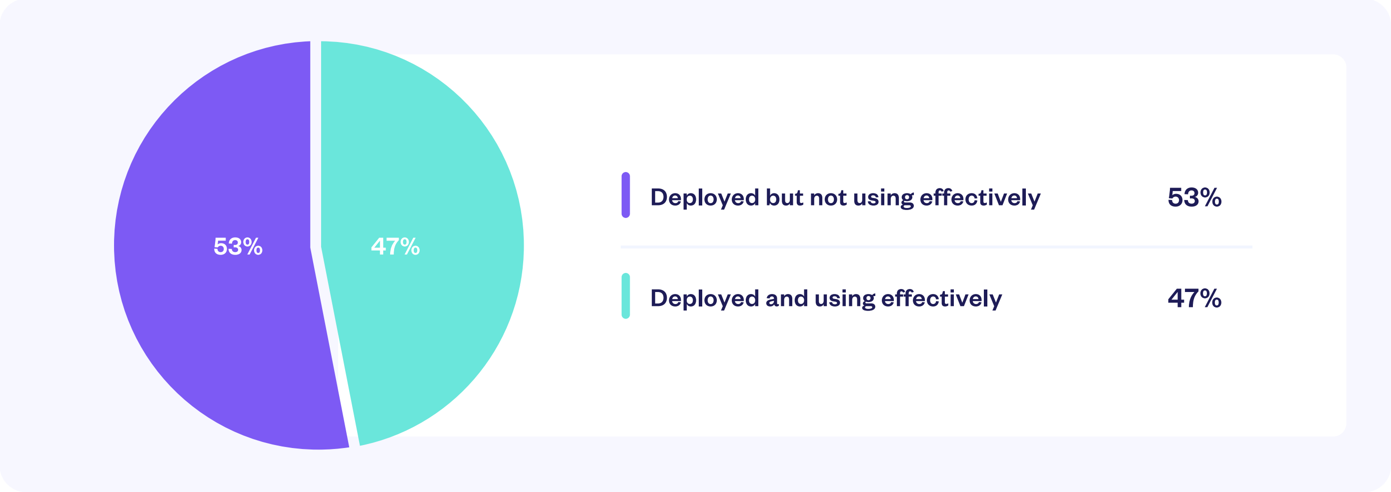
are aware of EO 14028



are aware of EO 14028 but not following it

FIGURE 04

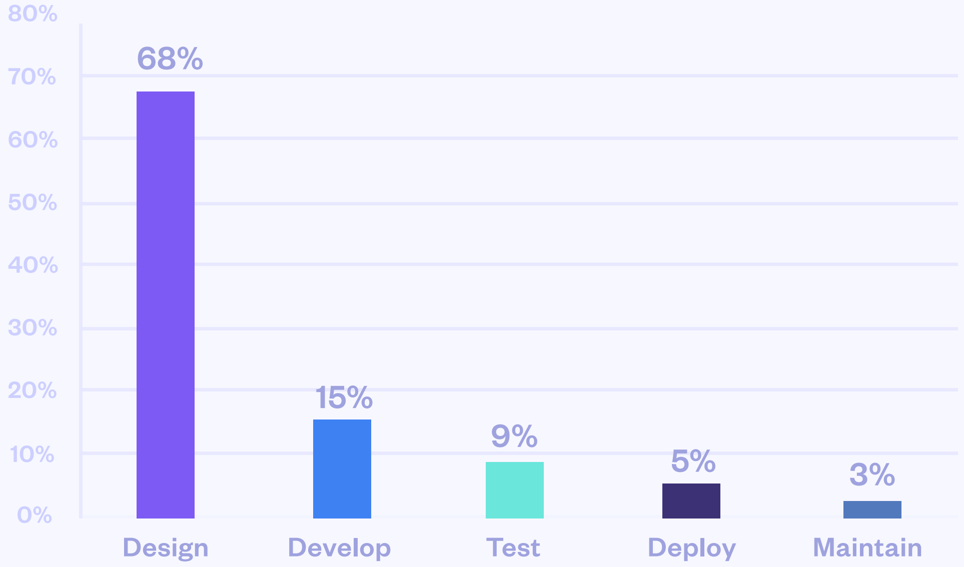
If you are requesting SBOMs from your third-party software providers, how well do you think your organization uses them?



While about half of all respondents say they are requesting SBOMs from third-party software vendors, more than half (53%) of those respondents say that they are not using them effectively (Figure 4).

FIGURE 05

How much of your AppSec focus and resources are dedicated towards each step of the SDLC?



Looking at how AppSec professionals allocate SSCS resources shows why this might be the case. The majority are allocating the most resources to the left of the SDLC, in the design and development stage (Figure 5).

This makes sense when we consider that decisions about open source use are made—and the SBOM itself is created—at this stage. But this is only half of the story. SBOMs are designed to come into their own when a zero-day vulnerability is released. That's when the organization needs to know where all its SBOMs are and how to interrogate them to discover whether the vulnerable library or version is in their application. This sits firmly in the "maintain" stage of the SDLC. More focus on this stage would strengthen the effectiveness of SBOMs and ensure they are playing the right role in SSCS, rather than becoming a tick-box compliance exercise.

Beyond SBOMs: Building an Interdisciplinary SSCS Program

SCA and SBOMs are common entry points to SSCS, but we know that organizations are ready to put a lot of effort into building it out further, so where's a good place to start? For the 50% that are not using SBOMs effectively, fixing that issue is the first order of business. But we can't place all our trust in SBOMs alone; they are part of the SSCS program, but not the whole of it. An SBOM lists the ingredients of an application, but they don't give insight into the process by which it was built, and they don't account for issues such as vulnerabilities in the component distribution system (see box).

In application development, the method is as important as the ingredients; they both contribute to the integrity of the application, and weaknesses in development processes and AppSec workflows can be just as problematic as vulnerabilities in the components used.

This means that there is a range of different SSCS tools and techniques that can be applied to different stages of the SDLC. It also means that SSCS, like application security, is a shared responsibility. An interdisciplinary approach is needed.

Checkmarx Expert Insight:

Tzachi Zornstein explains why an SBOM wouldn't help identify the Ledger Connect Kit attack:

Tzachi Zornstein

Head of Software Supply Chain, Checkmarx

"In the case of the Ledger Connect Kit attack, the primary issue was not with the components themselves but with the compromised distribution process due to an account takeover. The attacker published malicious versions of the package through a legitimate channel, which would not necessarily be flagged by an SBOM. Since the SBOM would list components as usual, it wouldn't identify the malicious code introduced by the attacker in the compromised versions.

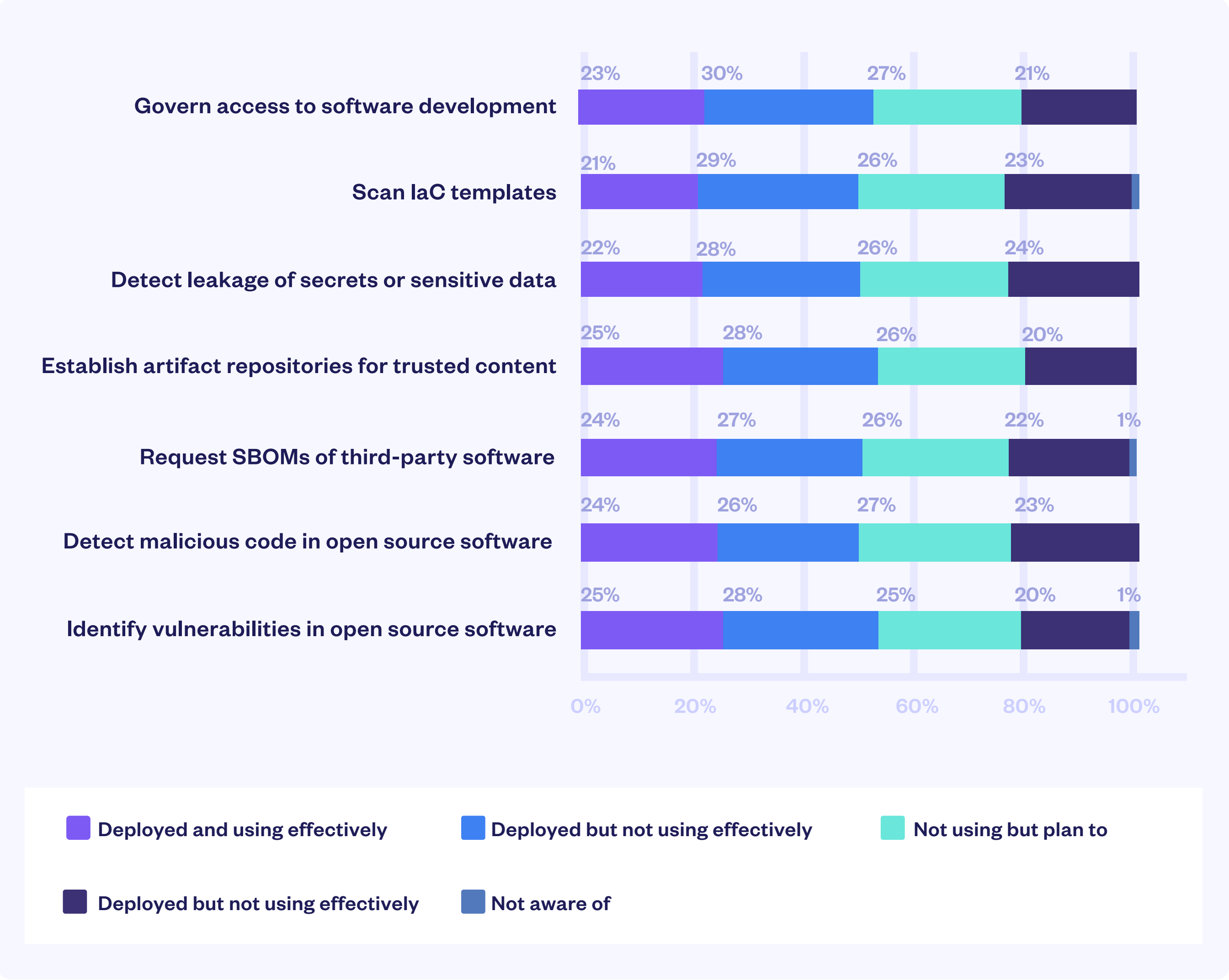
"So, while SBOMs are vital for component transparency, they must be complemented with fast, proactive scanning mechanisms that can detect unauthorized changes or malicious activities in real-time, beyond just component listing."

Allocating SSCS Responsibilities

When looking at some of the tools that contribute to SSCS, we can see that although many organizations are using tools in different areas, few are succeeding across the board (Figure 6).

FIGURE 06

Which of the following SSCS technologies/approaches do you use or plan to use?



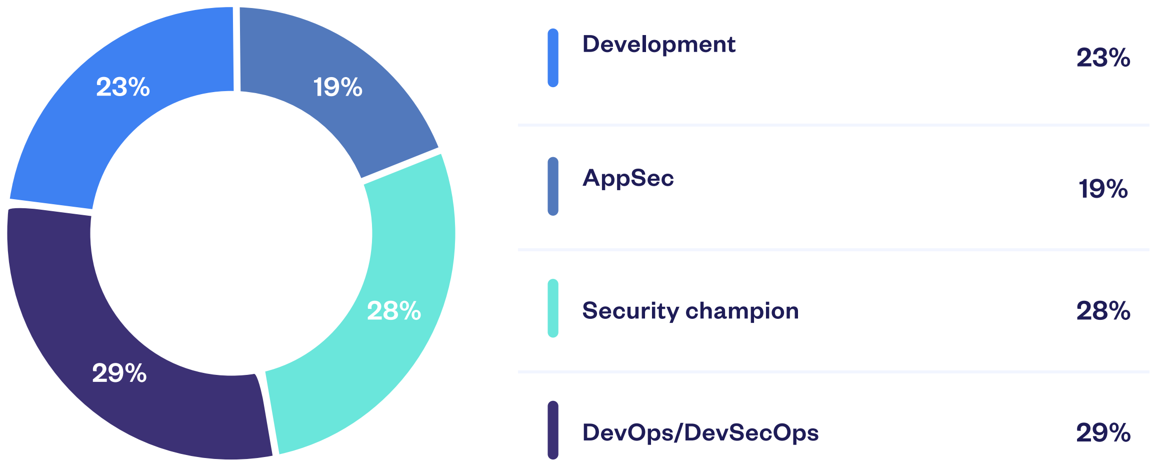
Some of these approaches sit logically with particular teams. Establishing artifact repositories for trusted content and requesting SBOMs for third-party software, for example, are practical actions for developers to take. However, developers won't necessarily prioritize these actions if they are not operating in a culture of SSCS responsibility that is led and promoted by AppSec managers.

Activities such as governing access to software development environments or CI/CD tools are shared between developers and IT departments.

Where AppSec teams come into their own is in working with development teams to specify, set up, and support adoption of tools such as SCA to identify vulnerabilities and malicious code in open source software, IaC security to identify misconfigured resources, and tools to detect secrets leakage.

FIGURE 07

Who has the primary responsibility for SSCS in your organization?



Our research shows that no single department has full responsibility for SSCS (Figure 7), and we'd argue that is just how it should be. It needs to be a shared, interdisciplinary process with good awareness among all stakeholders.

Where organizations have a security champion, they can play a bridging role to get different stakeholders across security and development aligned.

Get Moving:
The Next Steps in Your SSCS Journey

We've established that SSCS is a multifaceted, collaborative undertaking. It's a blend of technology and process, requiring action from development and security teams. It requires collaboration within a culture of security awareness. That said, someone must take the lead and ensure that the SSCS program is adopted by all the stakeholders involved. In that sense, AppSec professionals (drawing support from security champions) are best positioned to drive awareness of SSCS with development teams.

There is a lot of ambition to tackle SSCS right now, so if your organization is ready, how can you get started?

The interdisciplinary nature of SSCS means there are several steps you can take concurrently across different areas, so the first step is to assess what tools and processes you currently have in place and whether they are being used effectively. Once you have your baseline, you can build on it to cover gaps and establish visibility.

Evolving AppSec To Support SSCS

Just like with SBOMs, unlocking the value of AppSec tools depends on operationalizing and managing them effectively. This is a perennial struggle for organizations that typically have too many tools and a program that has grown organically.

Tacking SSCS considerations onto it is unwieldy and can lead to low adoption and low effectiveness—as organizations have already reported.

The AppSec market has matured significantly in recent years, meaning organizations can now consolidate multiple siloed point solutions into integrated platforms that deliver clear visibility over the organization's application security posture. Engage with vendors to see how their tools have evolved, but make sure you interrogate the vendor's roadmap to ensure that the platform will evolve with your needs.

Build on Your SCA Investment

Tackling OSS risk with SCA is the logical place to start your SSCS journey. Whether you are evaluating new SCA tools, or already working with a provider, you should look to do the following:

Expand existing SCA coverage

Check whether the SCA tool can identify malicious code inserted into OSS, not just vulnerabilities. If you can expand an existing tool and remediation process to cover more threats, you avoid introducing extra burden on your teams. And if you're choosing your first tool, you want to know that it detects open source weaponization as well as vulnerabilities.

Integrate SBOMs into SCA

Don't be tempted to buy a standalone tool for SBOM generation—it could quickly be relegated to shelfware. SBOM generation should be part of your SCA tool, and you should be generating an SBOM with every version of software you produce. As mentioned earlier, SBOMs have no value if not used correctly, so ensure they are properly catalogued and stored, with a documented process for how to use them when a zero-day happens.

Explore a platform approach

Consolidating AppSec capabilities such as SAST and SCA on a single platform allows you to correlate findings from different tools, which delivers an exponential improvement in visibility and actionability. It makes management easier, allowing you to manage and triage vulnerabilities in a single location and integrate more easily with developer workflow.

Look right

SSCS should cover the full software development lifecycle. When choosing an AppSec platform look for features that support security on the right side of the SDLC, such as container security and IaC security.

Implementing and refining software supply chain security will be a key activity for companies for the foreseeable future. The regulatory environment is increasingly focused on supply chains; EO 14028 now joined by NIS2 in Europe, with stringent penalties for non-compliance. Even organizations that aren't directly in the scope of these regulations will find they are affected if they want to sell to companies that are.

Every company is part of someone's supply chain. As such, organizations need to start building a comprehensive SSCS program, to better position themselves to compete and thrive in the future business environment.

Methodology

To get more insight into current trends in software supply chain security, we commissioned a survey of 900 CISOs and application security professionals to shed some light on their key challenges and priorities.

The survey was conducted online by Global Surveyz Research, an independent survey company. Respondents included a mix of CISOs, Deputy CISOs, VPs, Directors, and application security managers from companies in North America, W. Europe, and APAC with an annual revenue of $750M+, across a variety of industries, including: Banking & Finance, Insurance, Software, Technology, Engineering, Manufacturing, Industrial, and the Public Sector. The respondents were recruited through a global B2B research panel and invited via email to complete the survey. Answers to most of the nonnumerical questions were randomized to prevent order bias in the answers.

Checkmarx One

Empowering You to Secure Your Entire Software Supply Chain

Discover How →

Checkmarx

Checkmarx is the leader in application security and ensures that enterprises worldwide can secure their application development from code to cloud. Our consolidated platform and services address the needs of enterprises by improving security and reducing TCO, while simultaneously building trust between AppSec, developers, and CISOs. At Checkmarx, we believe it's not just about finding risk, but remediating it across the entire application footprint and software supply chain with one seamless process for all relevant stakeholders.

We are honored to serve more than 1,800 customers, which includes 40 percent of all Fortune 100 companies including Siemens, Airbus, SalesForce, Stellantis, Adidas, Wal-Mart and Sanofi.