WHITE PAPER

CIO Study: Software Build Pipelines Attack Surface Expanding

Current security controls no match for modern attack methods

Venafi®

Introduction:

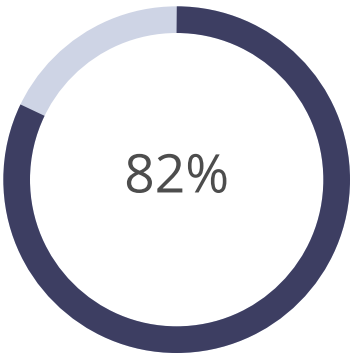
Shift left attacks on software build pipelines have metastasized in the short months since the SUNBURST malware attack on SolarWinds in December 2020. Today's threat actors, most of whom are software developers, quickly recognized that the code being developed in CI/CD pipelines was poorly or improperly secured. As a result, attackers are increasingly shifting left from targeting final software executables, instead infiltrating software build pipelines with malware. It's no surprise that ENISA (European Union Cybersecurity Agency) estimated that software supply chain attacks would increase four-fold in 2021 over 2020.[1]

Defending against supply chain attacks requires a fundamental change in security strategy. To better understand how prepared organizations are to counter this rapidly increasing threat, Venafi sponsored a study by research firm Coleman Parkes of 1,000 CIOs from six regions: United States, United Kingdom, France, DACH (Germany, Austria, Switzerland), Benelux (Belgium, Netherlands, Luxembourg) and Australasia (Australia, New Zealand). The results show that while CIOs understand the risk of these types of attacks, they have yet to grasp the fundamental organizational changes and new security controls they will need to incorporate into their security posture to reduce the risk of supply chain attacks that can be devastating to themselves and their customers.

Section 1: Most CIOs understand the need to bolster security for software build pipelines

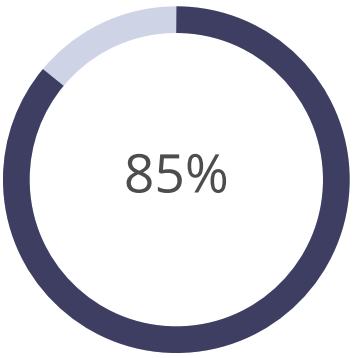
CIOs seem aware of the potential damage that a software supply chain attack can bring about. In addition to the attack on SolarWinds, news reports of similar attacks on Codecov and Kaseya are striking reminders that the rising instances of these types of attacks not only leave organizations vulnerable but their customers as well.

And CIOs are worried that their defenses against these types of attacks are inadequate at best:



82% of CIOs believe their organization is vulnerable to cyberattacks targeting their software build and distribution environments

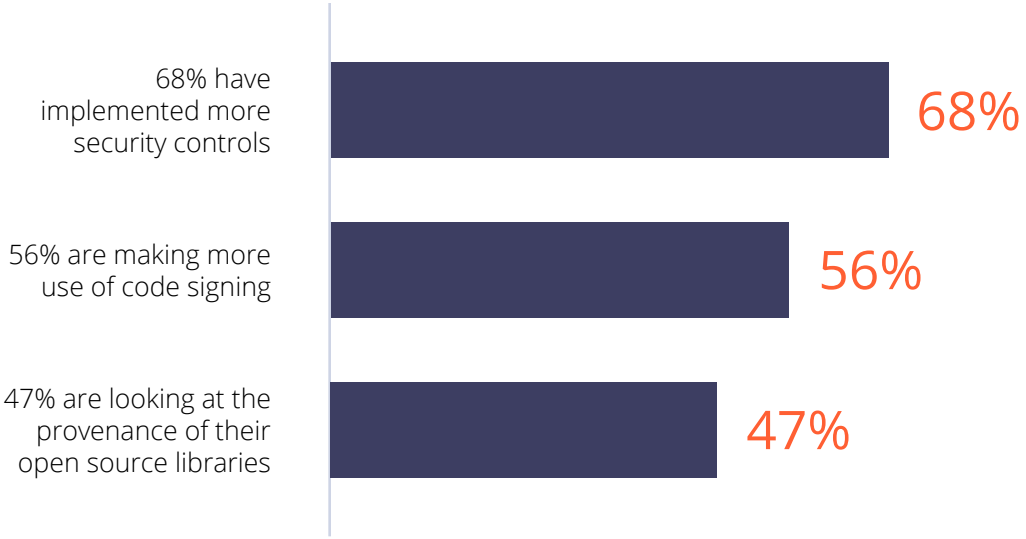
But there is an awareness that the software supply chain needs to be bolstered. In the wake of recent software supply chain attacks, CIOs have been given a mandate to neutralize the problem:



85% of CIOs have specifically been instructed by the board or CEO to take action to improve the security of software development and build environments

CIOs are aware that software build pipelines pose unique security challenges most enterprises haven't encountered in the past. Many have already taken preliminary steps to tackle the problem on multiple fronts:

How has your approach to securing software development and build environments changed?



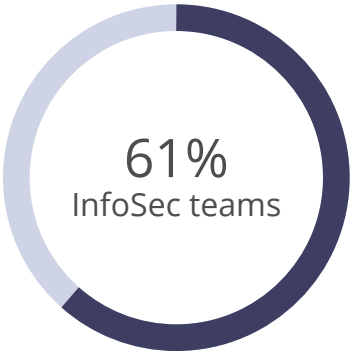
However, CIOs and organizations seem hesitant to address critical security controls needed to effectively support software build pipelines, which require a fundamentally different security structure. Software developers and engineers—the people most familiar with how these new and complex environments work—need to play a key role in their defenses.

Section 2: Siloed efforts prevent effective security responses to software build pipeline attacks

Enterprises have digitally transformed themselves in the past five years, migrating from the traditional data center to hybrid cloud and cloud native infrastructures. The modern enterprise is embracing public cloud instances (AWS, Azure, GCP), applications and services that include microservices, containers and APIs. As a result, development and software engineering teams oversee many of the security controls for these environments.

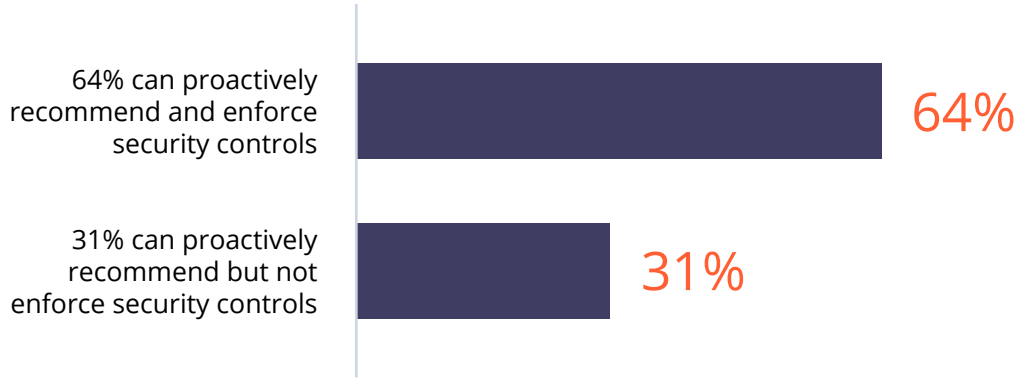
However, InfoSec teams still tend to be responsible for owning and managing security for software development and build environments, even though they often don't have the visibility into what software engineering teams are doing:

In your organization, who has oversight and ownership of the security of software development and build environments?



More notably, 95% of InfoSec teams have authority over what security controls should be used to protect software supply chains. At the same time nearly a third of InfoSec teams cannot enforce the policies they recommend:

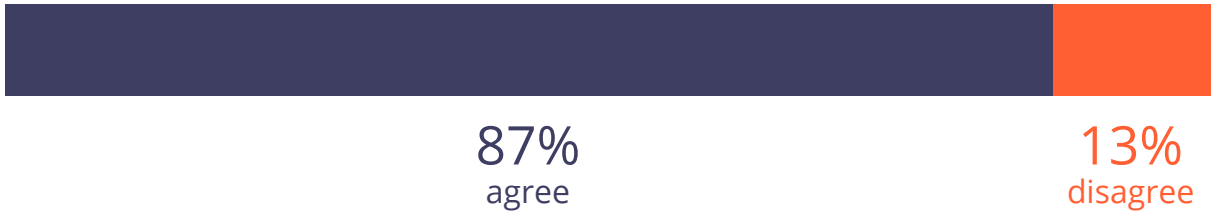
How much control does your security team have over software development and build environments?



This is concerning because InfoSec is expected to take responsibility for securing critical environments that they don't quite grasp. And more often than not, silos between InfoSec and development teams means that the former have little insight into how to approach the problem.

Meanwhile, developers are seen as less than committed to maintaining software supply chain security, as suggested by CIOs:

Software engineers/developers sometimes compromise on security policies and controls to meet the organization's drive to develop new products and services quickly.

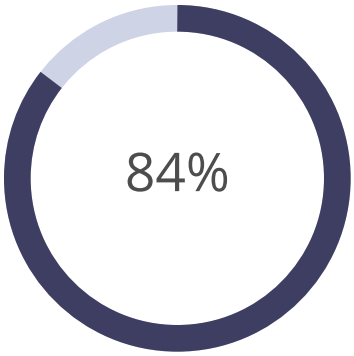


But this disconnect shouldn't be mistaken for a lack of concern. No developer wants to put out insecure code. The problem is that developers are incentivized to deliver features and functions necessary to meet rapidly evolving business goals. And these pressures are not alleviated when software engineering teams are saddled with traditional security controls that are recommended by the InfoSec team. When forced to decide between slowing down the pipeline and maintaining peak functionality with development, software engineering teams generally choose functionality to meet the business goals they've been tasked to achieve.

This outdated presumption that effective security is diametrically opposed to speed of development hinders organizations from addressing the challenges of securing software supply chains—despite the increases in budget dedicated to solving this problem.

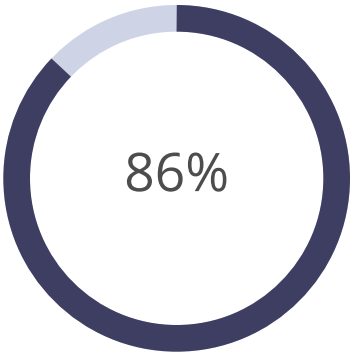
Section 3: Securing software build pipelines requires a commitment to change and additional budget

According to the data, organizations are not only concerned about the threat of software supply chain attacks, but they are also reallocating budget to address them in 2022:

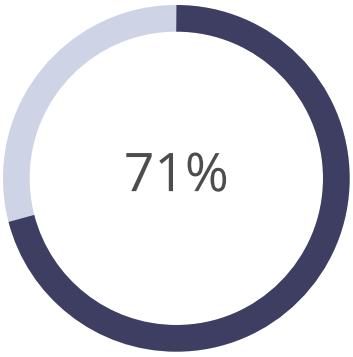


84% have increased their budget dedicated to the security of software development and build pipelines

In addition, organizations are increasing their 2022 security budgets for solutions directly related to software supply chain security:



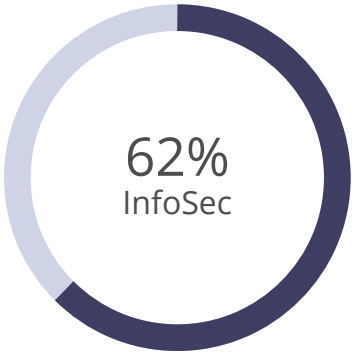
86% have increased their budgets dedicated to IAM (Identity and Access Management) for software engineering environments



71% have increased their budgets dedicated to code signing

But the real question is which team pays for these additional security controls? It shouldn't be surprising, based on the data in the previous section, that in a majority of organizations, InfoSec does:

In your organization who holds the budget dedicated to the security of software development and build environments?



This percentage aligns with the degree of control InfoSec teams have on managing security for software supply chains. But the problem with that scenario is that InfoSec teams often lack a depth of knowledge about the intricacies of software build pipelines. And unlike development teams, InfoSec is challenged to maintain the same levels of visibility for cloud native development that they have had in traditional data center environments. This challenge is complicated by the diversity of cloud instances and the applications that run on them, including the ways in which microservices are brought together using container orchestration and cluster management, among other aspects.

Moreover, any security controls that InfoSec teams recommend to secure the software supply chain must not hamper development times for software engineering teams. If anything, these security controls should facilitate even faster code building.

Conclusion: Why code signing must shift left in software build pipelines

In the past, code signing was only used when software, such as applications, updates and shell scripts, were finalized, so that end users could verify its authenticity. Software artifacts used in the process of developing these final versions traditionally were ignored.

Now that threat actors have shifted left to infiltrate poorly protected CI/CD pipelines, organizations are starting to realize that every piece of code used to build software—including source code, build scripts, software libraries, execution containers like Kubernetes and the tools developers use to build software—must also be signed. By shifting left and securing code signing processes throughout software build pipelines, organizations can catch unsigned malware before it can wreak havoc on their enterprises and their customers.

The CIOs in this survey seem aware of the importance of code signing in securing software supply chains, with 71% increasing their budgets dedicated to code signing and another 56% making more use of code signing to counteract potential attacks. However, to leverage the value code signing can provide, organizations should look for a solution that can automate most, if not all, code signing processes, so that developers can stay secure as they build code fast.

Learn more about how to get a free Secure Software Build Pipeline Risk Assessment: **venafi.com/code-signing-checklist**

A Venafi expert will meet with you and your software teams to:

- Evaluate your current software build pipelines risk
- Suggest a plan of action to address these risks

Or learn more at **venafi.com/platform/codesign-protect**

References

1. Enisa Threat Landscape for Supply Chain Attacks. ENISA. July 2021. 3.

Trusted by

5 OF THE 5 Top U.S. Health Insurers
5 OF THE 5 Top U.S. Airlines
3 OF THE 5 Top U.S. Retailers
3 OF THE 5 Top Accounting/Consulting Firms
4 OF THE 5 Top Payment Card Issuers
4 OF THE 5 Top U.S. Banks
4 OF THE 5 Top U.K. Banks
4 OF THE 5 Top S. African Banks
4 OF THE 5 Top AU Banks