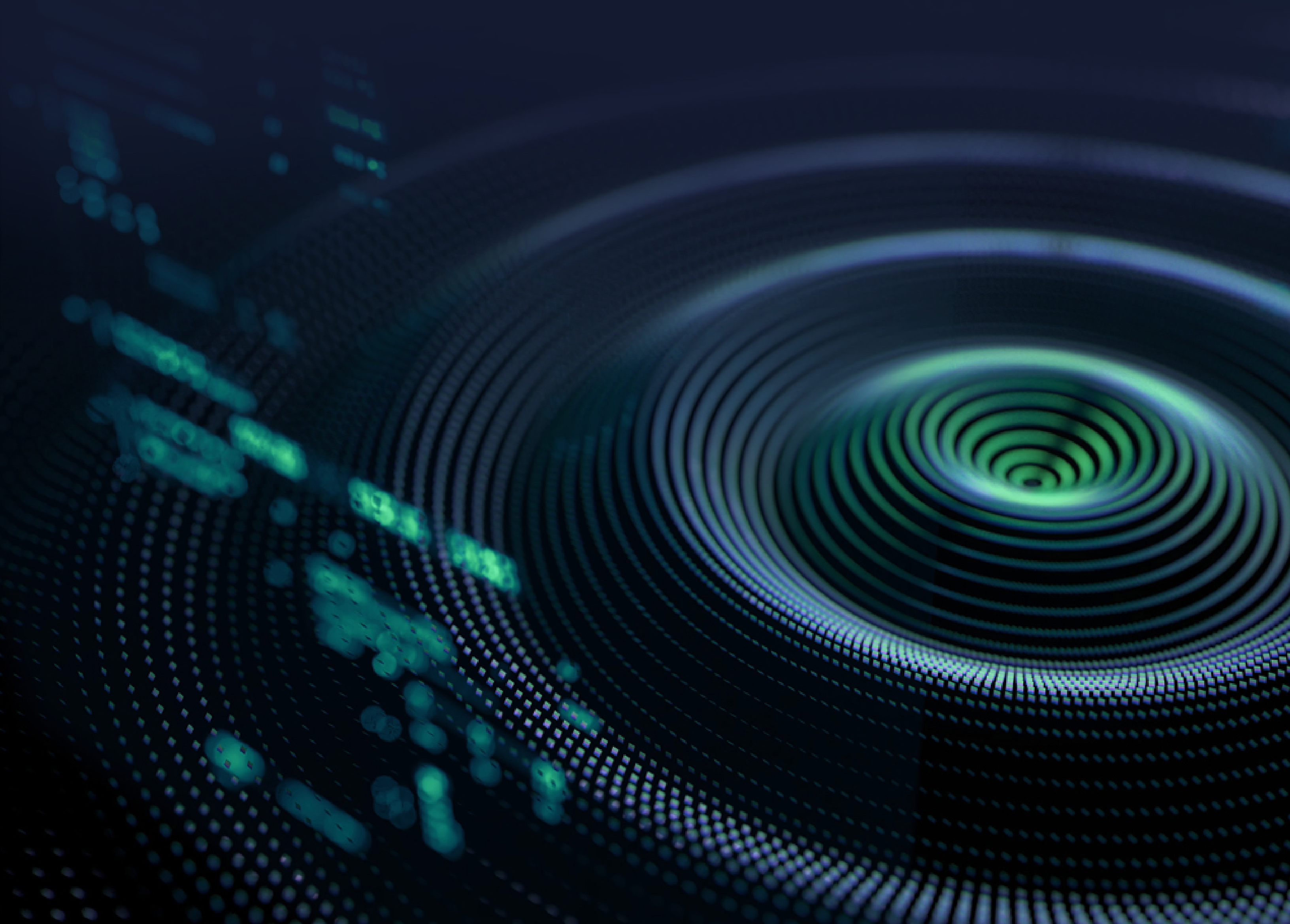
BUSINESS BRIEF

REVERSINGLABS

# The 3CX Software Supply Chain Hack and The Impact on Publishers and Consumers



The total cost of software supply chain attacks will exceed

**$80.6 billion**

by 2026

**77%**

of survey respondents have suffered from software supply chain attacks

The average company has

**254**

third party software applications

**40%**

of survey respondents believe that their software supply chain security measure are ineffective

By

**2025**

45% of companies are projected to suffer from software supply chain attacks

# Introduction

In 2022, Gartner predicted that by 2025, 45% of organizations will suffer from supply chain attacks[1]. To keep that from happening, software vendors need help managing supply chain risks, identifying active threats and attacks, and preventing them from occurring.

Software supply chain attacks occur when hackers access a vendor's software development and delivery environments and embed malicious changes into their product. This happens before updates are sent to customers. Once the malicious updates are distributed to customers, attackers are able to compromise the endpoint systems and network environments of the customers that downloaded and applied the update.

These incidents profoundly impact vendors and their customers. Organizations can establish preventive measures and reduce their risks by learning about how they occur.

# 3CX Supply Chain Attack - Timeline & Background Information

On April 1st, 3CX, a voiceover IP (VoIP) software vendor, reported that it was the victim of a supply chain attack, shipping malicious code to thousands of 3CX customers, a number of which subsequently reported compromises of their internal systems[2].

According to the company's account of the incident, 3CX was alerted of this attack on March 22nd, when SentinelOne flagged an update to the 3CXDesktopApp client as malicious. When 3CX tested the update against various antivirus engines, it was not detected as malicious, causing 3CX to believe that SentinelOne's alert was a false positive. On March 29th, Crowdstrike and others scanned the same client update and also declared it malicious. The company gave 3CX the "full details" regarding the attack, causing 3CX to disclose the incident to its customers shortly after that[3].

Sources:

[1] "Gartner Top Security Risk Trends in 2022"
https://www.gartner.com/en/articles/7-top-trends-in-cybersecurity-for-2022

[2] "Security Incident Update: Saturday 1 April 2023"
https://www.3cx.com/blog/news/security-incident-updates/

[3] "3CX Confirms Supply Chain Attack"
https://www.securityweek.com/3cx-confirms-supply-chain-attack-as-researchers-uncover-mac-component

ReversingLabs’ threat research teams analyzed the malicious update and concluded that it was likely derived from a compromise of 3CX’s build pipeline. Specifically, ReversingLabs found discrepancies in 3CX's versions of two standard libraries used with the Electron open source framework on which the 3CXDesktopApp client is built: ffmpeg and d3dcompiler_47.

RC4 encrypted shellcode was added to the signature appendix of d3dcompiler, a (Microsoft) signed library, without altering or invalidating the signature. A reference to the d3dcompiler library was added to the ffmpeg library, which was used to call and execute the appended shell code. Though there are legitimate applications of appending code in this way, ReversingLabs researchers noted that this strategy has been used by advanced persistent threat (APT) groups to shuttle malicious code onto a system alongside a signed executable.

As the 3CX incident shows, software supply chain attacks are highly targeted and sophisticated, making them difficult to identify and respond to. Many legacy tools like SAST, DAST, and SCA tools are unable to fully analyze modern software packages, leaving coverage gaps. This caused 3CX to unwittingly facilitate a supply chain attack and have a false sense of security as they allowed their users to continue to be exploited.

## Flawed Security Mindsets & How Legacy Tools Leave Users at Risk

Many development teams use legacy application security testing tools with inadequate coverage and are more interested in releasing code quickly than following security processes. This leaves development pipelines exposed to tampering and can lead to development and security teams overlooking threats lurking in deployed code.

For example, commonly used software composition analysis (SCA) tools typically locate vulnerabilities when scanning open source components and workloads. However, they fail to identify active threats embedded into their development environment. These tools only protect open source components; have limited policy customization; and may generate alerts with little to no context, providing partial coverage and inefficient security operations. SCA tools’ limitations lead to unidentified threats, inconsistent security practices and policy enforcement, and excessive noise, causing greater risk for software supply chain attacks.

Additionally, because they cannot identify severe threats embedded in the codebases of software products or detect malicious behaviors, software vendors and consumers falsely believe that their software or development environment are secure when they are not.

# How Vendors Protect Against Supply Chain Attacks

To prevent facilitating supply chain attacks, vendors must verify the integrity of software releases; design software to meet security standards; and review, test, and analyze code. These practices allow them to continuously scan for malware and tampering throughout the software development lifecycle (SDLC), create consistent security standards, and test the integrity of their code to detect severe threats, ensuring that they release secure updates to their customers.

| **Software supply chain - key practices for vendors** | **Why it's important** |
|---|---|
| Verify software releases' integrity | Confirm that software is secure and ready to be released |
| Design software to meet security standards | Create clear, complete, and consistent security standards |
| Review, analyze, and test code | Identify active threats and how they impact your code |

| **How ReversingLabs helps** | **How it works** |
|---|---|
| Analyze changes in packages and who made them to detect tampering and malware before updates are deployed | Monitors digital signatures and uses world's largest repository of malware and goodware to identify who is making changes and whether they are malicious |
| Continuously scan and test new products and builds and create custom policies to assess compliance with best practices | Uses custom policies to determine what to scan and how to prioritize alerts, and with continuous scanning, compliance is enforced in real time |
| Identify baseline behaviors, scan for anomalies, and detect suspicious behavior | Monitors the time of day and extent of changes to determine whether the behavior is suspicious |

ReversingLabs protects vendors from supply chain attacks by continuously analyzing changes in compiled packages, reviewing what actions occurred which led to malware and tampering being detected, supporting custom policies, and determining baseline behaviors to understand which actions are abnormal or malicious. The ReversingLabs Software Supply Chain Security platform enables vendors to identify severe threats and behaviors before updates are shipped to customers, preventing supply chain attacks.

# How Consumers Protect Against Supply Chain Attacks

It is crucial for software consumers to continuously monitor their attack surface, identify suspicious behaviors, track security posturing, and reassess integrity. These measures allow them to know the number of vendors they are working with to determine general risk, identify suspicious activity, continuously track vendors' security profiles and compliance, and understand how vendors update their products and if they are safe to deploy.

| **Key practices to monitor vendors** | **Why it's important** |
|---|---|
| Establish a software supplier inventory | Know all of the vendors that you are working with and define your risk |
| Establish criteria for a risk based approach | Monitor vendors for events such as malicious behaviors, weakening security postures, and deviations from compliance standards to identify new risks |
| Establish minimum software security standards for compliance tracking | Create policies to track vendors' ability to follow risk and compliance standards |
| Conduct continuous attack surface monitoring | Consistently reassess vendor integrity to ensure continued compliance and identify new risks and threats |

| **How ReversingLabs helps** | **How it works** |
|---|---|
| Generates a software bill of materials (SBOM) that lists all third-party software components | Discovers components' general information such as: Supplier, version, and author name, relationships with other dependencies, as well as the last time an SBOM was collected |
| Validates software components by analyzing codebase changes and who made them to detect tampering and malware before deploying product updates | Evaluates the timing, frequency, and location of new product updates to determine whether behaviors are suspicious |
| Tests product updates and creates custom policies to assess compliance with best practices | Creates custom policies to determine what to scan and how alerts are classified to find and quickly remediate issues |
| Continuously scans your environment and assesses third-party risks in real-time | Collects scheduled, recurring SBOMs to consistently evaluate the size of your attack surface and risk |

ReversingLabs protects software consumers from supply chain attacks by collecting a software bill of materials (SBOM) showing third-party software components, analyzing changes in compiled packages, and detecting tampering and malware before product updates are deployed. These capabilities help consumers understand risks and consistently and quickly identify severe supply chain threats.

# ReversingLabs SSCS vs. Legacy Tools

legacy tools use vulnerability databases, analytics, and built-in policies to determine whether open source components contain vulnerabilities, observe the health of users' environments, and enforce common security protocols. However, they do not identify active threats. Furthermore, they provide alerts with little to no context, leading to unidentified supply chain threats and longer response times for enterprises.

ReversingLabs' Software Supply Chain Security (SSCS) Platform uses the world's largest repository of malware and goodware, detailed threat intelligence and context, and custom policies to identify malware and tampering, provide alerts with recommended steps for remediation, and allow users to decide what to scan for and what is important to them. This allows vendors and software consumers to address and quickly respond to severe threats which directly lead to supply chain attacks. Also, SSCS has in-depth scanning, finding threats embedded 10+ layers into large file types with teams of malware researchers expanding the detection engine, helping users find and remediate threats that other tools cannot see. While SCA tools evaluate third-party and open source inclusions, the ReversingLabs SSCS platform analyzes the full software package, allowing for greater visibility into the most critical threats.

## Get Started!

REQUEST A DEMO

www.reversinglabs.com

## About ReversingLabs

ReversingLabs, the leader in software supply chain security, offers the only holistic software supply chain solution that secures third party software and open source components and identifies active threats. ReversingLabs provides protection against malware and tampering at all stages of the development, build and release process. For more information or to schedule a demo, contact us today.

REVERSINGLABS
Worldwide Sales : +1.617.250.7518
sales@reversinglabs.com