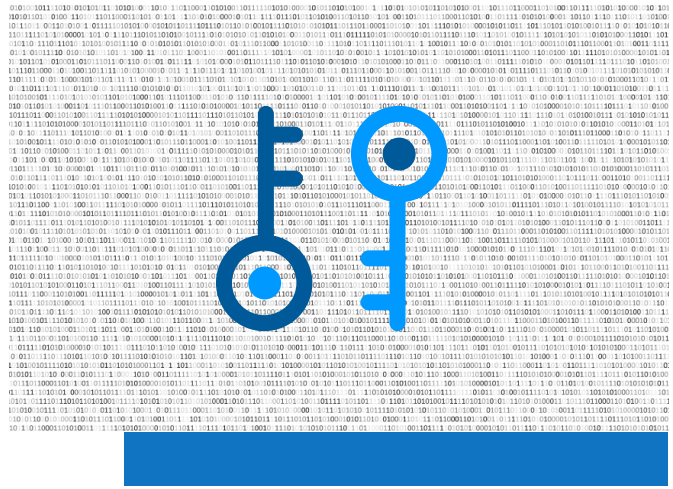
THE TOP 10 QUESTIONS CIOS SHOULD BE ASKING ABOUT THEIR DEVOPS—BUT AREN'T.

digicert®

01

Do we have controls in place that limit and track who has access to signing keys throughout our organization?

Without controls, legitimate keys can be used by malicious actors to sign software they've infected with malware, signaling to your customers that compromised software should be trusted. If you're not tracking all your signing keys and their locations, you're putting your customers, and the reputation and financial health of your organization, at risk.



02

Do we know who has permission to sign software, and the ability to control how those signing keys are used?

Signing must strictly follow key use policies. If you don't have controls or visibility in place, individuals can sign without following organizational or legal compliance. Tracking and monitoring key use allows administrators to set permission-based key access and intervene if misuse is detected.

03

Do we know which code signing certificates have been generated using our administrators' keys?

Signed software represents a trust agreement between your organization and your end user. If admins aren't tracking who issues certificates with their keys, legitimate certificates can be generated to sign compromised software or unauthorized software releases using the trusted identity of your organization.

04

Do we have the ability to remove user access from keys and certificates during an event, or when a user changes roles or leaves our company?

Whenever someone leaves your company, you need to revoke their signing privileges. If they change roles, their access should be amended to fit the signing needs of that role. User management controls allow you to set signing privileges according to role, responsibility, or project, so keys and certificates are used by the right people in the right timeframe.

05

Are we requiring that users provide multi-factor authentication for signing, keypair generation or certificate creation?

Signing, keypair generation and certificate creation are critical activities that must be secured. If unauthorized users perform these activities, they may cause a security breach. They may also trigger a supply chain attack by signing and releasing software that has been infected with malware.

Multi-factor authentication ensures individuals are who they say they are, so access to signing tools remains restricted to authorized users.

06

Are we tracking and securing physical tokens, USBs, and HSMs?

Like a house key or a car key, you need to ensure the physical keys used for code signing aren't lost, shared, or stolen. Your keys should be tracked at all times, so you know where they are, when they're used, and who used them.

07

Do our developers share code signing certificates?

Key sharing is so common in the DevOps world, many popular repositories actually recommend the dangerous practice in their user guides. When your developers share keys, you lose visibility and control over who signs what and when they sign. If you find yourself in a situation where you need to revoke the certificate associated with a key, you will have to revoke every piece of software with that attached certificate.

08

Are we making reproducible code to ensure no malware is injected during the build process?

Reproducible code allows you to duplicate the build process and compare releases. If the binary outputs are identical, you can feel more confident in signing and releasing software, knowing it is far less likely that malware was injected. This is one of the best defenses against supply chain attacks, especially if you're working with open-source code and 3rd party libraries.

09

Are we monitoring and auditing all signing events inside our organization?

Security isn't only about external releases. Internal CI/CD processes must be secured with as much scrutiny and trust as any software passed down the supply chain or released to outside end users. As with any other release, you need to know who signed, what they signed, and when they signed, even if the software isn't leaving your environment. Not only does this protect against accidents and malicious intent, it also helps you maintain compliance while strengthening your security profile.

10

Are we tracking whether code and software releases are signed at all?

Signed software tells your partners and customers they can trust the integrity of the code and software. When code has been scanned and protected during the build, using properly managed keys and key signing user access, that software should be signed before delivery. This protects the software in transit, creates an auditing record, and signals to your partner or customer that you've verified the integrity of the software before passing it along. Managed code signing services and automated tools can greatly increase the efficiency of this signing process, ensuring there's no interruption or delay to your agile CI/CD pipeline.

DevOps security should never be an afterthought. If you're asking the right questions, you're thinking about proactive protection for your CI/CD pipeline—and the entire supply chain.

We can help you bridge the security gap and finally close your DevOps loop. Contact us to find out how. pki_info@digicert.com