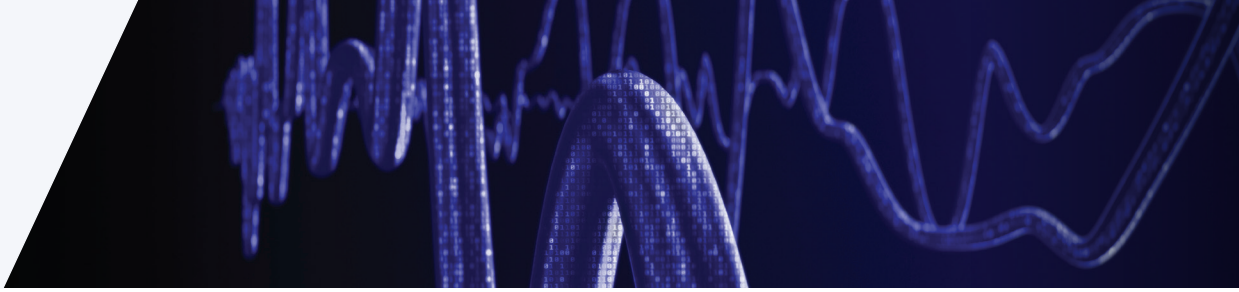
Venafi



CASE STUDY

Healthcare Technology Firm Implements Secure Code Signing in CI/CD Pipeline

Challenge: Homegrown code signing solution no longer serves developers' needs

A leading healthcare technology provider realized they had outgrown their current code signing processes. As a software vendor, they had long been concerned about securing their dev environment, having built their own code signing solution eight years earlier. That solution was designed for waterfall development processes and was no longer able to keep up with DevOps methodologies.

The most pressing problem was that app developers working in Jenkins couldn't easily sign JAR files. The homegrown tool routinely crashed as developers tried to move code from the CI (continuous integration) to the CD (continuous development) part of the CI/CD pipeline. Code signing could take hours—even days—to complete. These delays in the code signing process significantly slowed development cycles, creating stress for developers trying to meet their business goals for fast software development.

Meanwhile, the InfoSec team, already feeling pressure from frustrated development teams, found the homegrown solution failed to provide visibility into the company's code signing key population, let alone any intelligence into key ownership and usage. Once a developer was issued an internally trusted code signing key, InfoSec couldn't track what code was actually being signed with that key. And if that key were stolen, InfoSec wouldn't know about it until the damage had been done.

App developers and InfoSec mutually concluded it was time to retire their homegrown solution, which would be too costly to overhaul let alone maintain. Instead, the company needed a dedicated solution that would provide visibility and intelligence to InfoSec while helping developers sign code faster. Luckily, the CISO knew where to find it.

Solution: Venafi CodeSign Protect

The company already used the Venafi Platform for TLS machine identity management, and the CISO had been blown away by its effectiveness. He asked Venafi to perform a POC of CodeSign Protect for him and the app development and InfoSec team leads.

Venafi showed how CodeSign Protect would let developers use their existing processes to sign JAR files for their Oracle application, no changes required. "This approach seemed almost magical," said the CISO. "We assumed this particular use case would be hard, if not impossible to solve—but Venafi made it easy!"

Added the CISO: "I sensed the same company that eliminated our outage risk could help us with our code signing challenges. I was right."

Securing Private Code Signing Keys

Using CodeSign Protect, the InfoSec team could now secure code signing private keys in a centralized location. Developers could still sign code on their own laptops to compile and debug it just as they always had. But they no longer had direct access to private keys. Instead, the keys were stored in an HSM that the InfoSec team could monitor and audit for signs of misuse.

CodeSign Protect also enabled the InfoSec team to work with application developers to design approval and process workflows, as well as process controls. Not only did code signing keys remain secure, developers could sign their code quickly and without friction. The teams locked down who had permission to access code signing keys using the principle of least privilege rather than usernames and passwords that were harder to secure. They were also able to customize the level of access granted to developers to comply with corporate security policies and industry regulations.

CodeSign Protect even gave InfoSec the ability to pre-approve code signing processes granularly. Depending on the code being signed, the process could happen automatically if everything complied with policies. InfoSec could also add an extra step that would require the developer's manager to review and approve for specific requirements. CodeSign Protect's robust automation capabilities made deploying these changes easy.

Simplifying code signing processes, accelerating code development

Developers were pleased to discover the ways in which CodeSign Protect accelerated their overall productivity. They no longer had to worry about how to request new code signing certificates because CodeSign Protect automated this process. Moreover, they could now provision code signing keys without having to write scripts for each use case. And they appreciated how seamlessly CodeSign Protect worked with their preferred toolsets without any modifications on their end.

Also, CodeSign Protect eliminated the crashes developers previously experienced while signing JAR files. Now developers could sign their code during the CI part of the build pipeline without any problems. Because CodeSign Protect reduced complexity, the company's Jenkins pipelines now could use Venafi's standard SDKs. This was revelatory because before CodeSign Protect, developers had to configure their JDK every time they needed to sign code.

As a result, the company's Java developers were able to cut the code signing process from hours down to just a few minutes. They no longer had to wait to obtain code signing keys or change the way they developed code. And they appreciated no longer having to be responsible for the security of individual code signing keys. Keys were safely stored in an HSM, where they were managed using automated lifecycle management. In fact, CodeSign Protect now made it possible to automatically provision different keys for different code without developers having to think about it. Said one developer: "I don't miss having to keep track of private keys. The right key gets issued automatically, and I can focus on code development."

Turning developers into machine identity management evangelists

CodeSign Protect has been such a breakthrough for the company's application developers that they are now acting as evangelists for the solution with other teams. The InfoSec team is currently in the process of onboarding internal development teams that work extensively with Windows PowerShell—and they are seeing the same transformative benefits as the app developer teams did.

The CISO has never seen this level of excitement about a security project before—especially one that affects developers. "I can't say it's surprising," he noted. "After all, CodeSign Protect signs code quickly and easily. Whether it's JAR files, PowerShell scripts, installable Windows binaries or pretty much anything else, CodeSign Protect gives my guys the ability to say yes to development teams and help them protect any kind of code without having to do technical somersaults. It's easily one of the best security solutions I've purchased in my 20 years as a CISO."