digicert®

Best Practices for Automating Software Trust in Modern CI/CD Pipelines

GUIDE

Why automating code signing is important

Software trust used to be something teams only thought about at release time. Today, it is a continuous requirement that spans development, build, release, and long-term maintenance. Shorter certificate lifecycles, more frequent releases, new and deprecated encryption algorithms, and rising software supply chain threats are forcing organizations to rethink how trust is established and maintained at scale.

Recent customer survey data clearly highlights the gap between awareness of new certificate requirements and teams' ability to operationalize them at scale. While many teams understand that change is required, far fewer have implemented processes that remove risk and friction from day-to-day delivery.

The takeaway is straightforward. The challenge is not knowing what needs to change. It is executing software trust in a way that works with modern CI/CD pipelines instead of slowing them down.

This guide outlines seven actionable best practices that security and engineering teams can use to automate software trust, reduce operational risk, and remove unnecessary friction from the release process.

1. Establish Software Supply Chain Provenance and Transparency

Trust starts with knowing what you are signing. In complex environments with open-source dependencies, third-party components, and multiple build systems, it is often unclear which artifacts are entering the pipeline and why they require a trusted signature.

Strong software trust practices focus on provenance and transparency by consistently answering three questions:

1. What artifact is being signed?
2. Where did it come from?
3. Why does it require a signature?

This clarity becomes essential during audits, incident response, and customer inquiries. Without it, teams struggle to demonstrate software integrity or explain signing decisions after the fact.

Automated trust workflows make this easier by tying signing decisions to policy and pipeline context rather than individual judgment. This creates consistency and traceability across teams and tools.

2. Clearly Define Which Artifacts Must Be Signed

Inconsistency is one of the most common sources of friction. Some teams sign everything. Others sign only final binaries. Some rely on manual steps, while others automate selectively. Over time, this leads to gaps, duplication, and confusion.

A best practice is to explicitly define, document, and enforce which files and artifacts must be signed through a formal code signing policy that applies consistently across teams, tools, and pipelines. This helps avoid two common failure modes:

- Under-signing, where critical artifacts are released without protection
- Over-signing, where unnecessary files increase operational overhead

Clear definitions allow automation to work as intended. When expectations are standardized, pipelines can enforce signing rules reliably and at scale.

3. Centralize Signing Through CI/CD Pipelines Only

Signing should not be an exception-based activity. It should be part of the pipeline.

Conversations with customers consistently show that ad hoc and manual signing introduces delays, increases risk, and complicates audits.

Nearly 1/3 of surveyed teams report relying on manual processes, even as release frequency continues to increase.

Centralizing artifact signing through CI/CD pipelines helps ensure that:

- Signing happens the same way every time
- Developers do not need direct access to signing keys
- Off-process or unauthorized signing is reduced by design

This approach supports fast-moving teams while giving security leaders confidence that signing controls are applied consistently across environments.

4. Automate Certificate Renewals for Production Releases

Shorter certificate validity periods amplify existing weaknesses in renewal processes. Manual renewals do not scale, and calendar reminders are not a reliable control.

Survey data shows a clear execution gap. Only 17% of respondents have fully automated certificate renewal processes. The majority are still relying on manual steps or are in early-stage planning that has yet to reduce operational overhead.

More mature teams treat renewals as a background process rather than a scheduled event. Policy-driven lifecycle management allows certificates to renew automatically before expiration without requiring changes to pipelines or developer workflows.

In this model, expiration risk is significantly reduced, and releases continue without interruption, regardless of certificate lifespan.

5. Minimize Human Action Across Signing and Renewal Workflows

Every manual step is a potential failure point. Human involvement introduces delays, errors, and reliance on institutional knowledge that doesn't scale across teams or time zones.

Minimizing human action does not mean removing oversight. It means shifting control from people to policy. Automated workflows enforce rules consistently, while approvals and exceptions are handled deliberately and audibly when required.

This approach reduces bottlenecks, improves reliability, and allows security teams to focus on higher-value work instead of routine maintenance.

6. Maintain Signature and Audit Logs for Traceability

Visibility matters, especially when something goes wrong.

Signature and audit logs provide a verifiable record of:

- What was signed—successfully and unsuccessfully
- When it was signed
- Who initiated or approved the action
- Which key and certificate were used
- The originating system details, such as IP address

These records are essential for compliance, incident response, and internal accountability. They also allow teams to answer questions quickly instead of reconstructing events across disconnected systems.

In regulated or high-risk industries, this level of traceability is no longer optional.

7. Use Timestamping to Preserve Long-term Trust

Signing does not end at release. Many organizations need software to remain trusted for years, long after the original signing certificate has expired.

Timestamping allows signed artifacts to validate based on when they were signed rather than the current status of the certificate. This relies on a trusted Time Stamping Authority to cryptographically bind the signature to a verified point in time. This is especially important for long-lived software, archived releases, and extended support scenarios.

When combined with automated signing and renewal workflows, timestamping helps maintain trust over time without adding operational complexity.

Putting Policy-Driven Code Signing into Practice

Many teams already have tools in place to sign code and protect keys. The challenge is that those tools are often underutilized or loosely integrated, leaving signing disconnected from policy, visibility, and delivery workflows.

A more scalable approach treats code signing as a governed, automated control, rather than a standalone task. This typically includes:

- Standardizing signing rules across pipelines and teams
- Enforcing policy through CI/CD automation instead of manual steps
- Automating renewals and approvals to reduce operational load
- Using centralized logs and timestamps to preserve trust over time

DigiCert® Software Trust Manager is designed to help organizations move from isolated signing activities to a policy-driven approach that scales across teams, tools, and release pipelines. .

A useful question to ask is not “Are we compliant?” but
“Is our signing process doing as much work for us as it should?”

Talk to a DigiCert expert to review your current approach and identify where policy-driven automation can reduce risk and operational effort without slowing delivery.

About DigiCert

DigiCert is a global leader in intelligent trust. We protect the digital world by ensuring the security, privacy, and authenticity of every interaction. Our AI-powered DigiCert ONE platform unifies PKI, DNS, and certificate lifecycle management, to secure infrastructure, software, devices, messages, AI content and agents. Learn why more than 100,000 organizations, including 90% of the Fortune 500, choose DigiCert to stop today’s threats and prepare for a quantum-safe future at www.digicert.com.