digicert®

Evaluating Signing Software & Adopting Code Signing Policy

Best practices to consider in order to achieve stronger security in the software release process—without slowing down the CI/CD pipeline.

BUYERS GUIDE

Introduction

Software supply chain attacks and other forms of malware propagation can damage customer trust and be time-consuming and costly to remediate. These high-profile attacks can result in data breaches, ransomware demands, and full system shutdowns.

In tandem, the points of vulnerability in software security have proliferated. Agile development has increased the frequency of build cycles. Cloud deployment and orchestration have shifted the seat of software delivery. And, the proliferation of connected devices calls for secure firmware updates and communication.

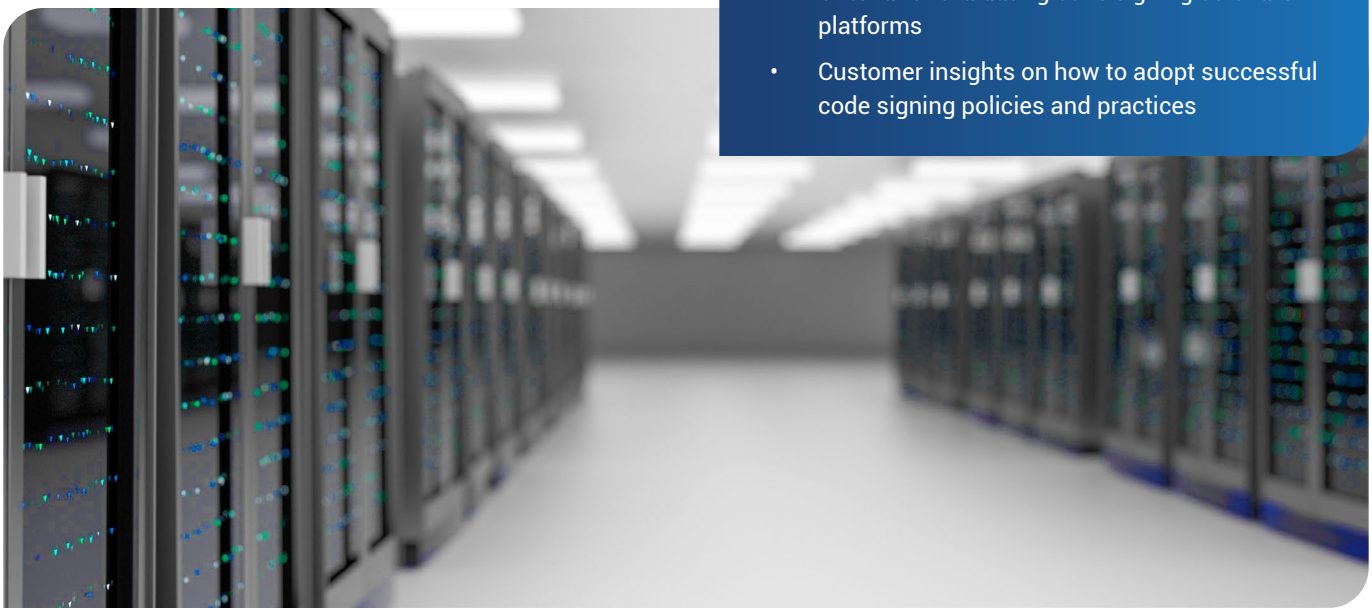
Now, more than ever, companies need to improve the security posture of their software development lifecycles. Code signing software centralizes the management of code signing activity, delivering standardization and enforcement of the code signing policies and practices that reduce software vulnerabilities.

When evaluating and implementing code signing software solutions, these criteria should guide your decisions:

1. Flexibility in configuration
2. Comprehensive key storage and handling methods
3. Integration and automation of workflows
4. Centralized and flexible reporting
5. Release process controls
6. Flexible and scalable deployment

This buyer's guide is for customers who want to improve their software release security, presenting:

- Criteria for evaluating code signing software platforms
- Customer insights on how to adopt successful code signing policies and practices



Flexibility in Configuration

Fine-grained configurability of account structures and user roles and permissions allows companies to map system functions to their unique organizational requirements.

Objectives

- Centralize management of the enterprise-wide signing key and certificate landscape
- Map account structures and functions according to security needs
- Restrict system access to designated users
- Eliminate single points of vulnerability for critical functions
- Enforce zero-trust policy for account access
- Improve efficiency in assigning and revoking system privileges

What to Look For

- Import of keys and certificates for consolidating the full key and certificate landscape
- Central account configuration and user structures, including setup of ICA and ICA attributes
- Granular user roles and permissions with access policy controls
- Quorum permissions for functions requiring two-party approval
- Support for multi-factor authentication
- Group level profiles

Code Signing Policy & Practices Tip #1

Customers reported different methods of pairing code signing policy with code signing software use. Some reported creation of a formal policy document. Others relied on guidance and training to guide code signing practices, and others relied on software use for centralized enforcement of policy.

Code Signing Policy & Practices Tip #2

Customers emphasized the following as the top three key handling concerns: who manages keys, where keys are stored, and definition and enforcement of signing permissions.

Comprehensive Key Storage and Handling Methods

Fine-grained key storage and handling controls allow companies to eliminate cross-organization inconsistencies in key handling practices and prevent unauthorized or errant access and usage.

Objectives

- Select key storage method that best fits security need (e.g., public or private trust)
- Implement tighter controls for sensitive projects
- Limit access when investigating possible threats
- Separate key usage practices for production and test environments
- Apply the appropriate key usage model to the requirements of the target software platform
- Reduce use of keys with outdated, weak, or non-compliant cryptography elements
- Increase efficiency and reduce human error

What to Look For

- Key storage options for Hardware Security Modules (HSMs) or encrypted storage
- Key access profiles that govern who can access what keys when Online and offline signing key modes, to allow for rapid response
- Production and test signing key types
- Static, dynamic, and rotating key usage models
- Granular keypair profile controls
- Certificate profile templates with 1-click generation workflows

Code Signing Policy & Practices Tip #3

Customers reported business/legal compliance and customer and platform requirements as the primary drivers for the development of code signing policies and practices.

Integration and Automation of Workflows

Integration and automation prevents manual errors, improves efficiency, drives consistency in signing practices, and supports the time-to-market goals of agile development.

Objectives

- Provide security without slowing down the development process
- Achieve higher levels of corporate security and compliance with increased convenience for users compared to manual signing processes
- Minimize the risk of source code interception during the signing process

What to Look For

- Integration with CI/CD pipelines, with command-line, console, or API options
- Support for multiple CI/CD platforms
- Centralized and preconfigured access and privileges for developers and build servers
- Preconfigured certificate templates and certificate generation
- Hash signing, eliminating the need for transfer of source code

Code Signing Policy & Practices Tip #4

Customers reported their code signing policy, enforced through code signing software and with governance by a designated security or IT organization, as being very effective.

Release Process Controls

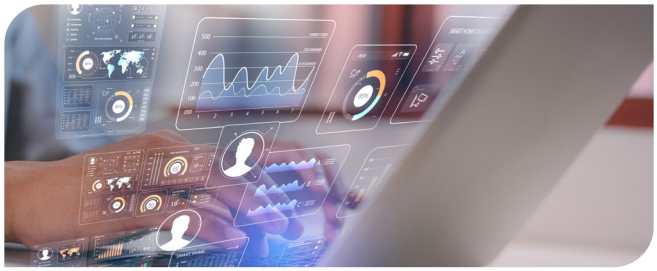
Release process controls provide additional checks that can help prevent the injection of malware during the build process itself.

Objectives

- Reduce opportunities for malicious activity during release process

What to Look For

- Release window attributes and controls that define user access, scheduled windows, and other criteria that limit access to authorized users
- Release comparison features that use the principle of reproducible builds to define a "gold standard" that is then used as a point of comparison with subsequent builds



Centralized and Flexible Reporting

Centralized reporting helps speed remediation, comply with audit requirements, and identify possible problems or threats.

Objectives

- Full visibility and control of code signing activities
- Know who signed what when, improving speed of remediation
- Provide audit compliance

What to Look For

- Centralized logging and reporting, supporting auditing and tracking activity
- Report filters, with export in raw and formatted file types

Code Signing Policy & Practices Tip #5

Customers noted that both security and IT teams were key participants in the creation of their code signing policies. Contributors noted by customers included the CISO, compliance and risk management, DevOps, SecOps, product management, and the cybersecurity and IT teams.

Flexible and Scalable Deployment

Companies should look for choice in deployment model, in order to be able to cover different deployment needs within their organization, and assess the scalability of solutions against their anticipated signing requirements.

Objectives

- Ability to be deployed in the preferred manner
- Ability to support high volume and throughput of signing activity
- Efficient setup with on-demand scalability

What to Look For

- On-premises, cloud, or hybrid options to address comprehensive delivery needs throughout the organization
- Fast and efficient setup of ICAs for divisions with specific security configuration requirements
- Container-based architecture with orchestration for fast and efficient scaling

Summary

Code signing software can significantly reduce the points of vulnerability in software release processes. Enterprise-grade solutions that are flexible in their configurability and comprehensive in their feature set are best suited to deliver against the goals of a company's code signing policies and practices. These solutions deliver:

Protection

- Eliminate points of vulnerability that can lead to theft or misuse of keys
- Prevent injection of malware into the release process

Management

- Drive consistency in security posture and compliance across build teams
- Centralize visibility and control over signing activity

Productivity

- Improve efficiency and reduce human error
- Create agility in cryptographic changes and compliance

Code Signing Policy & Practices Tip #6

Customers noted that standardization of the code signing process, making it documented, traceable, accountable, and secure was important, as was guaranteeing the source and integrity of the company's software packages to downstream customers.

Ready To Get Started?

DigiCert provides enterprise-class SSL, PKI and IoT security solutions for some of the world's biggest organizations — providing peace of mind and secured data at all times. Talk to our experts about your needs.

For more information, call 1.801.770.1736,
email pki_info@digicert.com,
or visit digicert.com/software-trust-manager