Enterprise Code Signing

Securing the Supply Chain at Scale

Background

Software is increasingly embedded in every facet of business operations. Even if you're not a software company, internal applications, automation scripts, device firmware, and packaged releases still need to be trusted by users, devices, and digital storefronts.

Code signing is the mechanism that makes software verifiable. It allows recipients to validate that a binary or package came from an approved publisher and has not been modified.

The Problem

- Software must be secured to be trusted by users, devices, operating systems, and distribution platforms
- Regulations and industry standards increasingly mandate not only that software is signed, but also how signing keys are protected and how signing operations are governed.
- Attackers have shifted from compromising code to compromising signing keys—because a valid signature can make malicious software look legitimate.

The Solution

Organizations should integrate code signing directly into the CI/CD pipeline to ensure that every build and release is signed consistently as part of the delivery workflow. A modern approach also centralizes signing keys and enforces strong access controls—such as MFA and just-in-time permissions—to reduce key exposure and support audit readiness.

What Is It:

A **SaaS-based** code signing service that protects signing keys, integrates with modern development workflows, and simplifies compliance.

Who It's For:

Any organization that creates software and has concerns around **security**, **compliance**, and **audits**.

Why It Matters:

Code signing is increasingly **required** by platforms, customers, and regulators—and attackers are actively targeting signed keys.

Powered by Garantir

HID

Code Signing Overview

What is Code Signing?

Code signing uses public key cryptography to attach a verifiable digital signature to software. When software is signed, recipients can validate several things:

1. Who published the code
2. When the code was signed
3. Whether the code was altered after signing

How it Works

- A signing key pair is created. The private key must remain confidential and strongly controlled.
- When a release artifact is ready, a signing request is made manually or via automation.
- The signing service produces a signature using the private key without exposing it.
- The signed artifact is distributed. The signature is validated against the public certificate.

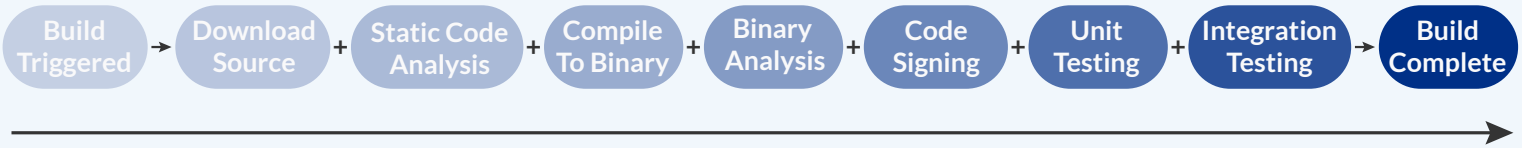

Figure 1: Code Signing Build Process

Why Traditional Implementations Break Down

Many organizations start code signing with local keys, ad-hoc scripts, or a single HSM deployment. As signing becomes more central to release processes and compliance, these approaches often fail to scale.

- **Key proliferation & risk:** Keys end up copied across build servers, developer machines, and scripts.
- **Integration complexity:** Teams struggle to to integrate signing into CI/CD, IDEs, and release management tools without slowing delivery.
- **Pipeline velocity:** Security controls can become bottlenecks, especially at scale.
- **Operational overhead:** Building and maintaining HSM infrastructure, policies, and audits is expensive and time consuming.
- **Compliance burden:** Regulations increasingly require strong controls around key protection, access, and logging—not just the existence of signatures.

The HID Solution

HID Code Signing is a modern, SaaS-based code signing service designed to integrate into real development workflows while meeting the security and compliance expectations of today's software supply chain.

Core Capabilities

- **Drop-in workflow support:** Integrates with CI/CD, IDEs, and release management tools without forcing a redesign of your pipeline.
- **Automated and manual signing:** Supports both CI-driven signing and developer-initiated signing from workstation tools.
- **Remote signing with non-exportable keys:** Keys remain protected while signing requests are served securely.
- **Robust access control:** Enforce MFA, device trust, and just-in-time access for signing operations.

Integrations

The service is designed to be tool-agnostic and compatible with modern engineering environments. Typical integration points include CI/CD systems (e.g. GitHub CL, Jenkins, Azure DevOps), IDE tooling, and release orchestration systems.

When paired with HID's PKIaaS platform, HID Code Signing becomes a complete end-to-end solution for certificate lifecycle management and signing governance.

Common Signing Use Cases

- Windows code signing (Authenticode)
- Apple / OSX signing
- Container and artifact signing (e.g. OCI images)
- Java & Android signing
- Linux package signing

Security by Design

- **FIPS-compliant key protection:** Designed to meet modern expectations for cryptographic key security.
- **Reduced attack vectors:** Centralized remote signing minimizes key sprawl across build infrastructure.
- **Fast access allocation and revocation:** Security teams can quickly grant or remove signing privileges as roles change.

Compliance & Audit Readiness

- **Centralized logging and notifications:** Visibility into signing operations across teams and pipelines.
- **SIEM/SOAR integration:** Exportable events and logs for detection, alerting, and response workflows.
- **Future-ready:** Includes support for both post-quantum and classical algorithms (e.g. ML-DSA, SLH-DSA, and SHA-256).
- **Crypto agility:** Quickly change algorithms when rotating or renewing signing keys to stay ahead of emerging requirements.
- **Designed for modern software supply chain standards:** Helps organizations align with CA/B Forum expectations, SLSA, CNSA 2.0, and NIST requirements.

Key Benefits

- **SaaS deployment:** No large up-front investment, long infrastructure projects, or complex rollouts.
- **Right-sized licensing:** License only the signing volume you need.
- **Faster time-to-value:** Deploy quickly and integrate into existing workflows with minimal disruption.
- **Lower operational cost:** Avoid the ongoing burden of building, operating, and auditing dedicated signing infrastructure.

The Bottom Line

HID Code Signing makes it practical to treat code signing as a scalable security control—not a fragile build-time script or long-term infrastructure project.

Next Steps

Whether your organization is introducing code signing for the first time or strengthening controls around existing signing workflows, HID can help you deploy a secure, compliant approach that supports developer velocity.

To learn more, contact us at hidglobal.com/contact or +1 (512) 776-9000