digicert®

SICHERHEITSSTEUERUNGEN FÜR DURCHGÄNGIGES CODE-SIGNING

Die Zahl der Angriffe auf Softwarelieferketten ist in jüngerer Zeit so stark gestiegen, dass Unternehmen die Sicherheit des gesamten Softwareentwicklungslebenszyklus (SDLC) auf den Prüfstand stellen sollten. Manuelles Code-Signing kann zu inkonsistenten Prozessen und Schwachstellen führen, die von Hackern ausgenutzt werden können. DigiCert® Secure Software Manager stärkt die Sicherheit durch die konsequente Nutzung von Schlüsseln und Zertifikaten, die den geschäftlichen Anforderungen gerecht werden, und bietet nahtlose und konsistent robuste Sicherheit beim Code-Signing.

Die Sicherheitsfunktionen des DigiCert® Secure Software Manager

- Steuerungen für die Kontenverwaltung
- Zugriffs- und Nutzermanagement
- Steuerungen für Schlüssel- und Zertifikatsicherheit
- Steuerungen für Software-Release-Sicherheit
- Verfolgung und Berichterstellung
- Automatisierung

Steuerungen für die Kontenverwaltung

Steuerungen für die Kontenverwaltung unterstützen das Durchsetzen von Sicherheitsrichtlinien, die Trennung von Rollen und Aufgaben sowie die Standardisierung von Best Practices für die Schlüsselsicherheit, um Diebstahl und Missbrauch vorzubeugen.

Wichtige Funktionen	Vorteile
Zentrale Kontenkonfiguration	Flexibilität bei der Konfiguration von Kontofunktionen, -steuerungen und Nutzerstrukturen zur Erfüllung der Sicherheitsanforderungen des Unternehmens.
Granulare Zugriffskontrollen	Verhindert, dass Nutzer unbefugt Schlüssel oder Zertifikate erstellen, die nicht mit ihrer Arbeit verbunden sind.
Steuerungen auf Kontenebene	Sorgen auf Kontoebene dafür, dass beim Generieren von Schlüsselpaaren den Richtlinien entsprechende Algorithmen, Schlüssellängen und Kurven genutzt werden.
Import anderer Schlüssel	Sorgen durch die Konsolidierung und Verwaltung des gesamten Inventars von Signierschlüsseln und Zertifikaten für Transparenz und Kontrolle.

Zugriffs- und Nutzermanagement

Das Zugriffs- und Nutzermanagement stellt sicher, dass nur berechtigte Nutzer auf die betreffenden Systemfunktionen zugreifen können und reduziert somit die Zahl der versehentlichen und absichtlichen Richtlinienverstöße.

Wichtige Funktionen	Vorteile
Granulare Nutzerrollen und Berechtigungen	Definieren die Zugriffsrechte der verschiedenen Nutzerrollen und unterstützen somit die Pflichtentrennung. Zugriffsrechte können rasch (und bei Bedarf auch zeitlich begrenzt) gewährt, ausgesetzt und zurückgezogen werden.
Quorumberechtigungen	Vermeiden den Missbrauch kritischer Funktionen, indem sie für bestimmte Aktivitäten eine Betätigung durch zwei Nutzer erfordern
Profile auf Gruppenebene	Steigern die Effizienz, da die Berechtigung zur Nutzung und zum Generieren von Schlüsselpaaren und Zertifikaten ganzen Nutzergruppen erteilt und entzogen werden kann.
Multifaktor-Authentifizierung (MFA)	Setzt Zero-Trust-Prinzip für den Kontozugriff durch.

Steuerungen für Schlüssel- und Zertifikatsicherheit

Ausgereifte Steuerungen für die Schlüssel- und Zertifikatsicherheit reduzieren Diebstahl und Missbrauch und bieten bessere Sicherheit beim Umgang mit Schlüsseln.

Wichtige Funktionen	Vorteile
Profile für den Schlüsselzugriff	Unterstützen mehrere Anwendungsbereiche mit „offenen“ Zugriffseinstellungen für Signierberechtigte und „eingeschränkten“ Zugriffseinstellungen für strengere Sicherheitsbeschränkungen.
Optionen für die Schlüsselspeicherung	Bieten Optionen für die Schlüsselspeicherung für öffentliche und private Trust-Systeme, einschließlich der Speicherung in Hardware-Sicherheitsmodulen (HSM) für öffentliche Schlüssel.
Verschiedene Schlüsselarten für Produktion und Test	Definieren verschiedene Schlüsselarten für Produktion und Test. Produktionsschlüssel laufen nie ab und werden zum Signieren von öffentlichem und privatem Binärcode sowie zur Erzeugung neuer Zertifikate verwendet. Testschlüssel sind nur 30 Tage lang gültig und werden bei Bedarf für die interne Veröffentlichungsvalidierung genutzt.

Statische, dynamische und rotierende Schlüsselnutzungsmodelle	Wählen aus mehreren Schlüsselnutzungsmodellen ein den Sicherheitsanforderungen der jeweiligen Softwareplattform genügendes Modell aus: • Android: Statische Schlüssel, die für alle Anwendungs-Releases denselben Schlüssel bzw. dasselbe Zertifikat verwenden • Java & IoT: Dynamische Schlüssel mit einer 1:1-Beziehung zwischen Schlüssel und Signatur • Microsoft Smartscreen Filter: Rotierende Schlüssel aus einem Pool von Zertifikaten
Online- und Offline-Signiermodi	Ermöglichen strengere Sicherheitssteuerung für sensible Projekte oder die Bedrohungsuntersuchung. Im Online-Modus können Schlüssel zu jeder Zeit von jedem berechtigten Nutzer verwendet werden. Im Offline-Modus wird die Schlüsselnutzung verhindert, bis Schlüssel wieder im Online-Modus sind oder ein geplantes Release-Zeitfenster eingerichtet ist.
Vorlagen für Zertifikatsprofile und Workflows für die Zertifikatserzeugung	Verbessern die Compliance, sparen Zeit und senken die Fehlerrate. Attribute können vorab definiert, als Vorlage für Zertifikatsprofile gespeichert und in Workflows für die Zertifikatserzeugung mit einem Klick auf der Benutzeroberfläche oder in einem Befehl oder Skript aufgerufen und genutzt werden.
Granulare Steuerungen für Schlüsselpaarprofile	Zentralisieren die Verwaltung von Schlüsselpaarprofilen, steigern die Flexibilität bei der Verschlüsselung und reduzieren die Nutzung von Schlüsseln mit schwachen oder nicht vorschriftsgemäßen Verschlüsselungselementen. Für Schlüsselpaarprofile mit verschiedenen Algorithmen, Schlüssellängen und Kurve können unterschiedliche Steuerungen definiert werden.

Steuerungen für Software-Release-Sicherheit

Steuerungen für die Software-Release-Sicherheit stärken die Sicherheit während des Release-Prozesses und tragen so zur Vereitelung von Angriffen auf die Softwarelieferkette bei, die auf interne Entwicklungsprozesse abzielen.

Wichtige Funktionen	Vorteile
Release-Zeitfenster	Bieten weniger Gelegenheiten für schädliche Aktivitäten während des Veröffentlichungsprozesses. Dazu werden die folgenden Attribute eines Release vorab definiert: befugte Nutzer, Schlüsselarten (z. B. online, offline oder Test), Zahl der Binärdateien, Anfangs- und Enddatum und -zeit, vorab genehmigte Signierzeitfenster sowie die Metadaten des Release. Die getrennte Einrichtung und Genehmigung unterstützt Richtlinien, die Aufgabentrennung fordern.
Release-Vergleich	Bestätigt, dass während des Release-Prozesses keine Malware eingeschleust wurde. Dazu wird ein Build als Standard definiert und dient dann als Muster dafür, was im Produktionsrelease signiert werden darf. Dies entspricht dem Prinzip reproduzierbarer Builds und bietet einen unabhängig verifizierbaren Weg vom Quellcode zum Produktbinärcode.

Durchgängige Verfolgung und Berichterstellung

Eine zentralisierte, durchgängige Verfolgung und Berichterstellung unterstützt die Bedrohungsanalyse und -erkennung, die schnelle Reaktion auf schädliche Aktionen oder Fehlermeldungen sowie Zertifizierungen und Audits gemäß den Branchenstandards.

Wichtige Funktionen	Vorteile
Zentralisierte Protokollierung und Berichterstellung zur Unterstützung von Audits und zur Nachverfolgung	Bietet einen vollständigen Überblick und Kontrolle über Code-Signing-Aktivitäten, da alle Aktivitäten in Verbindung mit Konten, Schlüsselpaaren, Zertifikatsvorgängen und Signaturen detailliert protokolliert werden. Sowohl erfolgreiche als auch fehlgeschlagene Vorgänge werden festgehalten, um Informationen für spätere Korrekturmaßnahmen bereitzustellen. Berichte können gefiltert und als formatierte beziehungsweise Rohdaten sowie über APIs exportiert werden.

Automatisierung

Automatisierung verhindert manuelle Fehler, verbessert die Effizienz und fördert eine konsistente Signierpraxis.

Wichtige Funktionen	Vorteile
Integration in CI/CD-Vorgänge (Continuous Integration/ Continuous Delivery)	Automatisiert Signierprozesse und bietet so Sicherheit, ohne die agile Entwicklung zu bremsen. Direkte Integration in die gängigsten CI/CD-Plattformen Bibliotheken auf DigiCert-Clients können automatisch über Skripte aufgerufen werden.
Zentralisierung und Vorkonfigurierung von Zugriff und Privilegien für Entwickler und Build-Server	Erleichtert die Einhaltung von Sicherheitsanforderungen und die Nutzung mit vorkonfigurierten Steuerungen, die die Signierung ohne direktes Eingreifen ermöglichen. Diese Steuerungen können auf einer zentralen Konsole eingestellt und in ebenfalls vordefinierten Zeitabständen gestartet werden. Build-Server können als API-Nutzer konfiguriert werden, sodass Signieranforderungen automatisch bearbeitet werden können.
Hash-Signing	Reduziert das Risiko, dass Quellcode während des Signierprozesses abgefangen wird, da binärer Quellcode die Entwicklungsumgebung nicht verlässt. Anstelle vollständiger Binärdateien werden nur Hashdateien zum Signieren hochgeladen. Dadurch werden sowohl die Größe der übermittelten Dateien als auch die Latenz reduziert.

Weitere Informationen sowie einen kostenlosen Test können Sie unter +1 801 770 1736, pki_info@digicert.com oder digicert.com/de/secure-software-manager anfordern.

© 2022 DigiCert, Inc. Alle Rechte vorbehalten. DigiCert ist eine eingetragene Marke von DigiCert, Inc. in den USA und in anderen Ländern. Alle anderen Marken und eingetragenen Marken sind Eigentum der jeweiligen Inhaber.