Compliance Program Management: Getting the Most From GRC Platforms

Bev Waller & Erik Jones February 25, 2026

Executive Summary

Most growth-stage SaaS companies adopt a GRC (Governance, Risk, and Compliance) platform expecting it to run their compliance program. The platform does not run the program; it organizes the artifacts the program produces. This guide explains the operating model that makes a GRC platform pay back its cost — and the conditions under which the platform is the wrong purchase entirely.

Why Compliance Program Management Matters

Compliance is no longer a single-framework exercise. A typical B2B SaaS at Series B holds **SOC 2 Type II**, prepares for **HIPAA** as healthcare customers come on, considers **ISO 27001** for international

expansion, and faces **PCI DSS** if it touches payment data. The frameworks share **60-80% of their controls** at the technical level (access control, encryption, monitoring, change management). The discipline is to implement those shared controls once and map the evidence to multiple frameworks — not to run four parallel programs that produce four parallel evidence packages.

The discipline is called **crosswalking**: maintaining a control register where each control is mapped to its corresponding requirement in every applicable framework. As the Cloud Security Alliance's *"Cloud Controls Matrix"* demonstrates, *"a single set of well-designed cloud controls can satisfy the majority of requirements across SOC 2, ISO 27001, NIST CSF, FedRAMP, and HIPAA simultaneously."*

The GRC Platform Question

GRC platforms (collectively — without naming names at this level) automate three functions: **continuous controls verification**, **evidence collection**, and **auditor collaboration**. Each is genuinely valuable; none replaces the underlying program.

What Platforms Do Well

Continuous Controls Verification

Integrations with AWS, Okta, GitHub, Jamf, and the major SaaS tenants pull control state continuously. Drift is flagged immediately

rather than discovered during audit fieldwork. The category-defining feature of the modern GRC platform.

Evidence Collection

Automated screenshots, exports, and configuration snapshots pulled on a schedule. Auditors get a folder of dated artifacts instead of last-minute manual collection. The single biggest time-savings on the audit calendar.

Framework Crosswalk

Pre-built control libraries mapped to SOC 2, ISO 27001, HIPAA, PCI DSS, NIST CSF, and many others. Implement a control once; the platform displays it under every framework that requires it.

What Platforms Do Not Do

Decide what controls you need: the platform shows you the gaps; closing them is engineering and policy work.

Build the controls: automated evidence presupposes there is something to evidence. The platform does not configure your IAM, write your runbooks, or build your access-review process.

Train your auditors: the platform organizes evidence, but the auditor still asks questions, samples controls, and writes the opinion. Auditor relationship management remains human work.

Stop control drift: the platform reports drift; remediating it is operational work the platform cannot perform.

Run risk assessments: templates exist but the analytical work is yours.

The Wrong Purchase

A GRC platform without a compliance lead is shelfware. Without an owner who runs the program, the platform fills with stale evidence, ignored alerts, and orphan controls. The license cost stays high; the audit cost stays the same as before.

The Operating Model

A working compliance program has the same shape regardless of platform choice.

Phase 1: Frameworks and Scope

Decide which frameworks apply now and which apply within 18 months. Decide scope per framework — which products, which environments, which data classifications. The decisions drive every subsequent control choice.

Phase 2: Crosswalk and Control Register

Build a unified control register where each control has:

A control identifier and description

A mapping to every applicable framework requirement

An owner (named human or named team)

An evidence source (automated where possible, manual cadence where not)

A test procedure the auditor can replicate

The register is the single source of truth. Frameworks are views into the register, not separate programs.

Phase 3: Implementation

Build the controls. Identity, endpoint, logging, vulnerability management, vendor risk, incident response, change management. Most of the work is the same regardless of which framework triggered it.

Phase 4: Continuous Verification

Wire up the platform (or the alternative). Continuous controls verification, automated evidence collection, anomaly alerting. The platform is most valuable here, after the controls exist.

Phase 5: Audit Readiness

Mock audits, evidence sampling, remediation cycles, gap closure. Internal audit function reviews the platform's evidence quarterly to catch issues before external auditors do.

Phase 6: External Audit

Auditor walks the controls, samples evidence, conducts interviews, writes the opinion. The platform shortens this phase from months to weeks.

The Crosswalk in Practice

A representative control: *"Production access requires approval from someone other than the requester."*

Framework Mappings

SOC 2 CC6.1, CC6.3: logical access controls

ISO 27001 A.5.15, A.8.3: access control, information access restriction

HIPAA 164.312(a)(1): access control standard

PCI DSS 7.2: access control system

NIST 800-53 AC-3, AC-6: access enforcement, least privilege

One Implementation

SSO with MFA via Okta or Azure AD; IAM Identity Center for AWS access; just-in-time elevation through **AWS IAM Identity Center** or third-party tools; PR-based change approvals through GitHub; access reviews on a quarterly cadence.

One Evidence Stream

Okta logs, IAM Identity Center session logs, GitHub PR history, quarterly access review records. The same artifacts satisfy all five framework requirements.

Common Pitfalls

Platform first, program second: buying the platform before defining the operating model produces shelfware.

Per-framework programs: running SOC 2 and ISO 27001 as separate workstreams duplicates 70% of the effort.

Generic control libraries: taking the platform's default control list verbatim ignores the parts that do not apply to your environment and miss the parts that do.

Auditor handoff at the platform: assuming the platform's evidence is auditor-ready without an internal review cycle catches issues only at fieldwork, when remediation is expensive.

No control owners: a control without a named owner has no owner. When the platform flags drift, nobody fixes it.

Misaligned cadence: annual access reviews on monthly access changes accumulate drift faster than the review catches it.

Measured Outcomes

One control register mapped to all applicable frameworks

60-80% control overlap across SOC 2, ISO 27001, HIPAA, PCI DSS — implemented once

Continuous evidence collection reducing audit fieldwork from months to weeks

Quarterly internal audits catching drift before external auditors do

Documented control ownership per control, per framework

How Jacobian Helps

The compliance program is the operating model; the GRC platform is one tool that supports it. Our team designs the operating model — frameworks, scope, control register, crosswalk, ownership — and then helps you choose whether a platform pays back its license cost in your environment. We pair well with whatever platform you bring; we run with no platform when the team and the cadence are sufficient. Either way, the controls are the same, the evidence is auditor-ready, and the program runs continuously rather than in audit-week panic mode.

Resource Details

Author: Bev Waller & Erik Jones

Published: February 25, 2026

Categories:

Compliance Security GRC

Download

Full Document

About This Resource

A practical guide to running multi-framework compliance programs (SOC 2, HIPAA, ISO 27001, HITRUST). Covers control crosswalks, evidence automation through GRC platforms, and how to decide whether you need a platform at all.

Categories:

Compliance Security GRC