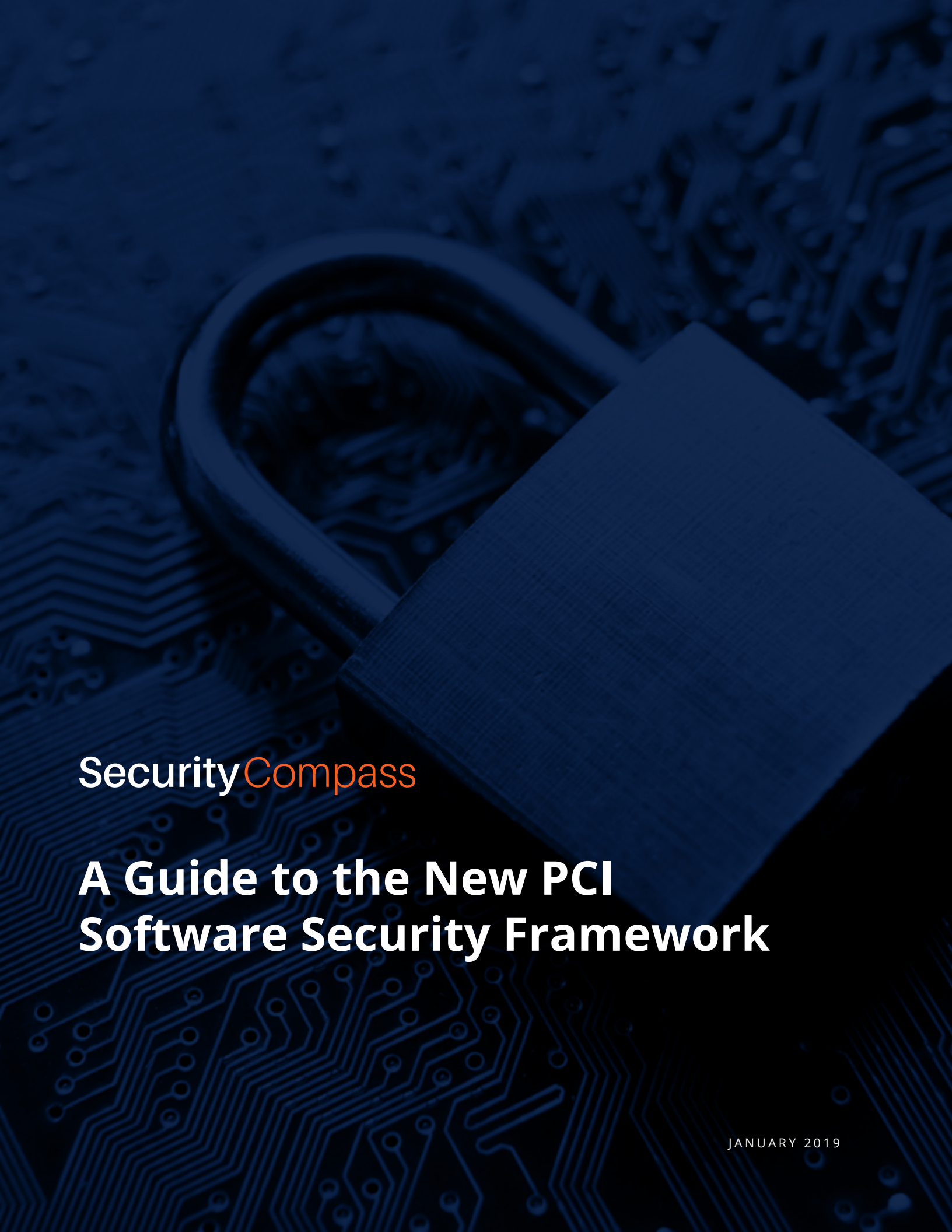


SecurityCompass

A Guide to the New PCI Software Security Framework

JANUARY 2019

Table of Contents

Introduction

Since its formation in 2006, the PCI Security Standards Council (SSC) has greatly evolved. Currently, the PCI SSC manages 12 standards, including the PCI Data Security Standard (DSS), which was unprecedented in its purpose– to protect against sophisticated forms of fraud in the burgeoning e-commerce industry. Since the council's formation, a number of new technologies have emerged, significantly changing payment solutions and payment data security industries. To keep up with these new payment technologies, the PCI DSS regularly updates their security programs and standards.

On January 16th, 2019, the PCI SSC released 2 new PCI Software Security Standards as part of the new PCI Software Security Framework. The PCI SSC developed the new Software Security Standards by forming a Task Force, including industry experts representing from Security Compass, Microsoft, and other organizations.

The framework represents an effort to create a higher caliber of software security in the payments ecosystem, now supporting validation programs for software products and qualification programs for software vendors. Nearly all other information security standards that have preceded the PCI Software Security Standards have been higher-level and, hence, less focused on particular software security details. Nevertheless, the release of the new standards is much needed. Few companies today describe themselves as being sufficiently mature to adhere to a secure software development lifecycle (SDLC) framework, and other industries with mission-critical applications, like nuclear and IoT, may go on to be dangerously liberal with their software security.

The following guide offers a comprehensive overview of the standards in the new PCI Software Security Framework. We review the new standards and offer our own software security solution for compliance. We also discuss our future plans for helping organizations fully align with the new PCI Software Security Standards.

Overview of the PCI Software Security Framework

The new PCI Software Security Framework was built with the understanding that, in order for payment software to be considered secure, it must first be developed and maintained in a way that protects the integrity of payment transactions and the confidentiality of all sensitive data collected in association with payment transactions. The PCI SSC's primary goal is to provide a way to secure payment applications that can support current as well as future industry technologies and best practices. The existing PCI PA DSS, which covers traditional payment applications, will eventually be integrated into the new PCI Software Security Standards.

The standards[1] are presented in 2 documents:

- The **Secure Software Requirements and Assessment Procedures:** A rigorous standard that relies on in-depth security testing techniques to validate whether a software release is compliant. This is also referred to simply as the ***"Secure Software Standard."*** [2]

- The **Secure Software Lifecycle (Secure SLC or SSLC) Requirements and Assessment Procedures:** An optional standard that assesses security throughout the software development and operations lifecycle. By complying with the Secure SLC requirements, organizations can forgo the need to have each release assessed by a qualified assessor, better enabling modern agile and continuous delivery software practices. This is also referred to simply as the ***"Secure SLC Standard."*** [3]

Mid-year 2019, the PCI SSC expects to release a third component in the Software Security Framework, called the '**Validation Program**.' This is a program for software vendors to validate how they can properly manage the security of payment software throughout the entire software lifecycle. [4]

The PCI Software Security Standards were released in an initial draft to stakeholders for their feedback. The creators declared two Request for Comment periods, which were offered to stakeholders since the new framework would have an impact on their respective industries. All PCI-Recognized Labs, Payment Application Qualified Security Assessors, and PCI SSC Participating Organizations were asked to review and comment on the draft during the Request for Comment periods.

The PCI Software Security Standards' coverage is comprehensive, addressing all payment software functionality as well as the identification and implementation of security controls. The new standards outline the responsibility of outside vendors in guiding customers' security practices. They also identify the necessary tools and software functions required to access critical assets, referencing execution environments, code libraries, requirements, and dependencies.

Who Will Be Affected By The New PCI Software Security Framework?

Initially, the new standards will affect vendors or providers of Payment Applications (PA), rather than those companies that procure and deploy PA for their e-commerce needs. Payment processor companies have historically been subject to compliance with the Payment Application Data Security Standard (PA-DSS). The new standards are an extension of this, with the addition of prescriptive instructions about how software ought to be secured. The new standards have been written so that they can be referenced by other PCI standards in the future. Anyone who participates in the credit card ecosystem, including merchants, should take note of these changes.

How Will The New PCI Software Security Framework Impact Your Organization?

There is now a stronger focus on implementing a secure software development process for Payment Applications (PAs). PA providers will be obliged to drastically improve their application security programs in order to comply. This may apply to other payment ecosystem participants in the future. Thus, PA providers will need to transform their organizational secure development practices. Unlike previous PCI standards, the Software Security Standards are objective-based, thus offering vendors flexibility in how they go about complying with each specific objective.

---

1 https://blog.pcisecuritystandards.org/just-published-new-pci-software-security-standards
2 https://www.pcisecuritystandards.org/documents/PCI-Secure-Software-Standard-v1_0.pdf
3 https://www.pcisecuritystandards.org/documents/PCI-Secure-SLC-Standard-v1_0.pdf
4 https://blog.pcisecuritystandards.org/update-on-pci-software-security-framework

How to comply with the PCI Software Security Standards

Software developers are adopting more competitive software lifecycle management techniques with faster release cycles, and the PCI Software Security Standards were designed to better support this agile development environment. In the advent of the new framework, the payment industry will see more consistency in how software is evaluated for security. To help organizations comply with the PCI Software Security Standards, we offer our own solution, comprised of our policy-to-procedure platform, SD Elements, Just-in-Time training and role-based eLearning, Implementation Services, and Verification Services.

.

An Overview of Our Solution

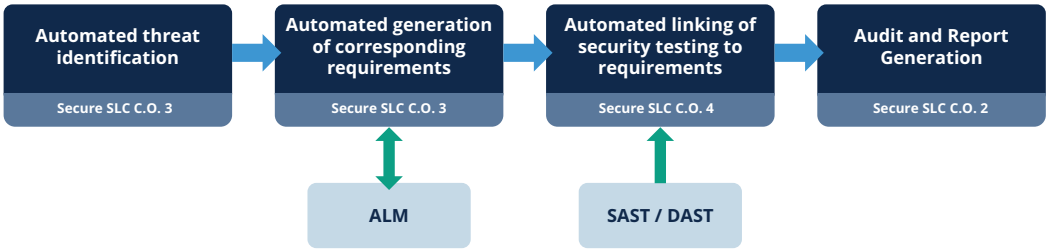
SD ELEMENTS PLATFORM

SD Elements is a policy-to-procedure platform: a technology that translates high-level security policy (e.g., policy statements like, “Developers should consider security in their application design”) into actionable work tasks that IT personnel can use to ensure that relevant elements of the policy are being followed (e.g., a detailed technical guideline on how to implement Single Sign On for a web-based application). Security and compliance teams often form their programs by crafting security policies. These policies are then delivered to operating teams, who are responsible for interpreting the policies and executing on them through a mix of manual processes and automated tools. In the case of software developers, this usually involves translating security and compliance policies into coding requirements and running security scanning tools. Since virtually all policy is open to nuances and interpretations, there is usually some degree of ambiguity between the policy and the actionable procedures that developers must follow. Tracing the policy to its execution is also a challenge developers face. The sheer number of policies and

standards is often too large for modern software teams to parse through during their development process. A policy-to-procedure platform, like SD Elements, is designed to fill these gaps. Using this platform, policy teams can define and communicate risk policies for their applications based on regulations, industry standards, and internal policies into actionable tasks. Security Compass' dedicated research team maintains a rich library of these procedures, complete with mappings to several industry standards. Once applications are on-boarded to the platform, SD Elements provides a contextually relevant set of actionable procedures that enables secure and compliant software, by design. This means less time is spent trying to interpret what the policy means and more time is spent working to improve security.

To start, a member of your organization completes a 15-minute survey to build a profile of your application's technical and business properties (e.g., a C++ embedded device that stores and processes personal data). Typically, a project manager or architect will complete the survey, providing information about the language, platform, features, compliance, and tools, which SD Elements uses to automatically determine relevant threats and countermeasures. Once the survey is complete, SD Elements generates a set of relevant security threats and compliance issues. These threats and compliance issues are then translated to preventative procedures or "tasks." For example, task T154 in SD Elements states, "Do not store or cache credit card information on client" and is linked to compliance with PCI regulations in the platform.

Our policy-to-procedure platform seamlessly integrates with your existing development processes, synchronizing with all popular Application Lifecycle Management (ALM) tools on the market, including Jira, Microsoft Team Foundation Server, or CA Agile. From here, SD Elements pushes security and compliance tasks to developers as tickets, user stories, or work items. These tasks can include Just-in-Time Training (JITT) modules and relevant code samples to help educate developers at the time of implementation. Once your developer marks a task as 'complete,' SD Elements provides traceability to testing. Software teams can either manually run tests, using our provided testing instructions, or automatically verify tasks by importing the results from Static Analysis Security Testing Tools (SAST) such as Veracode, Checkmarx, or Fortify and Dynamic Analysis Security Testing Tools (DAST) such as WebInspect or AppScan.



SD Elements can generate reports which help organizations track progress, risk profiles, and compliance. In the event of an audit, you can use our platform to generate custom reports and activity logs to show accountability. SD Elements also provides risk dashboards, making it easy to see your compliance status with internally defined risk policies. This saves considerable time during audits and security reviews, and it offers traceability in the event of an incident.

JUST-IN-TIME TRAINING AND ELEARNING

Our unique market offering, Just-in-Time Training (JITT), provides secure coding guidance for developers, delivering task-specific instructions at the point of need. In 2019, we will also be launching a new course specifically designed to train clients on the new PCI Software Security Standards. This course offers new material while leveraging some of our existing content, training users in threat modeling, application security testing, and more.

IMPLEMENTATION SERVICES

We offer a proven implementation methodology to ensure the successful adoption of our software. Working together with your team, we help develop processes and integrate tools, so that you can meet the new PCI standards in a cost-effective manner.

VERIFICATION SERVICES

Our verification services help to identify vulnerabilities, commensurate with the PCI Software Security Standard Control Objectives. We offer penetration testing, red teaming, vulnerability triaging for Static and Dynamic Application Security Testing tools, and more. Our security consultants have extensive experience with enterprise clients, fixing and breaking code while taking a strategic approach to your organization's security problems. We can also help you prepare for a successful security requirement audit.

Please see Page 17 for more information about our Security Compass solution components.

How Our Solution Helps Organizations to Comply with the Secure Software Lifecycle (Secure SLC) Standard

To initiate your compliance with the Secure SLC standard, we have created a software security policy template. The template helps satisfy the formal policy requirements identified in the PCI Software Security Framework, and it can be customized to your organization's specific needs. This, along with our solution and common security tools such as static analysis, will be the key tools you need to become compliant.

Below, we explain how each part of the Secure Software Lifecycle Standard aligns with our solution to help your organization comply. We also offer a concise summary table, showing how these new PCI requirements directly map on to our Security Compass solution, found in Appendix A.

SOFTWARE SECURITY GOVERNANCE

(1) Security Responsibility and Resources

Control Objective 1.1
To account for the security of the software vendor's products and services, your senior leadership team must assign this responsibility.

Control Objective 1.2
The person or team assigned such responsibility must keep the senior leadership team updated about security matters.

You can comply with this control by using our software security policy template. This document can guide your team in defining roles and responsibilities as they relate to security.

Control Objective 1.3
All individuals with software security roles and responsibilities possess and maintain skills in software security matters relevant to their specific role, responsibility, and job function.

We offer standard role-based eLearning and training reports to ensure that those held responsible have the right software security skills needed to fulfill this job function.

(2) Software Security Policy and Strategy

Control Objective 2.1
Regulatory and industry security and compliance requirements that are applicable to your vendor's operations, products, services, or data used must be identified and monitored. This means that a mature and repeatable process must be in place for identifying external regulatory and industry security and compliance requirements.

SD Elements helps to satisfy this control by delivering relevant compliance requirements for software via regulatory reports. You can also create a process task in SD Elements to help you identify and track applicable compliance regulations.

Control Objective 2.2
A software security policy must establish measurable rules for ensuring that your vendor's services and products are secure and that their security and compliance obligations are fulfilled.

Our platform, SD Elements, can provide such metrics. Our software security policy template also outlines such rules, helping to ensure that the vendor's offerings are secure and compliant.

Control Objective 2.3
A formal software security strategy is required to ensure that the security of your software vendor's products or services is meeting the security policy your organization set out to follow.

Our software security policy also comes with a software security strategy, helping to ensure that your organization's security policy is followed.

Control Objective 2.4, 2.5
Software security assurance processes must be implemented and maintained throughout the entire software lifecycle. Evidence is generated and maintained to demonstrate the effectiveness of software security assurance processes

Not only does SD Elements embed security requirements early on in the software development lifecycle, but it also delivers a completion status report, which improves visibility and accountability. Additionally, our software security policy can provide guidance on how to implement and maintain an assurance process throughout the software lifecycle.

Control Objective 2.6
The PCI Software Security Standard mandates that failures or weaknesses in software security assurance must be detectable and improved or replaced, if detected.

Using SD Elements, your team can generate a report that shows which tasks are complete, incomplete, or needing review. Our software security policy also advises on the detection, improvement, and replacement of software security assurance measures that show weaknesses or failures.

SECURE SOFTWARE ENGINEERING

(3) Threat Identification and Mitigation

Control Objective 3.1
Critical assets must be identified and classified. This includes pinpointing the confidentiality, integrity, and resiliency requirements for each critical asset. Critical assets include all sensitive data, resources, and functions related to a vendor's payment software.

Our software security policy provides a sample asset classification scheme, while the SD Elements risk policy function serves as a database of asset classification. You can also create a process task in SD Elements to help identify and classify critical assets.

Control Objective 3.2
A mature process must exist for identifying and assessing threats to the software and weaknesses inherent to its design.

Using SD Elements, your team can generate a problem summary report, helping to identify such threats. SD Elements also has a task for open-source patching, so that the process can be systematically tracked and monitored.

Control Objective 3.3
Software security controls must be implemented in software to mitigate threats and design weaknesses.

SD Elements can be used to show an application's completion status or problem summary related to its compliance with the PCI Software Security Standards.

Control Objective 3.4
Failures or weaknesses in security controls must be detected and improved or replaced if detected.

Our software security strategy, which appears in our software security policy, provides guidance on how to proceed in the event of failed or weak security controls. SD Elements also provides completion status reports, indicating which controls have succeeded, which controls have failed, or which controls require review.

(4) Vulnerability Detection and Mitigation

Control Objective 4.1
The presence of software vulnerabilities must be detected in a timely manner.

SD Elements integrates with Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST) scanners to show which controls have failed and which requirement the controls were related to, via task verification, speeding up the detection process.

Control Objective 4.2
Newly identified vulnerabilities must also be fixed in a timely manner, and the reintroduction of previously remediated vulnerabilities must be prevented.

The risk acceptance in SD Elements can be set so that only a predetermined number of failed tasks can be accepted. You can take advantage of task verification in SD Elements to confirm whether implemented controls are functioning properly, and you can use a process task to track that security testing has been performed. Lastly, our software security policy provides guidance on how vulnerabilities can be addressed in a timely manner.

SECURE SOFTWARE AND DATA MANAGEMENT

(5) Change Management

Control Objective 5.1
All changes to payment software must be identified, assessed, approved, and tracked.

SD Elements can be used to create process tasks to track that these activities take place.

(6) Software Integrity Protection

Control Objective 6.1, 6.2
The integrity of all software code, including third-party components, must be maintained throughout the entire software lifecycle. Software releases and updates are delivered in a secure manner that ensures the integrity of the software code.

Our software security policy also provides guidance related to maintaining software code integrity.

(7) Sensitive Data Protection

Control Objective 7.1, 7.2
Sensitive production data can only be collected and retained on vendor systems where there is a legitimate business or technical need. This data should be protected when kept on vendor systems and securely deleted when it is no longer required.

Our software security policy will outline these requirements and our software security strategy will provide guidance on how to follow them. We also have a process task in SD Elements for ensuring the proper collection and protection of production data, as well as a secure coding task which shows how to do this in code. These concepts are also addressed in some of our eLearning courses, including 'Defending Databases' and 'Cloud Security'.

SECURITY COMMUNICATIONS

(8) Security Guidance

Control Objective 8.1, 8.2, 8.3
Security guidance and instructions must be provided to stakeholders to lead them through the necessary steps involved in secure deployment and configuration of software. That is, a software vendor must provide security configuration guidance for each security-related option or parameter that is available. The security guidance must also be aligned with incoming software updates.

As part of our solution, SD Elements has a process task for tracking the development of guidance on secure installation and the maintenance of all software components.

(9) Stakeholder Communications

Control Objective 9.1, 9.2
Your software vendor must maintain communication with stakeholders regarding potential security issues and strategies to mitigate these, if they do arise. The communication channels must be defined and laid out in an accessible way for customers, installers, and other relevant parties, so that they can report and receive information related to security issues and mitigation strategies. In the event that there are no readily available security updates to address detected vulnerabilities or attacks, security notifications must be sent to all of your stakeholders, providing instructions on how to mitigate the corresponding risks.

You can refer to our policy for further guidance on stakeholder communications. You can also use the process task in SD Elements for tracking the creation of these channels.

(10) Update Information

Control Objective 10.1
Lastly, your software vendor must provide stakeholders with detailed explanations of all software changes and updates. Once any given software update is released, a comprehensive summary of the software changes must be communicated to the stakeholders.

You can use the process task in SD Elements to track the release of a change summary upon each software update.

How Our Solution Helps Organizations to Comply with the Secure Software Standard

This document generally provides lower-level instructions related to software development best practices.

It also provides more information on what security activities should be tracked and how. SD Elements' Tasks address most of the controls in the Secure Software Standard.

In fact, using SD Elements for a software project contributes to meeting "Control Objective 10: Threat and Vulnerability Management," since 'Problems' reports in SD Elements address the common attacks and weaknesses (which are mapped to CWEs), and 'Tasks' in SD Elements address mitigation techniques.

In its current state, SD Elements contributes to meeting most of these controls. As we adapt these standards, we will increase our coverage and include more references to necessary material for complete implementation of each control.

PCI Software Security Standards Compliance: Our Solution

SD Elements

Our policy-to-procedure platform, **SD Elements**, provides a strong foundation for compliance with the new PCI Software Security Standards. Created and maintained by our research team, SD Elements is a technology platform that translates policy into actionable tasks that developers can understand and implement directly into their code. Below is a list of SD Elements' specific features which can help with PCI compliance:

- **Compliance:** a knowledge base of actionable tasks and requirements for PCI software security standard compliance, the PCI Data Security Standard (PCI DSS) as well as multiple other compliance standards.
- **Enterprise Ready:** integrations fit with any development environment.
- **Automated Threat Modeling:** streamlines the process by automating the identification of commonly occurring software threats, allowing security experts to focus on unique and domain-specific threats.
- **Training:** comes with computer-based eLearning modules for developers, relevant to their work tasks and delivered at the point of need.
- **Force Multiplier for Security Architects and Privacy Engineers:** allows experts to accomplish more by automating part of their current workflow, allowing them to better attend to product, feature, and sprint planning.
- **A "Shift Left" and "Build Security In" Method:** identifies and eliminates application security vulnerabilities earlier on in the software development lifecycle.

- **Reports:** An important component of the PCI Software Security Standards and audit, SD Elements contains the project data necessary to automatically generate reports, offering a real-time view of compliance status, and it provides metrics to guide strategy. For example:
    - **Risk Policy Report.** Shows compliance to a specific policy for all projects and applications by business unit across the organization
    - **Compliance Report.** Line by line detail of development, completion, and validation status of the control from the PCI Software Security Standards
    - **Problem Summary Report.** A list of all threats that apply to the project (i.e., the threat model)
    - **All Tasks Report.** Shows all the tasks related to the project and their current status. Includes PCI SSS controls, process tasks, and any other applicable compliance standards and organization-specific tasks

- **Process tasks [Coming Soon]:** We provide high-level instructions on software security processes through our software security policy, and SD Elements can be used to track whether a process has been executed. Our platform also keeps an audit trail which can be integrated into ALM workflow systems, such as Jira. Below are some examples of process tasks we can help manage:

    - Identify applicable compliance regulations
    - Identify and classify critical assets
    - Identify, assess, and monitor software threats
    - Use a software security management solution to track security controls
    - Perform security testing using SAST and DAST to detect and mitigate vulnerabilities
    - Identify, assess, approve, and track software changes
    - Maintain the integrity of all software code including third-party
    - Ensure the integrity of software release and update delivery
    - Proper collection and protection of production data
    - Provide and maintain guidance on secure installation and maintenance of all software components
    - Establish and maintain a bi-directional communication channel for receiving security reports and sending security notifications
    - Maintain and release a change summary upon each software update
    - Perform independent security testing

Services

Enterprise Delivery Services. We offer a proven implementation methodology to ensure successful adoption of our software. Working together with your team, we help develop processes and integrate tools, so that you can meet the new PCI compliance standards in a cost-effective manner. These are categorized into the following high-level services:

- Process Design & Project Planning: in this stage, a project plan is created and success metrics are established. Executive buy-in is obtained and change management planning is done. Process design is facilitated, teams and champions are identified, and a rollout plan is created. AppSec Program Gap Analysis and Planning

- SD Elements Enablement (i.e., Technical Implementation)
    - Configuration of roles (global and project-specific), business unit workflow options, notification settings, custom statuses, and risk policies, including PCI Software Security Standards compliance as a requirement
    - Enterprise integrations setup including SSO, ALM, SAST, DAST and CI/CD (e.g. Jenkins)
    - Localization of Corporate Content (addition of custom content if required): remove or hide unused content, apply corporate terminology, review and update 'Frequently Customized' tasks, create custom compliance to report against required tasks, and add organization-specific content for process tasks. Subject Matter Expert Development

- Training of Subject Matter Experts/End users on solution: we train for 5 different roles, including System/Server Admin (1-2 persons), Software Admin (2-3 persons), Content Admin (1-10 persons), Project Admin or 'SD Elements Process Champions' (# of persons scales with # of applications), and End Users (teaching a self-serve model for using eLearning, documentation, and in-app tips).

- Organizational Change Management: we create user guides and FAQs, offer onboarding assistance for ~5 apps (which includes an introductory demo of SD Elements, answering the survey with our clients, an initial review of compiled controls for the project, and syncing outstanding work to their ALM).

Documentation. Security Compass can help organizations to create the documentation required for compliance with the PCI Software Security Standards. We start with our template policy in the areas listed below, and we help you customize them to your organization's unique environment. The policy template includes:

- Roles and responsibilities throughout the SDLC, governance and oversight, and necessary skills for a secure SDLC
- A software security strategy with a description of the general strategy for software security that aligns with ISO 27034, and how SD Elements is used to achieve compliance. This can also provide guidance on how to implement control objectives 5 –10, including instructions on how to create security guidance, vendor security communication, and documentation for change management.

Just-in-Time Training and eLearning

Security Compass offers various forms of online training for security activities that are relevant to the new PCI Software Security Framework. Just-in-Time Training (JITT) course modules are delivered through SD Elements, while our traditional eLearning courses can be consumed on our own LMS or uploaded into your own LMS. With our course curriculum, students will learn:

- Core concepts of application security, security requirements, secure coding and testing
- How (and when) to conduct a manual Threat Model Express (TME)
- How to assess for weak security controls
- Handling sensitive data
- How to identify risks when sourcing software from the supply chain, including open source
- **Coming Soon:**
    - A course specific on the new PCI Software Security Framework that covers topics such as:
        - Common security roles and responsibilities,
        - Understanding specific control requirements,
        - Understanding penetration testing
        - Understanding Software Composition Analysis (SCA), also known as open source scanning
        - Understanding SAST/DAST
        - Protecting release integrity,
        - Vulnerability disclosure handling,
        - and Creating security guidance for products.
    - How to configure and use SD Elements

Conclusion

Innovations in payments technologies are rapidly advancing, with new software constantly appearing on the market. As a result, payment application providers will now see a stronger focus on software security and secure development practices related to the payment card industry. The new PCI Software Security Framework has potential to significantly affect other payment standards, which payment application providers should bear in mind. This is an unprecedented move in the payment card industry which may have an impact on how other industries, including health care, automotive, and Internet of Things, treat security in the future. These new standards will also offer software vendors the flexibility and transparency required to achieve reasonable security objectives while supporting an agile approach to software development.

To learn more about how our SD Elements can help you comply with the new PCI Software Security Framework, contact us:

Email: info@securitycompass.com
Website: www.securitycompass.com/contact/

Note: The PCI Software Security Framework Guide covers requirement procedures and controls covered under the new PCI Secure Software Standard and the PCI Secure Software Lifecycle (Secure SLC) Standard, released as part of a new PCI Software Security Framework, published by PCI SSC on 16th January, 2019.

Appendix A - PCI Secure SLC Standard Mapping for Security Compass Solution

Below, we've mapped the PCI Secure Software Lifecycle controls to our Security Compass solution. Components of our solution include our policy-to-procedure platform, SD Elements, Just-in-Time Training, and our eLearning courses.

Control Objective Section	Control Objective #	The Security Compass Solution
Control Objective 1: Security Responsibility and Resources The vendor's senior leadership team establishes formal responsibility and authority for the security of the vendor's products and services. The vendor allocates resources to execute the strategy and ensure that personnel are appropriately skilled.	1.1 Overall responsibility for the security of the vendor's products and services is assigned by the vendor's senior leadership team.	We created a software security policy template which outlines roles & responsibilities as they relate to software security. It also stipulates senior management oversight as well as bi-annual reviews. Our software security policy can be customize so that it's relevant to organizations' own work environments.
	1.2 Software security responsibilities are assigned.	Our software security policy, which outlines common roles & responsibilities in software security.
	1.3 Software development personnel maintain skills in software security matters relevant to their specific role, responsibility, and job function.	Our software security policy outlines the necessary skills for security roles. We offer Just-in-Time Training, as well as standard role-based eLearning, to train individuals in software security skills, including secure coding
Control Objective 2: Software Security Policy and Strategy The vendor defines, maintains, and communicates a software security policy and a strategy for ensuring the secure design, development, and management of its products and services. Performance against the software security strategy is monitored and tracked.	2.1 Regulatory and industry security and compliance requirements applicable to the vendor's operations, products, and services and the data stored, processed, or transmitted by the vendor are identified and monitored	SD Elements provides all necessary compliance requirements for software. It also generates a report to show progress on any relevant regulations and industry standard frameworks. Our software security strategy & process template document refers to SD Elements as the central repository for storage and maintenance of this information.

Control Objective Section	Control Objective #	The Security Compass Solution
(continued)	2.2 A software security policy is defined and establishes the specific rules and goals for ensuring the vendor's products and services are designed, developed, and maintained to be secure, resistant to attack, and to satisfy the vendor's security and compliance obligations.	Our software security strategy & process template defines specific rules and goals to ensure that the vendor's offerings are designed to be secure and that they are maintained that way. Security rules and goals can be measured using SD Elements' metrics and reports.
	2.3 A formal software security strategy for ensuring the security of the vendor's products and services and satisfying its software security policy is established and maintained.	Just by using SD Elements, organizations can easily implement industry standard frameworks' processes for application security, such as ISO 27034. We can help you customize our software security strategy & process template to suit your organization's needs.
	2.4 Software security assurance processes are implemented and maintained throughout the entire software lifecycle.	Adhering to our software security strategy & process document will provide guidance on how to implement assurance processes. SD Elements is built to ensure that security assurance processes are established and maintained throughout the whole software development lifecycle, through the all pre-development and some post-development stages.
	2.5 Evidence is generated and maintained to demonstrate the effectiveness of software security assurance processes	SD Elements generates a completion status report, clearly showing whether software security assurance processes are working successfully.
	2.6 Failures or weaknesses in software security assurance processes are detected. Weak or ineffective security assurance processes are updated, augmented, or replaced.	Our software security strategy & process document provides guidance on how to augment or replace security-assurance processes that are weak or ineffective. SD Elements' completion status reports show failures and weaknesses in software security assurance processes.

Control Objective Section	Control Objective #	The Security Compass Solution
Control Objective 3: Threat Identification and Mitigation The vendor continuously identifies, assesses, and manages risk to its payment software and services.	3.1 Critical assets are identified and classified.	Our software security strategy & process document helps to identify and define critical assets for your organization. SD Elements' risk policies allow you to assign higher priority levels to tasks related to your identified critical assets.
	3.2 Threats to the software and weaknesses within its design are continuously identified and assessed	SD Elements offers lightweight, automated threat modeling, so that you can identify all domain-agnostic threats that apply to your software project. Our knowledge base of threats is continually updated by our in-house team of application security researchers. SD Elements generates a problem summary report, indicating software threats and design weaknesses. SD Elements creates and tracks process tasks, including: • keeping an inventory of open source components • analyzing vulnerabilities and patching in open source components
	3.3 Software security controls are implemented in the software to mitigate threats and design weaknesses.	Once threat modeling is complete, SD Elements automatically generates the controls from our knowledge base to mitigate threats. SD Elements generates a problem summary report, indicating outstanding software threats and design weaknesses. SD Elements generates a completion status report, indicating which controls have been successfully implemented and verified.
	3.4 Failures or weaknesses in software security controls are detected. Weak or ineffective security controls are updated, augmented, or replaced.	Our software security strategy & process document offers guidance on how to identify, augment, or replace ineffective security controls. SD Elements generates completion status reports, indicating which controls have been successfully implemented and verified. Just-in-Time Training gives developers the guidance needed to implement effective security controls in their code.

Control Objective Section	Control Objective #	The Security Compass Solution
Control Objective 4: Vulnerability Detection and Mitigation The vendor detects and mitigates vulnerabilities in the software and its components to ensure that payment software remains resistant to attacks throughout its entire lifetime.	4.1 The existence and emerging of software vulnerabilities are detected in a timely manner.	SD Elements integrates with SAST and DAST tools, providing enhanced verification for PCI security control requirements. SD Elements features comprehensive task verification statuses for security controls. These can be updated manually or automatically via integration with security tools.
	4.2 Newly identified or discovered vulnerabilities are fixed in a timely manner. The reintroduction of similar or previously resolved vulnerabilities is prevented.	Our software security strategy & process document defines best practices for remediation of discovered vulnerabilities. SD Elements has task verification functionalities, verifying which security controls have been successfully implemented. SD Elements generates a completion status report which details accepted risks and comments associated with these, along with the verification status from either manual or automated testing.
Control Objective 5: Change Management Identify and manage payment software changes throughout the software lifecycle.	5.1 All changes to payment software are identified, assessed, and approved.	SD Elements can be used to create process tasks for defining and tracking change management processes.
Control Objective 6: Software Integrity Protection Protect the integrity of the payment software throughout the software lifecycle	6.1 The integrity of all software code, including third-party components, is maintained throughout the entire software lifecycle.	Our software security strategy & process document provides guidance on software code integrity. SD Elements can be used to create process tasks for ensuring that the integrity of software is maintained. Training [coming in 2019]: Understanding the PCI Software Security Framework
	6.2 Software releases and updates are delivered in a secure manner that ensures the integrity of the updated code	Our software security strategy & process document provides instruction on ensuring the integrity of software releases and updates. SD Elements can be used to implement a process task on protecting release integrity. Training [coming in 2019]: Understanding the PCI Software Security Framework

Control Objective Section	Control Objective #	The Security Compass Solution
Control Objective 7: Sensitive Data Protection The confidentiality of customers' sensitive production data on vendor systems is maintained.	7.1 Sensitive production data are only collected and retained on vendor systems where there is a legitimate business or technical need.	Our software security strategy & process document details the conditions under which sensitive production data can be collected on vendor systems. SD Elements can be used to create a process task for seeking authorization related to the collection & retention of sensitive data Training [coming in 2019]: Understanding the PCI Software Security Framework
	7.2 Sensitive production data are protected when retained on vendor systems and securely deleted when no longer needed	Our software security strategy & process document details best practices for protection and retention of sensitive data. SD Elements can be used to create a process task for seeking authorization related to the collection & retention of sensitive data. Training [coming in 2019]: Understanding the PCI Software Security Framework
Control Objective 8: Vendor Security Guidance The vendor provides stakeholders with clear and thorough guidance on the secure implementation, configuration, and operation of its payment software applications.	8.1 Security guidance and instructions are provided to stakeholders to guide them through the secure implementation and configuration of the software.	Our software security strategy & process document assists with best practices in delivering guidance (in the form of user manuals) to your customer about secure implementation of your payment solutions SD Elements can be used to create a process task delivering security guidance and manuals to your clients regarding your payment solution.
	8.2 Security guidance includes detailed instructions on how to securely install, configure, and maintain all software components and supported platforms.	Our software security strategy & process document provides instructions on secure installation and maintenance of all software components. SD Elements can be used to create a process task on how to create security guidance and manuals for your solution Training [coming in 2019]: Understanding the PCI Software Security Framework
	8.3 Security guidance is aligned with software updates.	Our software security strategy & process document outlines a process for updating manuals and guidance along with software releases SD Elements can be used to create a process task for updating security guidance and manuals, timed with software release dates

Control Objective Section	Control Objective #	The Security Compass Solution
Control Objective 9: Stakeholder Communications The vendor maintains communication channels with stakeholders regarding potential security issues and mitigation options.	9.1 Communication channels are defined and made available for customers, installers, integrators, and other relevant parties to report and receive information on security issues and mitigation options.	Our software security policy templates help to identify stakeholders and define roles for handling stakeholder communications Our software security strategy & process document defines best practices for communicating with a variety of stakeholder groups Training [coming in 2019]: Understanding the PCI Software Security Framework
	9.2 Stakeholders are notified about security updates in a timely manner.	SD Elements can be used to create a process task for notifying stakeholders about security updates.
	9.3 Where security updates are not readily available to address known vulnerabilities or exploits, security notifications are issued to all relevant stakeholders to provide instructions for mitigating the risks associated with the known vulnerabilities and exploits.	Our software security strategy & process document defines best practices for known vulnerability disclosures. Training [coming in 2019]: Understanding the PCI Software Security Framework
Control Objective 10: Software Update Information The software vendor provides stakeholders with detailed explanations of all software changes.	10.1 Upon release of any software updates, a summary of the specific changes made to the software is provided to stakeholders.	Our software security strategy & process document defines best practices for communicating software updates to stakeholders. SD Elements can be used to create a process task for ensuring that updates to software are communicated to all stakeholders.

SecurityCompass

Security Compass believes in a world where people can trust technology, and it is our mission to help customers proactively manage cybersecurity risk, without slowing down business. Our holistic program offerings are tailored to your organization's needs. Through advisory services, training, and SD Elements, our award-winning policy to procedure platform for security and compliance, we set up your company with all of the resources and tools it needs to develop secure software. Security Compass serves some of the world's largest enterprises, including 16 of the largest financial institutions in North America, South America, and Asia, as well as 4 of the largest tech companies in the world.

OFFICES

GLOBAL HEADQUARTERS
1 Yonge Street
Suite 1801
Toronto, Ontario
Canada M5E 1W7

TORONTO
390 Queens Quay W
2nd Floor
Toronto, Ontario
Canada M5V 3A6

CALIFORNIA
1001 Bayhill Drive
2nd Floor
San Bruno, California
USA 94066

NEW JERSEY
621 Shrewsbury Avenue
Suite 215
Shrewsbury, New Jersey
USA 07702

INDIA
#4.07
4th Floor, Statesman House
Barakhamba Road, New Delhi
India 110001

1.888.777.2211
info@securitycompass.com
www.securitycompass.com

@SECURITYCOMPASS
SECURITY COMPASS