Payment Card Industry (PCI)

Secure Software Standard

Technical FAQs for use with the Secure Software Standard v1.x

May 2025

Table of Contents

Document Changes

Date	Description
May 2023	Initial publication.
May 28 2025	**New Technical FAQ(s):** General: Q5 Module C: Q1

PCI Secure Software Standard: Technical FAQs

This document addresses common questions related to the *PCI Software Security Framework: Secure Software Requirements and Assessment Procedures* (i.e., the Secure Software Standard) version 1.x and its associated validation and listing program. This document is an integral part of the Secure Software Standard and Program and must be fully considered during a Secure Software assessment.

Technical FAQs are a mechanism to provide additional clarifications regarding the interpretation and application of security and program requirements between major revisions of a given PCI security standard. Clarifications provided in Technical FAQs are typically incorporated into a given standard or program upon the next major revision.

New or updated questions and/or answers since the last revision of this document are shown in red.

General

Q1 May 2023: Do all control objectives in the Secure Software Standard need to be met?

A *It is the expectation that all control objectives must be satisfied for payment software to be considered "compliant" with the Secure Software Standard. It may be possible, however, for alternative approaches to be considered acceptable if it can be demonstrated through assessor testing and evidence generation that at least one of the following is true:*

- *The control objective is based on a specific condition that does not exist within a given implementation of the assessed software. For example, the presence of clear-text account data or the use of cryptography. See Q2 for more information on addressing this scenario.*
- *There is a legitimate technical constraint that prevents the control objective from being met. See Q3 for more information on addressing this scenario.*

In both scenarios, it is expected that the control objective be evaluated and the existence of such conditions or constraints be documented and justified in the Report on Validation (ROV). Failure to sufficiently describe and justify such conditions or constraints will result in the ROV being rejected by PCI SSC.

Q2 May 2023: When is it appropriate to mark a control objective in the Secure Software ROV as "N/A" (Not Applicable)?

A *Some control objectives may be based on the existence of certain conditions that do not exist in all payment software implementations. For example, control objective A.2.2 is based on clear-text account data being stored, processed, or transmitted by the assessed payment software.*

In such cases, it may be permissible to mark a control objective as "N/A" in the ROV if it can be demonstrated, through assessor testing and evidence generation, that the control objective does not apply to the assessed payment software. Using the previously-noted example, Control Objective A.2.2 may be marked as "N/A" if it can be demonstrated that the payment software does not store, process, or transmit clear-text account data.

All "N/A" findings in the Secure Software ROV must include a detailed description explaining how it was determined that the control objective does not apply and references to the evidence that was obtained and/or generated that supports the "Not Applicable" finding.

Q3 May 2023: What should be done if a control objective cannot be met as stated due to a technical constraint?

A *In some software implementations, it may be impossible for the assessed payment software to meet a particular control objective due to legitimate technical constraints. For example, limitations of the software development languages or frameworks used or other technical limitations within the execution environment.* *(continued on next page)*

In such circumstances, all such constraints must be documented and justified in the ROV. Where possible, additional security controls must be implemented to mitigate any additional risk associated with the inability to satisfy the control objective. Examples of appropriate mitigations may include the use of sandboxing techniques or other capabilities available within the execution environment.

If the implementation of such mitigations requires user input or interaction, then it is expected that the software vendor provide, at a minimum, guidance on how to implement such mitigations and/or direct the user to where appropriate configuration information can be obtained.

If the additional risk(s) cannot be mitigated to a reasonable degree, then the control objective cannot be considered met ("In Place").

Q4 May 2023: What is the deadline for new Secure Software v1.1 submissions?

A *The deadline for new submissions to the Secure Software Standard v1.1 has been extended from June 30 to August 31, 2023. As of September 01, 2023, all new Secure Software submissions must be completed using the latest versions of the Secure Software Standard, Program Guide, Report on Validation (ROV) Template, and Attestation of Validation (AOV).*

All 'In Process' submissions (i.e., those submitted, and fees paid to PCI SSC prior to September 01, 2023) will have until November 29, 2023 to complete those submissions.

Q5 May 2025: Is a Full Software Assessment always required as part of a High Impact change?

A *No, it may not always be required. The vendor and assessor work together to determine the impact of the software change and all PCI Secure Software Standard requirements potentially affected by the change.*

The assessor performs all the required assessment activity based on the software change, and as part of the delta assessment, completes a redlined ROV (i.e., not a new ROV), along with all other required documentation as part of the delta submission.

Module C – Web Software

Q1 May 2025: Is the term 'Internet Accessible Interfaces' in the Secure Software Standard intended to restrict the applicability of the Web Software Module solely to software that is exposed to the Internet?

A *No.*

The term 'Internet Accessible Interfaces' is intended to encompass any software interface that could potentially be exposed to the Internet. This ensures that appropriate mechanisms are implemented in software to protect all software interfaces, regardless of how the software may be deployed or configured in a production environment.