Payment Card Industry (PCI)

Software Security Framework

Glossary of Terms, Abbreviations, and Acronyms

Version 1.2

December 2022

Document Changes

Date	Version	Description
January 2019	1.0	Initial release
February 2021	1.1	Updates to add or modify terminology to support the Secure SLC Program expansion and the introduction of the Secure Software Standard: Terminal Software Module.
December 2022	1.2	Updates to add or modify terminology to support the introduction of the Secure Software Standard: Web Software Module.

Term	Definition
Abstraction layer	A fundamental term in object-oriented programing describing a process to hide implementation details from users. As part of a layered architecture with hardware at the base and other layers building on top of the hardware, each layer can access the features of layers below it, but no layer can access layers above it. While abstraction can improve code flexibility and maintenance, it can also pose problems to incident handling and forensics. Entities should understand the complex hierarchies of abstraction layers, such as in many cloud-computing environments, and understand the different ways in which digital evidence is lost due to abstraction layers.
Adversarial testing	Methods or techniques used during a software evaluation to force the software to behave in unintended ways or to bypass software security controls.
Application program interface (API)	A series of communication protocols, subroutines and tools for building software that allows two applications to interact with each other.
Assessor	Individuals approved by PCI SSC (as defined in the associated program documentation) to perform security assessments against PCI standards, including those standards associated with the PCI Software Security Framework.
Assessor company	Companies approved by PCI SSC (as defined in the associated program documentation) to perform security assessments against PCI standards, including those standards associated with the Secure Software Framework.
Authentication credentials	A combination of the user ID (or account ID) and the authentication factor(s) used to authenticate an individual, device, application, system, or process.
BCrypt	A password-hashing algorithm based on Blowfish.
Big-number library functions	Library functions that allow for the large numbers often used in cryptography to be processed and stored correctly (with needed levels of precision) in languages that may otherwise default to a less precise format.
Common Weakness Enumeration (CWE)	A category system for software weaknesses and vulnerabilities.
Confidential data	A form of sensitive data that explicitly requires protection from unauthorized disclosure. Examples of confidential data include cardholder data (CHD), sensitive authentication data (SAD), and private cryptographic keys.

Term	Definition
Control objective	The high-level security objective that must be met.
Critical assets	Term used to collectively reference all sensitive data, sensitive functions, and sensitive resources. Critical assets are those data, functions, and resources that if exposed, misused, altered, or disabled, could impair the software's ability to function properly or meet its security objectives.
Data element	Term used to represent a single piece of information or datum.
Default software settings	Software settings that are configured or active upon software installation, initialization, or first use.
EMVCo	A global technical body owned by American Express, Discover, JCB, Mastercard, UnionPay, and Visa that facilitates the worldwide interoperability and acceptance of secure payment transactions by managing and evolving the EMV® Specifications and related testing processes.
Entropy	Term used in cryptographic operations to represent the measure of the unpredictability of a random seed value. Entropy is generally measured in "bits," where a higher number indicates that the particular event is less predictable than an event with a lower number. Entropy is used to measure the security strength of cryptographic keys.
Execution environment	The collective hardware, software, and services required by a software application for operation. This includes, but may not be limited to, the hardware, networks, operating systems, databases, storage systems, and services required by the software to function as intended. Also commonly referred to as the execution platform or runtime environment.
External communications	Any communication method that uses a wireless, local-area network, wide-area network, or a public domain protocol or security protocol to transport data. This includes, but is not limited to, Bluetooth, Wi- Fi, cellular, or Ethernet, and a serial point-to-point connection that is wireless or through a hub, switch, or other multiport device.
Federal Information Processing Standard (FIPS) Publication 140-2	A standard that provides four increasing, qualitative levels of security and is related to the Cryptographic Module Validation Program (CMVP), which provides for vendors to submit products to a laboratory to validate cryptographic modules to the FIPS 140-2 standard and other cryptography-based standards. Products validated as conforming to FIPS 140-2 are accepted by the U.S. and Canadian federal agencies for the protection of sensitive information (United States) or designated information (Canada).

Term	Definition
Firmware	Any code within the POI device that provides the protections needed to comply with the *PCI PIN Transaction Security (PTS) Point of Interaction (POI) Modular Security Requirements* (PCI PTS POI Standard) or that can impact compliance with the standard. Firmware may be further segmented by code, as necessary, to meet subsets of device requirements. Other code that exists within the device that does not provide security and cannot impact security—with the exception of prompt control and secure reading and exchange of data (SRED) applications—is not considered firmware. See the *PCI PTS POI Standard* for more information.
Flash wear-leveling function	A technique used to extend the life of solid-state drives (SSDs) and USB (non-volatile RRM, a.k.a flash) drives that involves arranging data so that erasures and re-writes are distributed evenly across the drive. This technique makes hard-drive-oriented techniques for individual file sanitization ineffective on SSDs. Flash-based solid-state drives (SSDs) differ from common hard drives in both the technology used to store data (flash chips vs. magnetic disks) and the algorithms used to manage and access that data. As wear leveling creates a layer of indirection between the logical block addresses that computer systems use to access data and the raw flash addresses that identify physical storage, it can produce copies of the data that are invisible to the user but which a sophisticated attacker can recover.
Functional testing	The evaluation of software against functional requirements to verify the software has met those requirements.
Hooking	A technique for intercepting application calls to a function for some purpose, usually to customize and extend its functionality as well as to monitor aspects of an application.
Index token	A cryptographically-generated token that is based on a given index for an unpredictable value. Index tokens are often used as a substitute for sensitive data where persistent storage is required.
Initialization vector (IV)	Blocks that are used to mask data (plaintext) prior to encryption with a block cipher. Without the addition of an IV, identical plaintext messages would not encrypt to different ciphertext messages.
Install base	The number of units of a software application that are currently implemented and in use. The install base is generally regarded as the size or amount of code comprising all software functionality and is typically measured in “bytes”.
“K” value	A randomly or pseudorandomly generated integer used in the creation of digital signatures.

Term	Definition
Mature process	A process that is established (i.e., performed at least once before), is repeatable across personnel and geographical locations, and whose output or outcomes are predictable.
NIST Statistical Test Suite	Examinations for determining whether a random number generator is suitable for a particular cryptographic application per *NIST Special Publication 800-22 Rev 1a: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.*
One-time pad	An encryption technique that combines plaintext with a randomly-generated key or "pad" that is as long as the plaintext. One-time pads are used only once. If the key is truly random, never reused, and kept secret, the one-time pad is considered “unbreakable.” One-time pads are typically used to encrypt data and/or messages between two parties where both parties possess matching one-time pads and keys.
Open Web Application Security Project (OWASP)	A non-profit organization focused on improving the security of web application software. OWASP maintains a list of critical vulnerabilities for web applications. (See http://www.owasp.org).
Padding	A collection of techniques used in cryptography to prevent an attacker from knowing the exact length of a plaintext messaging and using that predictability to break the encryption.
Payment data	Data created, captured, or exchanged for the explicit purpose of conducting an electronic payment transaction. Examples of payment data may include cleartext, encrypted, and/or tokenized Account Data.
Payment environment	Term used to holistically describe all manual and automated processes and systems involved in the execution of payment transactions.
Payment software	Software that stores, processes, or transmits payment data.
Payment terminal	A PCI-approved POI device.
PBKDF2	A widely used key-derivation function (KDF) published by RSA Laboratories. In general, KDFs take a source of initial keying material and derive from it one or more pseudorandom keys. KDFs that input user passwords are known as password-based KDF (PBKDF). KDFs use CPU-intensive operations on the attacker side that increase the cost of an exhaustive search. By applying a KDF to a user password, legitimate users spend a moderate amount of time on key derivation, while the time needed for an attacker to test each possible password is increased, hopefully beyond a practical limit.

Term	Definition
PCI-approved POI device	An electronic transaction-acceptance product that complies with the *PCI PTS POI Standard* and appears on the PCI SSC's *List of Approved PTS Devices*.
PCI Software Security Framework	A collection of standards, programs, and supporting documentation for the purposes of enhancing software security in the Payment Card Industry.
Persistent data	Data that is retained in non-volatile storage and persists even if power to the device is shut off.
Protection methods	The practices, processes, techniques, tools, procedures, or mechanisms implemented to protect from a risk event.
Rainbow table attack	A data-level attack using a pre-computed table of hash strings for the purposes of identifying the original data source. Rainbow table attacks are typically used for cracking passwords or cryptographic hashes.
RAM disk or RAM drive	A block of random-access memory that software treats as if the memory is a physical disk drive.
Regression testing	A software evaluation technique used to confirm that updates to address specific software issues (such as faulty functionality or security problems) sufficiently address those issues, do not introduce other software issues, and remain compatible to existing code.
Resiliency	The extent to which software can maintain normal operations amid adverse conditions, including the ability to recover from a fault or an attack.
Roll	The act of changing a discretionary data element (such as a cryptographic key) at a predefined period or event of obsolescence.
Sampling	The process of selecting a subset of a group of objects or subjects that is representative of the entire group.
Secure deletion	The process of removing or overwriting data residing on a hard disk drive or other digital media (including memory), rendering the data irretrievable.
Secure SLC qualified vendor	A software vendor that has had its software lifecycle management practices assessed and qualified to the Payment Card Industry SSC Secure SLC Requirements and meets all PCI program requirements associated with Secure SLC vendor qualification.

Term	Definition
Software lifecycle management practices	The evolution process of a software application from inception through design, development, deployment, maintenance, and finally decommission. Secure software lifecycle management practices include security-related activities and processes to facilitate the secure design, development, operation, and management of software.
Security testing	The process of identifying flaws related to elements of confidentiality, integrity, authentication, availability, authorization, and non-repudiation in the assessed system component(s) and security mechanisms. This process usually includes, but is not limited to, activities such as threat modeling, code reviews, vulnerability assessment, penetration testing, fuzz testing, etc.
Seed data	A starting value used in the process of generating random numbers to initialize a pseudorandom number generator.
Sensitive data	Any data that requires protection from unauthorized disclosure (confidentiality) or modification (integrity). Sensitive data includes, but is not limited to, cardholder data (CHD), sensitive authentication data (SAD), tokens, cryptographic key material, and authentication credentials, internal system information, and other data defined by a software vendor as requiring protection. Sensitive data may also be present in software design characteristics, session data, token data, status information, and error messages.
Sensitive functions	Any software function that facilitates access to or the modification of sensitive data. Examples of sensitive functions may include authentication functions, cryptographic functions, communication protocols, processing daemons, etc.
Sensitive production data	Sensitive data that is owned and/or generated by an entity other than a software vendor. Sensitive production data is typically obtained from software that has been deployed into a production environment owned and/or managed by another entity, such as customers, partners, or other stakeholders.
Sensitive resources	Any software resource that constitutes, contains, or otherwise facilitates access to sensitive data. Examples of sensitive resources may include files, registry keys, data sets, environmental settings, cryptographic keys, or seed values.

Term	Definition
Software components	An abstract concept to describe parts of a larger software application that are typically self-contained and may be sold, licensed, or distributed independently. Software components are typically embedded into the code of a larger application (as open-source or third-party packages or libraries) and may also consist of third-party APIs, services, and other software dependencies that are external to an application but are considered an essential part of that application's architecture.
Software-development personnel	The staff or personnel who are responsible for or involved in the design, creation, and maintenance of software. Depending on the activities being performed it may include individuals who specify, design, develop, document, test, and fix bugs in software.
Software function	A distinct and uniquely identifiable collection of software code for the purpose of performing a specific task or series of tasks. Software functions may also be software components if they are packaged, distributed, and made accessible independently from other application functions or components.
Software resource	Any uniquely identifiable electronic data object that is required for a software application to provide its intended functionality (such as a static HTML file, a configuration file, or stylesheet) or that is generated as a result of a related software function (such as a query result or output file).
Software security assurance processes	A method for determining a level of confidence that the security-related functions of software work as intended and are free of vulnerabilities that may have been included in the software.
Software security controls	Security-related features and functionality built into or relied upon by software to protect against software threats and attacks.
Software user	Any user of a software application, interface, or system. Software users may be human or machine and may include, for example, software functions interacting with other software functions via an Internet accessible API.
Software vendor	A software provider, supplier, developer, or other entity that produces or otherwise distributes software and/or software components for commercial purposes.
Split knowledge	A method by which two or more entities separately have key components that individually convey no knowledge of the resultant cryptographic key.
SSD over-provisioning	A technique used by solid-state drive (SSD) manufacturers to improve performance by reserving free space. SSD manufacturers reserve an additional percentage of the total drive capacity for over-provisioning (OP) during firmware programming to use in various disk-administration functions.

Term	Definition
Stakeholder	Any entity affected by the security of software at any stage during the software's lifecycle. Stakeholders include entities that use, install, or integrate with software. Stakeholders may also include personnel, business partners, and other third parties.
Stream ciphers	Methods to convert plaintext to ciphertext one bit at a time.
Substitution box (S Box)	A nonlinear substitution table used in several byte-substitution transformations and in the key-expansion routine to perform a one-for-one substitution of a byte value.
Terminal software	Software that is intended for deployment and execution on a PCI-approved POI device that does not meet the definition of firmware as defined in the *PCI PTS POI Standard*.
Test requirements	The validation activities to be performed by an assessor to determine whether a specific control objective has been met.
Transaction types	Payment transaction functions that include, but may not be limited to the following: authorization (goods and services), cash (ATM), debit adjustment, refund, available funds inquiry, balance inquiry, payment from account, payment to account, etc. See *ISO 8583:2003 – Financial transaction card originated messages – Interchange message specifications for more information*.
Transient data	Data that is created and retained (usually in volatile memory) for the purposes of a single application session. At the end of the session, the data is securely deleted or is reset back to its default values and is not stored persistently.
White-box cryptography	A method used to obfuscate a cryptographic algorithm and key with the intent that the determination of the key value is computationally complex.
XOR	A connective in logic known as the "exclusive or," or exclusive disjunction. It yields true if exactly one (but not both) of two conditions is true.