Payment Card Industry

Software Security Framework

Glossary of Terms, Abbreviations, and Acronyms

Version 1.0

January 2019

Document Changes

Date	Version	Description
January 2019	1.0	Initial release

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.
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 Software Security Framework.
Authentication credentials	A combination of the user ID or account ID plus the authentication factor(s) used to authenticate an individual, device, 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. See *Sensitive Data*.
Control objective	The specific software security controls and outcomes required to satisfy the parent security objective.
Critical assets	Collective term to describe any software element that if exposed, misused, altered, or disabled could impair the software's ability to function properly or meet its security objectives. Sensitive data, sensitive functions, and sensitive resources are also considered critical assets.

Term	Definition
Data elements	Individual fields carrying payment-card transaction information. They are defined in ISO 8583, the International Organization for Standardization (ISO) standard for systems that exchange electronic transactions initiated by cardholders using payment cards.
Default software settings	Software settings that are configured or active upon software installation, initialization, or first use, depending upon the software architecture or deployment environment.
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	In cryptographic operations, 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	In the context of the Software Security Framework, a collection of services and components relied on by the payment software during operation. It includes all components and services that are used and relied on by the software to make a complete system—for example, the processors, hardware components, networks, operating systems, databases, and cloud platforms (e.g., PaaS and SaaS). May also be known as execution platform or runtime environment.
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).
File integrity monitoring (FIM)	A technique or technology under which certain files or logs are monitored to detect if they are modified. When critical files or logs are modified, alerts should be sent to appropriate security personnel.

Term	Definition
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	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.
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 units of a product or service that are currently implemented and in use.
"K" value	A randomly or pseudorandomly generated integer used in the creation of digital signatures.
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 or not a 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	In cryptography, the one-time pad is an encryption algorithm with text combined with a random key or "pad" that is as long as the plaintext and used only once. Additionally, if the key is truly random, never reused, and kept secret, the one-time pad is unbreakable.
Open Web Application Security Project (OWASP)	A non-profit organization focused on improving the security of application software. OWASP maintains a list of critical vulnerabilities for web applications. (See http://www.owasp.org).

Term	Definition
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 exchanged for purposes of conducting a transaction which may include account or tokenized data.
Payment software	In the context of the Software Security Framework, software involved in or directly supporting or facilitating payment transactions.
Payment software components	Software components of a larger software application that are typically self-contained and often sold, licensed, or distributed independently. Examples of software components include open-source libraries, third-party APIs, and other dependencies that are part of the software architecture.
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.
Persistent sensitive data	Sensitive data that is retained in non-volatile storage and persists even if power to the device is shut off.
Protection methods	Actions (e.g., practices, processes, techniques, tools, procedures, methods, principles, rules) taken to protect from a risk event.
Rainbow table attack	A method of data attack using a pre-computed table of hash strings (fixed-length message digest) to identify the original data source, usually for cracking passwords or cardholder data 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	Evaluation 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 of software to recover from a fault or an attack.

Term	Definition
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 cross-section of a group that is representative of the entire group.
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.
Secure deletion	A method of removing or overwriting data residing on a hard disk drive or other digital media (including memory), rendering the data irretrievable. Additional guidance on methods for secure deletion are provided in *ISO 27038:2014 Security techniques – Specification for digital redaction* or *NIST Special Publication 800-88, Guidelines for Media Sanitization.*
Security objective	The security objectives form the primary goal and shape the intention of the control objectives and test requirements in the Software Security Framework.
Secure Software Lifecycle (Secure SLC or SSLC)	The evolution process of a software application from inception through design, development, deployment, maintenance, and finally decommission. A Secure SLC ensures that security is integrated at all stages of the software lifecycle.
Secure Software Lifecycle practices	Security-related activities and processes to facilitate the secure design, development, operation, and management of secure software.
Security testing	Security testing is a 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. The process usually includes, but is not limited to, activities such as threat modeling, code reviews, vulnerability assessment, penetration testing, fuzz testing, etc.
Seed data (for random number generators)	A starting value used to initialize a pseudo-random number generator.
Sensitive data	In the context of the Software Security Framework, 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 the software vendor as requiring protection. Sensitive data may also be present in software design characteristics, session data, status information, and error messages.

Term	Definition
Sensitive functions	Any functionality of the payment software that alters other software functionality or configuration, processes sensitive data, provides security features, or interacts with sensitive resources. Examples of sensitive functions include authentication functions, cryptographic functions, communication protocols, processing daemons, etc.
Sensitive resources	External resources upon which software relies to provide security features or process sensitive data. Sensitive resources are often provided by or shared with the underlying platform, operating environment, or other applications that coexist within or outside the software's operating environment. Examples of sensitive resources include shared files, registry keys, environmental settings, communication channels, cache, shared libraries, system interfaces, web services, etc. In many cases, sensitive resources may also constitute "sensitive data" and may require protection from unauthorized disclosure or modification.
Software-development personnel	The staff or personnel who are involved with the development of secure payment software including the design, creation, and maintenance of applications, frameworks, or other software components. Depending on the activities being performed it may include individuals who specify, design, develop, document, test, and fix bugs in payment software.
Software security assurance processes	A method for determining a level of confidence that the security functions of software work as intended and are free of vulnerabilities that may have been included in the software.
Software security controls	Security features and functionality built into payment software or the software's operating environment to protect against software threats and attacks.
Software Security Framework	A collection of software security standards that leverage a common validation and certification model.
Software vendor	Entity that produces and sells payment software.
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
SSLC-Qualified Software Vendor	A software vendor who has had its software lifecycle management practices assessed and validated to the Secure SLC Requirements and meets all PCI program requirements associated with Secure SLC validation.
Stakeholder	Entity affected by the security of the payment software at any stage during the software's lifecycle. Stakeholders include entities that use, install, or integrate the payment software. Stakeholders may also include software vendor personnel, business partners, and other third parties as defined by the software vendor.
Stream ciphers	Methods to convert plaintext to cipher text one bit at a time.
Test requirements	The validation activities to be performed by an assessor to determine whether a specific control objective has been met.
Transaction types	In the context of the Software Security Framework, may include: authorization (goods and services), cash (ATM), debit adjustment, refund, available funds inquiry, balance inquiry, payment from account, payment to account, etc. See ISO 8583.
Transient sensitive data	Sensitive data that is created within an application session. At the end of the session, it is intended to be securely deleted or reset back to its default values or settings and not stored. See also *Persistent Sensitive Data*.
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.