PCI Security Standards Council®

Guideline:	PCI Mobile Payment Acceptance Security Guidelines
Version:	2.0
Date:	September 2017
Author:	Emerging Technologies, PCI Security Standards Council

PCI Mobile Payment Acceptance Security Guidelines for Developers

Table of Contents

Foreword

The PCI Security Standards Council (PCI SSC) is an open global forum for the ongoing development, enhancement, storage, dissemination, and implementation of security standards for account data protection. The rapid development of payment-acceptance alternatives using mobile technologies has led PCI SSC to consider its approach to developing and providing guidance to secure all implementations.

The PCI Security Standards Council charter provides a forum for collaboration across the payment space to develop security standards and guidance for the protection of payment card data wherever it may be stored, processed, or transmitted—regardless of the form factor or channel used for payment. All this applies when a merchant, service provider, or other entity accepts payment card data from its customers. When individuals load their own primary account numbers (PAN) into their personal devices, however, they are not required to validate those devices to PCI standards. At the same time, when one of those personal devices is transformed into a point of sale (POS) for a merchant to accept account data, there is the responsibility to protect that information. Thus, PCI standards begin to apply when a mobile device is used for payment card acceptance and may be subject to PCI DSS compliance as dictated by the payment brands' compliance programs.

This document focuses on payment applications that operate on any consumer electronic handheld device (e.g., smartphone, tablet or wearable—or collectively, "mobile device") that is not solely dedicated to payment-acceptance transaction processing, where the electronic handheld device has access to clear-text data, and the device is not PCI PTS eligible. For ease of reference, this subcategory is referred to as "Category 3, Scenario 2." This scenario does not include the use of a validated P2PE solution. Separate PCI standards and documentation available on the PCI SSC website deal with all other categories and scenarios:

- *Mobile Payment-Acceptance Applications and PA-DSS FAQs*
- *PCI PTS POI Modular Security Requirements* (Category 1) – Payment application operates only on a PTS-approved mobile device.
- *PCI Payment Application Data Security Standard (PA-DSS)* (Category 2) – Payment application meets all of the following criteria:
  - i. Payment application is only provided as a complete solution "bundled" with a specific mobile device by the vendor;
  - ii. Underlying mobile device is purpose-built (by design or by constraint) with a single function of performing payment acceptance; and
  - iii. Payment application, when installed on the "bundled" mobile device—as assessed by the Payment Application Qualified Security Assessor (PA-QSA) and explicitly documented in the payment application's Report on Validation (ROV)—provides an environment that allows the merchant to meet and maintain PCI DSS compliance.
- *Accepting Mobile Payments with a Smartphone or Tablet* (Category 3, Scenario 1) – Payment application operates on any consumer electronic handheld device (e.g., smartphone, tablet, wearable—or collectively, "mobile devices") that is not solely dedicated to payment acceptance for

transaction processing. The scenario includes the use of an approved hardware accessory in conjunction with a validated P2PE solution.

PCI SSC agreed (see *PA-DSS and Mobile Applications FAQs*) that mobile payment-acceptance applications that qualify, as Category 3 will not be considered for PA-DSS validation until the development of appropriate standards to ensure that such applications are capable of supporting a merchant's PCI DSS compliance. The PCI SSC recommends that mobile payment-acceptance applications that fit into Category 3 be developed using PA-DSS requirements and the guidance provided in this document as a baseline.

The purpose of this document is to raise awareness and to provide guidance to those in the best position to protect the trust needed for a payment application that executes within mobile devices: the solution developers. This document encourages the development of secure payment-acceptance solutions including applications using secure coding practices, and encourages both monitoring for advancements that improve integrity and preparing for newly discovered threats. While not exhaustive, this document outlines a variety of both traditional and less conventional mechanisms to isolate account data and protect it from exposure.

Disclaimer

Please consider carefully the limitations of this document. In particular:

- No presumption should be made that meeting the guidelines and recommendations expressed in this document would cause a solution to be compliant with PA-DSS. Entities wishing to use such solutions would need to make their own risk assessments around the use of such solutions in consultation with their acquirers and applicable payment brands. Such solutions would be included in an entity's annual PCI DSS assessment to ensure that the application and its operating environment are compliant with all applicable PCI DSS requirements.
- Due to its rapid evolution, payment brands may have differing approaches to mobile payment acceptance. The guidelines and recommendations expressed in this document may not, by themselves, be sufficient to meet the specific requirements of all payment brands or territories. For example, manual key entry on a merchant-owned consumer mobile device may be prohibited in some territories but permitted in others. For information and in the event of any doubt, please contact your acquirer and/or the relevant payment brands/territories.

1 Document Overview

1.1 Document Purpose and Scope

The Payment Card Industry Security Standards Council (PCI SSC) recognizes that merchants may use consumer electronic handheld devices (e.g., smartphones, tablets, wearables—or collectively, “mobile devices”) that are not solely dedicated to payment acceptance for transaction processing. For instance, a merchant might use an off-the-shelf mobile device for both personal use and payment acceptance. Most of these devices do not meet security characteristics required by generally accepted information security standards. Examples of information security standards can be found in Appendix C.

The purpose of this document is to educate stakeholders responsible for the architecture, design, and development of mobile applications and their associated environment within a mobile device that merchants might use for payment acceptance. Developers and manufacturers can use these guidelines to help them design appropriate security controls within their software and hardware products. These controls can then be applied to mobile payment-acceptance environments, thus supporting the deployment of more secure solutions.

This document focuses on two areas: controls that may be currently satisfied by technology in today’s environment, and controls meant to give guidance and direction for the design of mobile payment-acceptance apps and their associated environment within a mobile device. Where merchants’ mobile-device hardware and software implementation cannot currently meet the guidelines documented herein, they may choose to implement a PCI-validated, point-to-point encryption (PCI P2PE) solution. Implementing such a solution would include the addition of a PCI-approved point-of-interaction (POI) device. With the use of a validated P2PE solution, account data is encrypted by the POI, and the mobile device simply acts as the conduit through which the encrypted payment transaction is transmitted.

1.1.1. Definition

This document defines mobile devices as consumer electronic devices that are not solely dedicated to payment acceptance for transaction processing. These devices span a broad spectrum of features and functions ranging from cellular handsets that only support telephone functionality to smartphones, tablets, or wearables—or collectively, “mobile devices—with a Rich OS, which gives the mobile devices broader functionality.

1.2 Security Risks of Mobile Devices

Any risk that exists on a standard desktop or laptop computer also exists on a mobile device. In addition, mobile devices may have a broader set of functionalities than standard desktop and laptop computers, resulting in the possibility of more security vulnerabilities. Along with the standard communication methods of traditional desktop and laptop computers, mobile devices may also incorporate multiple cellular technologies (e.g., CDMA and GSM), GPS, Bluetooth, infrared (IR), and near-field communication (NFC) capabilities. While no longer unique to mobile devices, sensors like cameras and microphones also have vulnerabilities and risks associated with their use. Risk is further increased by removable media (e.g., SIM card and SD card), the internal electronics used for testing by the manufacturer, embedded sensors (e.g., tilt or motion sensors, thermal sensors, pressure sensors, and light sensors), and biometric readers. Furthermore, vendor and network operator-level logging and debugging configurations may introduce additional risks.

Security risks are also inherent to the developmental life cycle and infrastructure associated with mobile devices. For example, the original equipment manufacturer, the operating-system software developer, the application developer, the integrator, the reseller, the mobile-network operator (or cellular service provider), and the mobile payment-acceptance solution provider each play a part in the overall security of a mobile device. (NOTE: This may not be a complete list.) Some developers are involved in multiple stages of the development process, making it potentially easier for them to address more aspects of the device from the silicon layer to the applications running on the operating system; other stakeholders are involved in only one stage of security development. Other third parties may introduce security risks through device drivers, mobile apps, peripheral equipment, and removable media. All of these represent potential vectors for unauthorized access to device operations or unauthorized disclosure of account data. Deciding who is responsible for which best practice may be confusing given the large number of contributors to the development of a mobile device. For more clarity, see the “Best Practices and Responsibilities” matrix in Appendix B.

2 Mobile Payments Guidance Overview

The cardholder data environment (CDE) is comprised of people, processes, and technology that store, process, or transmit cardholder data or sensitive authentication data, including any connected system components. This document does not focus on a PCI-validated P2PE solution, but on providing guidance for reducing security risks in otherwise noncompliant mobile devices. For mobile payment acceptance, the mobile device would be considered part of the CDE with full PCI DSS applicability unless used in tandem with a PCI-validated P2PE solution—refer to the PCI AT A GLANCE Mobile Payment Acceptance Security document entitled *Accepting Mobile Payments with a Smartphone or Tablet*. In this type of solution, the mobile device would act only as a “pass-through” for the encrypted data sent from the POI device.

This document organizes the mobile payment-acceptance security guidelines into the following two sections:

- **Section 3:** Objectives and Guidance for the Security of a Payment Transaction

  This section addresses the three main risks associated with mobile payment transactions:

  i. Account data entering the device,

  ii. Account data residing in the device, and

  iii. Account data leaving the device.

- **Section 4:** Guidelines for the Risk and Controls in the Supporting Environment

  In addition to the guidelines specific to payment transactions, this section addresses security measures that are essential to the integrity of the mobile platform and associated application environment.

3 Objectives and Guidance for the Security of a Payment Transaction

This section addresses the three main risks associated with mobile payment transactions:

i. Account data entering the device,

ii. Account data residing in the device, and

iii. Account data leaving the device.

. An objective with associated guidance is given to address each of the three risks.

Objective 1: Prevent account data from being intercepted when entered into a mobile device.

Guidance:

Ensure account data is appropriately encrypted prior to entry into a mobile device. This can be accomplished via a validated PCI P2PE solution.[1]

– OR –

Ensure a trusted path[2] exists between the data-entry mechanisms—e.g., manual key entry or entry via a card reader—and the mobile device such that account data cannot be intercepted by an unauthorized party. One option to accomplish this is using a trusted execution environment that restricts access between the mechanism receiving account data and secured memory located inside the device.

If an external device is used for account data entry into the mobile device, that device should also have a means of demonstrating that it is authorized to communicate with the mobile device.

If the external device is wireless—e.g., Wi-Fi or Bluetooth—the wireless communication channel should be secured via strong cryptography.[3]

Regardless of the process used, assure the account-data-entry channel is secured against client-side injections. Client-side injections include but are not limited to buffer overflows, data-type mismatches, embedded code or other unexpected data, and malicious or unauthorized apps and services on the mobile device.

If the solution permits PIN entry, it should only occur through a PCI PTS-approved PIN entry device or a solution that has been approved by the payment brand(s).

[1] For more information, refer to AT A GLANCE Mobile Payment Acceptance Security document entitled *Accepting Mobile Payments with a Smartphone or Tablet,* available at www.pcisecuritystandards.org.

[2] See Appendix A for the definition of “trusted path.”

[3] See *PCI DSS and PA-DSS Glossary of Terms, Abbreviations, and Acronyms* for the definition of “strong cryptography.”

Objective 2: Prevent account data from compromise while processed or stored within the mobile device.

Guidance:

Ensure that account data is only processed inside a trusted execution environment[4]. A trusted execution environment may be accomplished through multiple technologies, and the level of security may vary accordingly. In order to prevent data leakage, account data should not be accessible outside a trusted execution environment. A data-leakage prevention methodology should be adopted based on industry best practices and guidelines. The methodology should include, but is not limited to:

- Secure distribution of account data
- Secure access to and storage of account data
- Controls over account data while in use (e.g., preventing copy/paste, screen shots, file sharing, and printing)
- Prevention of unintentional or side-channel data leakage[5]

Temporary storage of account data prior to processing and authorization should be in a secured storage environment, such as a secure element, to prevent third party eavesdropping.

If account data is stored on the mobile device post-authorization, that data should be rendered unreadable per PCI DSS Requirement 3.4. If encrypted account data is stored, any related cryptographic keys must be managed in accordance with PCI DSS Requirement 3.5 so keys are not accessible to unauthorized people, applications, and/or processes.

Per PCI DSS Requirement 3.2, do not retain sensitive authentication data (SAD)[6] after authorization. This includes ensuring that neither the mobile device nor any attached device retains SAD after authorization.

[4] See Appendix A for the definition of “trusted execution environment.”

[5] OWASP Top 10 Mobile Risks—see Appendix C #10.

[6] See Appendix A for the definition of “sensitive authentication data (SAD).”

Objective 3: Prevent account data from interception upon transmission out of the mobile device.

Guidance:

Ensure that account data is encrypted—i.e., using strong symmetric or asymmetric cryptography—per PCI DSS Requirement 4, prior to transmission out of the trusted execution environment of the mobile device. Ensure encrypted account data is transmitted from a trusted source.

Mobile payment-acceptance applications are vulnerable to numerous types of attacks meant to intercept data—such as, but not limited to: MITM or passive eavesdropping on a compromised device, poorly implemented cryptography, or eavesdropping through the cellular infrastructure (i.e., air interface, carrier)[7].

[7] NIST Mobile Threat Catalogue https://pages.nist.gov/mobile-threat-catalogue/

4 Guidelines for the Risk and Controls in the Supporting Environment

This section addresses security measures essential to the integrity of the mobile platform and associated application environment.

4.1 Prevent unauthorized logical device access.

Protect mobile device from unauthorized logical access. Include design features that prevent unauthorized use. For example, include in the design one of the more secure lock screens: “Biometric[8]”, “Password,” “Pattern,” or “PIN.” Do not rely on “Slide,” since it does not add security. Include a feature that would force the user to re-authenticate to the device after a specified amount of time. Bypassing of the lock screen may be prevented by enabling full media encryption and/or disabling USB debugging.

Disabling USB debugging and disallowing untrusted sources should be enforced on an ongoing basis. Use of application-hardening techniques may also help prevent unauthorized logical access to a device by making it more difficult for an attacker to modify or reverse engineer the software by reducing the attack surface.

The mobile app developer should include the capability for the mobile app to determine whether USB debugging is disabled and whether full media encryption is enabled. In addition, the operating-system developer should include controls that can prevent the user from enabling USB debugging or disabling full media encryption.

***Note:** Biometrics is a maturing technology. For the particular biometric chosen, the developer must ensure the biometric has the appropriate strength and security controls for their particular payment-acceptance solution.*

4.2 Create server-side controls and report unauthorized access.

Develop the overall payment-acceptance solution to include capabilities for preventing and reporting unauthorized access attempts, identifying and reporting abnormal activity, and discontinuing access—i.e., the payment-acceptance solution would prevent further access by the mobile payment-acceptance app on that device until an administrator restores access. Controls include, but are not limited to:

- Support for authorized access—e.g., access control list
- Ability to monitor events and to distinguish normal from abnormal events
- Ability to report events—e.g., via a log, message, or signal—including cryptographic key changes, escalation of privileges, invalid login attempts exceeding a threshold, updates to application software or firmware, and similar actions

[8] See Appendix C for information on biometric standards and guidance.

4.3 Prevent escalation of privileges.

Controls should exist to prevent the escalation of privileges on the device (e.g., root or group privileges). Bypassing permissions can allow untrusted security decisions to be made, thus increasing the number of possible attack vectors. Therefore, the device should be monitored for activities that defeat operating-system security controls—e.g., jailbreaking or rooting—and, when detected, the device should be quarantined by a solution that removes it from the network, removes the payment-acceptance application from the device, or disables the payment application. Offline jailbreak and root detection and auto-quarantine are key since some attackers may attempt to put the device in an offline state to further circumvent detection. Hardening of the application is a method to that may help prevent escalation of privileges in a mobile device. Controls should include, but are not limited to:

- Providing the capability for the device to produce an alarm or warning if there is an attempt to root or jailbreak the device;
- Providing the capability within the payment-acceptance solution for identifying authorized objects[9] and designing controls to limit access to only those objects.

4.4 Create the ability to remotely disable the payment application.

The payment application should support a mechanism that permits it to be disabled by the merchant or solution provider responsible for the payment-system application. The feature should not interfere with other, non-payment functions of the mobile device.

4.5 Detect theft or loss.

A process should exist for the detection and reporting of the theft or loss of the mobile device. Inherent to such a process should be a means for testing and for confirming that it remains active. Examples include the use of GPS or other location technology with the ability to set geographic boundaries, periodic re-authentication of the user, and periodic re-authentication of the device.

4.6 Harden supporting systems.

Supporting systems that either provide management for mobile devices or receive payment card data should be hardened to prevent unintended access or exposure of a mobile payment transaction. Therefore, any system used to support the mobile payment-acceptance solution should be compliant with PCI DSS.

4.7 Harden the application.

Mobile payment-acceptance applications should be hardened to prevent unintended logical access or tampering with the app such as code injection or reverse engineering. Numerous techniques can be used for hardening of the mobile payment-acceptance application that will reduce the attack surface.

[9] See Appendix A for the definition of *object.*

4.8 Prefer online transactions.

When the mobile payment-acceptance application on the host is not accessible, the mobile device should neither authorize transactions offline nor store transactions for later transmission.

4.9 Conform to secure coding, engineering, and testing.

Mobile payment-acceptance applications should conform to secure coding, engineering, and testing conventions, such as the requirements and testing procedures outlined in the *Payment Application Data Security Standard* (PA-DSS). Other examples include CERT Secure Coding Standards[10], Institute for Security and Open Methodologies (ISECOM)'s Open Source Security Testing Methodology Manual (OSSTMM)[11], or International Systems Security Engineering Association (ISSEA)'s Systems Security Engineering Capability Maturity Model (SSE-CMM – ISO/IEC 21827) [12] and OWASP Projects.[13] .

Developers should be trained on PCI standards and secure-coding best practices. Training should cover prevention of common coding vulnerabilities in software development processes to include but not be limited to injection flaws, buffer overflow, insecure cryptographic storage, insecure communications, improper error handling, and improper access control—e.g., OWASP Top 10 Mobile Risks[14].

Developers should also document their implementation and create a formal response plan to identify and mitigate new risk. Developers should establish a process to identify and assign a risk ranking to newly discovered security vulnerabilities and to test their applications for vulnerabilities. Any underlying software or systems that are provided with or required by the application should be included in this process.

4.10 Protect against known vulnerabilities.

Provide a secure means for keeping mobile device software and all applications up to date through patch management and other means to prevent compromise of the mobile device due to vulnerable software. Controls should include but are not limited to:

- Evaluating updates prior to implementing them.
- Ensuring that updates are received from a trusted source.
- Applying updates in a timely manner.

4.11 Protect the mobile device from unauthorized applications.

All authorized mobile apps, drivers, and other software that form part of the payment solution should have a mechanism that permits authentication of the source and integrity of the executable file. The system should prevent the loading and subsequent execution of applications that cannot be authenticated. Developers should ensure that a process exists for the secure distribution of their software such that an end user can determine that the software came from a trusted source before installing it. For instance, it

[10] www.cert.org/secure-coding/

[11] http://www.isecom.org

[12] http://www.sse-cmm.org/index.html

[13] https://www.owasp.org/index.php/Mobile_Top_10_2016-Top_10

may not be permissible to download apps from an online store whose security cannot be validated.

Solution providers should regularly update their payment application and ensure a process is in place to notify merchants when updates are available and safe to install. The solution provider should also provide documentation to merchants that details any update procedures for installation, as well as ensure a process is in place to notify merchants when newly discovered vulnerabilities in their payment-acceptance solution are found and how to install tested patches for said vulnerabilities.

4.11.1 When apps will be distributed through online app stores, only use stores that have formal acceptance processes that include software integrity controls and source-authentication controls. Note that the OS of a device may have built-in restrictions regarding what app stores are acceptable without altering the OS.

4.12 Protect the mobile device from malware.

Enhance current capabilities to protect mobile devices from malware. Deploy anti-malware products on all systems including antivirus, antispyware, and software-authentication products to protect systems from current and evolving malicious software threats.

If anti-malware software is not available, employ MAM (Mobile Application Management) or MDM (Mobile Device Management) solutions that can monitor, evaluate, and remove malicious software and applications from the device. Furthermore, if possible, it is ideal to deploy both anti-malware and MDM solutions (mentioned above) to protect the device from malicious software and applications. As another example, consider application wrapping, which can be employed with an MDM solution to prevent and/or remove malicious software and applications. Application hardening may also be used as a method to protect against malware.

Mechanisms (such as a displayed icon) should exist to demonstrate that persistent protection is active and that it is from a trusted source.

4.13 Protect the mobile device from unauthorized attachments.

If an enry device (e.g., card reader) is attached to the mobile device—whether the connection is physical or wireless—it needs to identify itself uniquely to the mobile payment-acceptance app to ensure that the correct entry device is paired to the correct mobile device. Mutual authentication between the entry device and the mobile device provides the best integrity assurance for the path. When the entry device is attached, the mobile payment-acceptance app validates the account-data-entry device via a serial number or other unique identifier.

4.14 Create instructional materials for implementation and use.

Documentation should exist specifically to address the proper, secure use of mobile devices in the merchant environment, including instructional material on the hardware, operating system, and application software.

4.15 Support secure merchant receipts.

Regardless of the method used for producing receipts—e.g., e-mail, SMS, or attached printer)—the method should mask the PAN in support of applicable laws, regulations, and payment-card brand policies. Insecure channels such as e-mail and SMS should not be used to send PAN or SAD.

4.16 Provide an indication of secure state.

A trusted execution environment (or equivalent) must include a mechanism for indicating to the mobile-device user that the payment-acceptance mobile app is executing in a secure state. This would be similar to the indication that an SSL session is active in a browser.

4.17 Provide audit and logging mechanisms for user and device access.

Developers should provide an audit and logging mechanism for their payment-acceptance solution. The mechanisms are expected to log user and device access to be used by the merchant. The developer should document how logs would be made available to the merchant and how to access the log reports.

Appendix A: Glossary

This glossary contains definitions of words and phrases that are specific to *PCI Mobile Payment Acceptance Security Guidelines for Developers.* For all other definitions, please refer to the *PCI DSS and PA-DSS Glossary of Terms, Abbreviations, and Acronyms*.

Term	Definition
Application hardening (anti-tampering)	Application hardening is a process of addressing application security weaknesses. This may be done by implementing software patches and updates, using the latest versions of protocols, and following policies and procedures in order to minimize attack surfaces or down time.
Application wrapping	Application wrapping typically involves the addition of a dynamic library to the existing application binary. This library can provide additional controls for certain aspects of the application (e.g., required user authentication, forced use of a VPN or prohibit cut and paste).
Bluetooth	Wireless protocol using short-range communications technology to facilitate transmission of data over short distances.
Cardholder data	At a minimum, cardholder data consists of the full PAN. Cardholder data may also appear in the form of the full PAN plus any of the following: cardholder name, expiration date and/or service code. See *Sensitive authentication data (SAD)* for additional data elements that may be transmitted or processed (but not stored) as part of a payment transaction.
Card reader	A mechanism for reading data from a payment card.
Clear text	Intelligible data that has meaning and can be read or acted upon without the application of decryption.
Developer	An organization that architects, designs, or builds hardware or software components (e.g., manufacturer, operating-system software company, mobile network operator [MNO], third-party application software company, integrator, or implementer); this may include solution providers or merchants who modify or create hardware or software.
Encrypting PIN pad (EPP)	A device for secure PIN entry and encryption in an unattended PIN-acceptance device. An EPP may have a built-in display or card reader, or rely upon external displays or card readers installed in the unattended device. An EPP is typically used in an ATM or other unattended device (e.g., an unattended kiosk or automated fuel dispenser) for PIN entry and is controlled by a device controller. An EPP has a clearly defined physical and logical boundary, and a tamper-resistant or tamper-evident shell. Encrypting PIN pads require integration into UPTs or ATMs.

Term	Definition
Entry Device	A type of electronic device that interacts directly with and takes input from humans to facilitate mobile payment acceptance.
GPS (Global Positioning System)	A satellite communication system that provides location and time information.
Jailbreak/jailbroken	The rendering of a cell phone such that it is no longer subject to the limitations originally imposed on it by its manufacturers/proprietors. Jailbroken mobile devices allow access to their proprietary operating system, which then allows the installation of third-party applications not released or controlled by the manufacturer or proprietor. Also, see *Rooting.*
Malicious software/malware	Software designed to infiltrate or damage a computer system without the owner's knowledge or consent. Examples include viruses, worms, Trojans (or Trojan horses), spyware, adware, and rootkits.
Mobile app	A program for a phone, tablet, or other mobile electronic device.
Mobile device	A consumer electronic handheld device (e.g., smartphone, tablet, or wearable) that is not solely dedicated to payment acceptance for transaction processing and that has wireless connectivity to a network (e.g., cellular or Wi-Fi).
Near field communication (NFC)	A short-range, wireless RFID technology that makes use of interacting electromagnetic radio fields instead of the typical direct radio transmissions. Refer to ISO/IEC 18092 for specifications.
Object	A process, application, hardware device, or other identity over which access control is exercised.
PAN	Acronym for "primary account number" and also referred to as "account number." Unique payment card number (typically for credit or debit cards) that identifies the issuer and the particular cardholder account.
Rich OS	An environment created for versatility and richness where device applications—such as Android, Symbian OS, and Windows Phone for example—are executed. It is open to third-party download after the device is manufactured. Security is a concern here but is secondary to other issues.
Rooting	Gaining unauthorized administrative control of a computer system; also, see *Jailbreak/jailbroken.*
Secure Digital (SD) card/Micro-SD card	A non-volatile memory card format often used as additional memory for mobile devices.
Secure element	A formally certified, tamper-resistant, stand-alone integrated circuit often referred to as a "chip" as defined by the European Payments Council or other recognized standards authority.

Term	Definition
Sensitive authentication data (SAD)	Security-related information (including but not limited to card validation codes/values, full magnetic-stripe data, PINs, and PIN blocks) used to authenticate cardholders and/or authorize payment card transactions.
Side-channel leakage	An implementation-specific form of information leakage, usually from a cryptographic implementation, in a manner not considered in the data flow model of the implementation. It is generally an exploitation of physical leakages—e.g., power consumption, acoustical vibrations, or electromagnetic radiation. This can facilitate the determination of the secret key and the reconstruction of plaintext data.
Subscriber identity module (SIM)	A memory card that typically stores the IMSI (International Mobile Subscriber Identity) and other related information used to authenticate subscribers.
System's applications	The collection of apps and applications where "apps" refers to the software running on the mobile device and "applications" refers to the software running on the host system (e.g., servers or mainframe computers).
Trusted execution environment	An execution environment that runs alongside but isolated from an operating system. A trusted execution environment has security capabilities and meets certain security-related requirements. It: ▪ Protects trusted execution environment assets from general software attacks, ▪ Defines rigid safeguards as to data and functions that a program can access, and ▪ Resists a set of defined threats. Multiple technologies can be used to implement a trusted execution environment, and the level of security achieved may vary accordingly.
Trusted path	An unspoofable and incorruptible channel used to move data in and out of a trusted execution environment.
UPT (Unattended Payment Terminal)	A cardholder-operated device that reads, captures, and transmits card information in an unattended environment, including, but not limited to, the following: ▪ ATM ▪ Automated fuel dispenser ▪ Ticketing machine ▪ Vending machine

Appendix B: Best Practices and Responsibilities

The table below outlines each best practice described within this document along with who should be responsible for its implementation. The definitions of those entities that are responsible for the best practices include:

- **Device Manufacturer (DM):** Includes mobile-device manufacturers, integrators, firmware developers, and any manufacturer responsible for the development of any OEM hardware.
- **OS Developer (OD):** Includes the entity that creates and maintains the operating system, including but not limited to the entity responsible for OS architecture, device drivers, and patch development.
- **Application Developer (AD):** Includes any software developer that creates and maintains an application used as part of the payment-acceptance solution. This includes the merchant as an application developer.
- **Merchant as an End User (M):** Any entity that utilizes the mobile payment-acceptance solution to accept payments.
- **Mobile Payment-Acceptance Solution Provider (SP):** The entity that integrates all pieces in the mobile payment-acceptance solution and is responsible for the back-end administration of the solution. This includes the merchant as a solution provider.

Best Practice	DM	OD	AD	M	SP
1. Prevent account data from being intercepted when entered into a mobile device.	X	X	X		X
2. Prevent account data from compromise while processed or stored within the mobile device.	X	X	X		X
3. Prevent account data from interception upon transmission out of the mobile device.	X	X	X		X
4. Prevent unauthorized logical-device access.		X	X	X	X
5. Create server-side controls and report unauthorized access.		X	X	X	X
6. Prevent escalation of privileges.	X	X	X		X
7. Create the ability to remotely disable payment application.		X	X		X
8. Detect theft or loss.	X	X	X	X	X
9. Harden supporting systems.			X		X

Best Practice	DM	OD	AD	M	SP
10. Harden the application.			X		X
11. Prefer online transactions.			X		X
12. Conform to secure coding, engineering, and testing.		X	X		X
13. Protect against known vulnerabilities.		X	X		X
14. Protect the mobile device from unauthorized applications.	X	X	X		
15. Protect the mobile device from malware.		X	X	X	X
16. Protect the mobile device from unauthorized attachments.	X	X			
17. Create instructional materials for implementation and use.	X	X	X		X
18. Support secure merchant receipts.		X	X		X
19. Provide an indication of a secure state.	X	X	X		X
20. Provide audit and logging for user and device access.			X		X

Appendix C: Industry Documents and External References

Following are the sources of reference for this document.

1. ANSI X9.112-2016, *Wireless Management and Security — Part 1: General Requirements*.
2. *Best Practices for Mobile Device Banking Security.* ATM Industry Association (ATMIA). 2008.
3. CTIA-The Wireless Association®: *Best Practices and Guidelines for Mobile Financial Services*, Version 01.14.2009, Effective Date: January 28, 2009.
4. World Bank Working Paper No. 146, *Integrity in Mobile Phone Financial Services: Measures for Mitigating Risks from Money Laundering and Terrorist Financing,* May 2008.
5. NIST Special Publication 800-124, *Guidelines for Managing the Security of Mobile Devices in the Enterprise*, June 2013. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology. Gaithersburg, MD.
6. *Security of Proximity Mobile Payments* – A Smart Card Alliance Contactless and Mobile Payments Council White Paper, May 2009, Publication Number: CPMC-09001.
7. *White Paper Mobile Payments*, Version 4.0, 8th March 2017, Document EPC492-09.
8. NIST Special Publication 800-57, *Recommendation For Key Management,* March 2007. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology. Gaithersburg, MD.
9. ISO/IEC 11770-5:2011 *Information technology -- Security techniques -- Key management*
10. *OWASP Top 10 Mobile Risks*. OWASP Mobile Security Project, The OWASP Foundation. February 13, 2017. WWW.OWASP.ORG
11. "Biometric Standards Program And Resource Center". NIST, 2017, https://www.nist.gov/programs-projects/biometric-standards-program-and-resource-center.
12. *ANSI X9.84-2010 (R2017) - Biometric Information Management And Security For The Financial Services Industry*. American National Standards Institute, 2010.
13. European Union Agency for Network and Information Security. Smartphone Secure Development Guidelines. ENISA, 2016, p. all.
14. *Effective Daily Log Monitoring*. PCI Security Standards Council, May 2016.

About the PCI Security Standards Council

The PCI Security Standards Council is an open global forum that is responsible for the development, management, education, and awareness of the PCI Data Security Standard (PCI DSS) and other standards that increase payment data security. Created in 2006 by the founding payment card brands American Express, Discover Financial Services, JCB International, Mastercard, and Visa Inc., the Council has more than 700 Participating Organizations representing merchants, banks, processors, and vendors worldwide. To learn more about playing a part in securing payment card data globally, please visit: pcisecuritystandards.org.