Nets & Nexi Group; Nexi Digital Finland

Software Security Implementation Guide 24.3.0

Payment Card Core

Npay payment terminal application

Contents

1. VERSION HISTORY

Version	Date	Author	Description
1.4	2015-07-29	DFo	PA-DSS v3.1 related updates: - Explicit troubleshooting guidance - Retention policy clarified for merchants - Explicitly TLS 1.2 - Clarified centralized logging 5.6 - Updated Appendix A - Clarified distribution of this document - Updated 5.5 Key Management - Removed SW library versions - Clarify IG distribution process
1.4.1	2016-01-27	DFo	X.Y.pp and subtitle for the document
1.4.2	2016-04-05	DFo	Editorial
2.0	2017-06-29	JRä	Describe how masked PANs are generated Remove text that does not require customer actions Rename CardSvc to Payment Card Core Add Spire and Spica update descriptions Add Payment Frontend description Fix Payment Card Core version to 2.0.x Remove section Cardholder data handling as unnecessary Clarifications related to requirements 2.1, 2.2, 11.1, 11.2 Include versioning character set and clarify version numbering component descriptions Terminology correction Poplatek PA → Payment Card Core
2.0.1	2018-06-11	JRä	Update company: Poplatek Oy → Poplatek Payments Oy
2.2	2018-10-26	JRä	Update Payment Card Core version Update PAN masking description to match Payment Card Core 2.2.x
2.3	2019-12-04	JRä	Update Payment Card Core versions Update PTS references for Samoa terminals 5.3.1
3.0	2021-06-22	JRä	Update Payment Card Core version Update references to PCI DSS and PA-DSS requirements Update samoa updater endpoint Add Castles as terminal vendor Update dependencies 5.3.2 Fix typos Update document layout
3.1	2022-06-02	JRä	Update Payment Card Core version Remove the requirement for not to broadcast SSID in 4.2 Remove Spire and Worldline terminals from hardware dependencies in 5.3.1 | Remove update descriptions for Spire and Worldline terminals in 5.3.3 Add MP200 to hardware dependencies in 5.3.1 Update DNS name for required server in 3 Add VEGA3000 PTS 6.x to dependencies 5.3.1 Clarify differences between PAN masking in merchant and cardholder receipts
4.1	2023-07-05	JRä	Adapted from PCI PA-DSS to PCI SSF Change company name to Nexi Digital Finland Change Secure Software standard version to 1.2.1 Add missing IG-related requirements to Appendix A
4.2	2023-10-13	JRä	Change company name to Nexi
4.3	2023-11-28	JRä	Change Secure Software standard version to 1.2.1 Add missing IG-related requirements to Appendix A
4.3.1	2024-01-22	JRä	Documented non-security impacting change
24.0.0	2024-02-21 2024-03-26	JRä JRä	Added reference to control objective 11.2 Added reference to test requirement 5.2.d
24.1.1	2024-03-11 2024-04-29	JRä SVa	Added recommendation for min 23 character Wi-Fi password Added requirements for acceptable Wi-Fi networks
24.2.0	2024-05-24	JRä	Added details to terminal inspection instructions
24.3.0	2024-05-27	JRä	Add service code, expiry and vault encryption to CHD Handling chapter

2. INTRODUCTION

The Payment Card Industry Data Security Standard (PCI DSS) [1] specifies requirements for the configuration, operation and security of payment cards transactions. These requirements apply to organizations that store, process or transmit cardholder information and target to prevent credit card fraud and to increase security.

The PCI Secure Software Requirements [2] are defined to protect the integrity of payment transactions and the confidentiality of all sensitive data stored, processed, or transmitted in association with payment transactions.

The purpose of this Implementation Guide (IG) is to instruct Merchants and Resellers how to install and use Nexi Digital Finland (Nexi) payment terminals running Payment Card Core in Merchant’s environment in a PCI compliant manner. It is not intended to be a complete installation guide. Integrators may include e.g. Electronic Cash Register (ECR) vendors integrating Nexi payment terminal with their POS system using Nexi POS protocol [3], or Vending Machine vendors integrating Nexi payment terminal into the vending machine.

NOTE: Merchant/Reseller/Integrator responsibilities and actions are marked with ***MERCHANT ACTIONS*** in this document.

Version and review history is shown in section Version history. This section provides introduction, describes the review and update process, lists abbreviations and references. Payment Card Core usage is described in section Payment Application usage and the details are described in section Payment Card Core module including centralized logging. Troubleshooting information is provided in section Troubleshooting procedures. Appendix A lists and addresses all the PCI Secure Software requirements related to this document.

2.1 Document review and update process

Nexi must review this document on an annual basis and update it as needed to document all major and minor changes to the Payment Card Core module. The Secure Software scope is the Payment Card Core module only as Payment Card Core takes care of all sensitive data handling. Changes outside the Payment Card Core module need not to described or managed with the IG review process. If this scope is changed by QSA, this document must be updated to reflect the new scope. Also, this document is updated and reviewed in a timely manner whenever the PCI Secure Software Standard (PCI SSS) is updated.

Review process includes Nexi internal review by an individual other than the editor of the change knowledgeable of the Payment Card Core module internals. Document must be reviewed by a PCI Secure Software QSA, during the change control process with the QSA.

2.2 Distribution

This document is initially distributed to all customers and resellers latest with the first product delivery via customer support portal. Re-sellers and direct sales are also notified via e-mail about the roll-out of new software version. Support portal has release notes for the payment terminal software and release notes include link to the correct version of the IG. Whenever this document is updated, approved, and applicable to the payment terminals on the field with customers, the IG is uploaded to the support portal and the link to the document is updated in release notes.

Note also that the latest version of this guide can be obtained from technical support. The master document is stored into the Nexi internal version management system.

2.3 *Abbreviations*

Abbreviation	Meaning
CHD	CardHolder Data
ECR	Electronic Cash Register
IG	Implementation Guide
PAN	Primary Account Number
PCI DSS	Payment Card Industry Data Security Standard
PCI SSF	Payment Card Industry Secure Software Framework
PCI SSS	Payment Card Industry Secure Software Standard
POS	Point-of-Sale, used to refer to the system including payment terminal and ECR
QSA	Qualified Security Assessor
TLS	Transport Layer Security

3. PAYMENT APPLICATION USAGE

Strong access control measures must be used in all PCI scoped system components by using unique user Ids, strong passwords, and PCI DSS compliant secure access authentication, for more details see [1]. However, note that when using Nexi terminal there is no need to handle or store cardholder data outside the terminal.

In the Nexi payment terminal there are no user configurable Payment Card Core settings. Also, there are no user accounts (or administrative accounts) to be configured or any user passwords/credentials to be updated or reset.

Nexi payment terminal supports ECR integration with JSONPOS protocol. Nexi payment terminal can only be used with Nexi payment gateway.

Nexi payment terminal requires Internet connection for communicating with the payment gateway. Ethernet wiring can be used to provide the network connection and connection with the ECR. Cellular or Wireless LAN may be used for communication if wireless communication is preferred.

4. INSTALLATION ENVIRONMENT

4.1 *Payment Terminal handling*

The payment terminal must be installed according to Nexi installation instructions and along the Merchant actions requirements described in this document. Payment terminals must be periodically inspected for evidence of tampering and substitution (e.g. additions of card skimming devices) and merchant personnel must be trained for payment terminal inspections (see [4]). Also, an up-to-date list of payment terminals must be kept either by Merchant or provided by Nexi.

· ***MERCHANT ACTIONS***: Train personnel working with payment terminals on how to inspect payment terminals for evidence of tampering and substitution. The training must include at least the following:

- Verify the identity of any third-party persons claiming to be repair or maintenance personnel, prior to granting them access to modify or troubleshoot payment terminals.
- Do not install, replace, or return payment terminals without verification.
- Be aware of suspicious behavior around payment terminals (for example, attempts by unknown persons to unplug or open devices).
- Report suspicious behavior and indications of payment terminal tampering or substitution to appropriate personnel (for example, to a manager or security officer).

· ***MERCHANT ACTIONS***: Inspect payment terminals when received from shipping. The inspections should cover:

- Check that the delivery source information matches the purchase order
- Check that terminal quantity matches the delivery information
- Check that terminal product name and serial number matches the delivery information
- Inspect for any signs of tamper on the cartons, boxes, and/or the terminal If anything found abnormally during inspection, please contact your sales representative for detail checking.

· ***MERCHANT ACTIONS***: Periodically inspect payment terminals for evidence of tampering and substitution. The period can be based on Merchant’s own risk analysis. Check at least the following:

- Inspect chip card opening to make sure that there are not any untoward obstructions or suspicious objects at the opening
- Inspect magnetic stripe slot to make sure that not any other additional reader or inserted bugs
- Inspect appearance of device to make sure that not any tamper evidences. It is important checking especially for keypad and touchscreen area

· ***MERCHANT ACTIONS***: Merchant must have payment terminal registry for all its payment terminals. For Nexi payment terminals, this registry is provided by Nexi. Ensure Nexi has most up-to-date information about each payment terminal. The registry must include model of the device, location, and device serial number. Merchant must inform Nexi, whenever a payment terminal is relocated, decommissioned, removed, or added into production.

4.2 Network firewall configurations

There are no specific requirement on network segmentation when using Nexi payments terminal and when the Nexi payments terminal is the only medium used to read payment cards. See above.

· ***MERCHANT ACTIONS***: Nexi payment terminal uses external service provided only by Nexi. For the Nexi payment terminal the TCP port 443 for host pt.api.npay.eu to the Internet (outbound) must be opened. Also DNS resolution for the host must be allowed. In addition port 10001 must be allowed to connect to the payment terminal (inbound) from the ECR (when integrated with ECR). See protocols below.

4.3 Wireless LAN

If merchant uses wireless LAN to route the Nexi payment terminal connection to the Internet or the payment terminal uses wireless technology, it must be configured securely. This means that PCI DSS requirements must be followed when implementing the wireless networks:

· ***MERCHANT ACTIONS***:

- Payment terminal must only be connected to a trusted, merchant owned and controlled wireless LAN network. Shared networks, such as shopping center shared wireless LAN networks, must not be used regardless of encryption. Access to the wireless LAN network must be limited to trusted devices only.
- Encryption keys must be changed from default at installation, and must be changed anytime anyone with knowledge of the keys leaves the company or changes positions.
- Default SNMP community strings on wireless devices must be changed.
- Default passwords on access points must be changed. To protect against a brute-force attack, a truly random passphrase of 13 or more characters is recommended.
- Firmware on wireless devices must be updated to support strong encryption for authentication and transmission over wireless networks. WEP algorithm is not allowed.
- Any security-related wireless vendor defaults must be changed, if applicable.
- The default Service Set ID (SSID) must be changed.
- Firewall(s) must be installed between any wireless networks and systems that store cardholder data. This firewall(s) must be configured to deny or control (if such traffic is necessary for business purposes) any traffic from the wireless environment.

5. PAYMENT CARD CORE MODULE

5.1 Initial Payment Card Core distribution

The Payment Card Core is initially distributed along with the payment terminal or if absent, the payment terminal will install the Payment Card Core from the update server during first boot with network connection. In any case, during the boot up sequence the Payment Card Core will be updated from the update server if needed.

5.2 Versioning scheme for Payment Card Core module

The versioning scheme for the Payment Card Core module is <x>.<y>.<z>[.<n>] where the components are as follows:

x – version major, incremented for major changes like new terminal platform or major new feature

y – version minor, incremented on each security impacting change, reset to 0 when version major is incremented

z – hotfix, incremented on security impacting changes backported from main line version on top of the release line version, 0 used for mainline version

n – non security impacting change, this component is only present when release don't contain any security impacting changes after the previous version

Numbering for x, y and z starts from 0. Numbering for n starts from 1.

5.2.1 Hardware dependencies

Nexi payment terminal uses hardware provided by the Castles Technology Co. Ltd payment terminal manufacturer. Dependent hardware are:

Terminal type	PCI PTS
Castles Technology Co. Ltd VEGA3000	PTS Approval 4-30332 PTS-approved attended terminal
Castles Technology Co. Ltd V3P3	PTS Approval 4-80073 PTS-approved attended terminal

Dependent Castles operating system version is Castles Linux OS xx20.

Payment Card Core can be used in different configurations:

- Integrated with an attended ECR
- Standalone

5.2.2 Payment Card Core software dependencies

Nexi provides the payment application including Payment Card Core, which runs inside the terminal operating system on the terminals listed above. For developing the Payment Card Core, SDK from terminal manufacturers is required.

Nexi payment terminal supports connection to the network with Ethernet wiring or wireless using WLAN or cellular.

In addition Payment Card Core requires the some external software components during the software development process and running the Payment Card Core. These components are managed and updated by Nexi. These software components are linked into the software package and thus delivered as part of the payment terminal SW. Nexi takes care that these software components are up-to-date e.g. through a vulnerability management and software update processes and procedures.

5.2.3 CHD handling

Payment Card Core does not expose sensitive data in plain text. Full magnetic stripe data, chip equivalents or encrypted PIN block are never stored into non-volatile memory. PAN, full magnetic stripe data or chip equivalents and encrypted PIN block a is encrypted using RSA 2048 encryption prior to sending data to Nexi Payment Gateway over mutually authenticated TLS connection. RSA 2048 encrypted PAN is stored into Payment Card Core vault database. The RSA 2048 encrypted data can only be decrypted by the Nexi Payment Gateway. For receipts, PANs are masked (only the first six and last four digits are shown) and masked PANs are stored into the vault database. Encrypted PAN is removed from vault database after being sent to the Nexi Payment Gateway.

The service code of the magnetic stripe data and expiry date are stored to the vault data base and sent to the Nexi Payment Gateway over mutually authenticated TLS connection.

All data stored to the vault database is encrypted using AES-128 encryption using PCI PTS terminals generated random encryption key stored in the secure area of the terminal.

5.2.4 Protocols used by the Nexi payment terminal SW

Payment Frontend Nexi payment terminal SW uses TLS 1.2 or newer with strong cryptography to communicate with the payment frontend on TCP port 443. The payment terminal authenticates the frontend using a Nexi CA root, while the client is authenticated using OAuth2 tokens. All requests to the frontend are OAuth2-authenticated HTTPS requests which the frontend forwards based on the request URI. A Websocket connection, initialized using a HTTPS Upgrade header, is used to carry a JSON-RPC connection to the Payment Gateway, providing transaction data transfer, authorizations, and other payment related messaging.

The payment terminal communicates only with the Nexi payment gateway.

POS Protocol for ECR integrations The terminal uses Nexi JSONPOS protocol for communicating with the ECR and listens on TCP port 10001. The ECR initiates communication with the terminal. The JSONPOS protocol never transmits sensitive cardholder data to the ECR, PANs are masked (only the first six and last four digits are shown) for transaction receipt purposes.

Software Updates Payment application including Card Core is packed into a platform specific SW package that will be automatically updated to the payment terminals. Chapters below describe the details per platform.

Software Updates (Castles terminals) The payment terminal checks for updates and downloads update package(s) using JSON-RPC requests sent to the Payment Frontend which forwards them to the update server. The payment terminal reports its current software versions in an update check request, and the update server response indicates either that software is up-to-date or that specified updates need to be installed. The server is responsible for preventing unintended downgrades.

The update packages are SHA256 hash validated before installation, and the update package format itself contains a digital signature which Castles system software checks before installation. Software update packages are signed by Nexi under dual-control as specified by Castles.

5.3 Key Management

Payment terminal key management happens automatically. No user or merchant can have access to the payment terminal keys. There are no settings menu or other inputs to the terminal that would affect the key management. Software updates take care of updating keys, if ever needed. Also, the software updates happen automatically. No user or merchant actions are required.

The payment terminal uses OAuth2 to manage a refresh token and a bearer token. The first OAuth2 refresh token is obtained using an initial token fixed in the software build. When the first refresh token has been successfully taken into use, the initial token is no longer accepted; refresh tokens are then chained so that a new refresh token is fetched using the current refresh token. An administrator may manually allow a token refresh if a terminal loses its token state. The terminal requests for a token update on every boot (every 24h) which may update the refresh token and the bearer token. The bearer token is used for other HTTPS requests such as update checks, payment gateway connection, etc.

The PCI SSF scoped Payment Card Core modules uses RSA 2048-bit encryption to encrypt sensitive cardholder data. Only the payment gateway can decrypt the RSA encrypted data. The RSA public key is installed into the terminal from signature verified packet and updated automatically by Nexi.

5.4 Centralized logging

Nexi payment terminals implement centralized logging into the payment gateway. The centralized logging is based on a reliable event delivery protocol implemented by Nexi.

Logging is enabled automatically. Interfering with the logging functionality or disabling logs is not allowed and will result in non-compliance with PCI DSS. Note that it is not possible to disable logging from the payment terminal itself.

Merchants can obtain centralized logging events for their payment terminals on request from Nexi.

· ***MERCHANT ACTIONS***: If merchant needs logging data for some of its terminals containing Payment Card Core, contact Nexi.

6. TROUBLESHOOTING PROCEDURES

Nexi will never request Sensitive Authentication Data (SAD) from customers, including e.g. full PAN, in any situation, including possible troubleshooting cases. In some cases masked PAN (first six and last four digits) as printed on transaction receipt may be requested.

7. REFERENCES

[1] PCI Security Standards Council, 2018 Payment Card Industry (PCI) Data Security Standard, Requirements and Security Assessment Procedures, Version 3.2.1

[2] PCI Security Standards Council, 2023 Software Security Framework, Secure Software Requirements and Assessment Procedures, Version 1.2.1

[3] Nexi Digital Finland JSONPOS API https://poplapay.com/dev/jsonpos-api/

[4] PCI Security Standards Council, 2014 Information Supplement, Skimming Prevention: Best Practices for Merchants, Version 2.0

8. APPENDIX A: PCI SSS V1.2 REQUIREMENTS FOR THE IMPLEMENTATION GUIDANCE

| 2.2 | All software security controls, features, and functions are enabled upon software installation, initialization, or first use.

- Software doesn’t have any user configurable security options or parameters (2.2.c)

| 2.3 | Default authentication credentials or keys for built-in accounts are not used after installation, initialization, or first use

- Software doesn’t require any user actions to set authentication credentials or keys

| 3.1 | The software only retains the sensitive data absolutely necessary for the software to provide its intended functionality.

- There are no user configurable retention periods in the software (3.1.d)

| 3.2 | Transient sensitive data is retained only for the duration necessary to fulfill a legitimate business purpose.

- There are no user configurable options for retention of transient sensitive data in the software (3.2.d)

| 3.3 | The software protects the confidentiality and integrity of sensitive data (both transient and persistent) during retention.

- There are no user protections methods requiring user input or interaction (3.3.e)

| 3.6 | The software does not disclose sensitive data through unintended channels

- There are no user protections methods requiring user input or interaction (3.6.c)

| 4.2 | Software security controls are implemented to mitigate software attacks.

- There are no user input or interaction that could be used to disable, remove, or bypass any mitigations for software attacks. The mitigations are enabled by default and there is no possibility for the user to query status of the mitigation contols (4.2.c)

| 5.1| Access to critical assets is authenticated.

- The software doesn’t allow any non-console access to the system on which the software is executed or directly to the software itself. Thus there are no options to configure authentication mechanisms (5.1.c)

| 5.2 | Access to critical assets requires unique identification.

- Payment Card Core does not provide API that would allow for automated access to critical assets (5.2.b)

- Payment Card Core users’s don’t have access to any critical assets and thus don’t have identification or authentication parameters needed for unique identification to critical assets. (5.2.d)

| 6.2 | Sensitive data is secured during transmission.

- The software doesn’t rely on third-party or execution-environment features are upon for the security of the transmitted data (6.2.c)

| 6.3| Use of cryptography meets all applicable cryptography requirements within this standard.

- The are no user configurable options for the cryptographic methods used by the software and cryptography used by the software follows the industry standard (6.3.b)

| 7.2 | The software supports industry standard key management processes and procedures. Industry standard key management processes and procedures are those recognized by industry standards bodies, such as NIST, ANSI, and ISO.

- Payment Card Core doesn’t use any external key files or certificates (7.2.f)

| 7.4 | Random values have entropy that meets the minimum effective strength requirements of the cryptographic primitives and keys that rely on them.

- Payment Card Core does not have any cryptographic keys generated through processes which require direct user interaction (7.4.b)

| 8.1 | All access attempts and usage of critical assets are tracked and traceable to a unique user.

- Payment Card Core doesn’t provide console or non-console access to the critical assets

| 8.3 | The software supports secure retention of detailed activity records.

- Payment Card Core does not use 3rd party services for maintenance of tracking data. Tracking data is delivered to Nexi PCI-DSS certified Payment Gateway only. (8.3.b)

| 9.1 | The software detects and alerts upon detection of anomalous behavior, such as changes in post deployment configurations or obvious attack behavior.

- Payment Card core doesn’t have any user accounts or cryptographic-key input fields (9.1.f)

| 11.2 | Software releases and updates are delivered in a secure manner that ensures the integrity of the software code.

- Payment Card Core software is updated over the air by Nexi and no user input or interaction is required to validate the integrity of the software code (11.2.b)

| 12.1 | The software vendor provides stakeholders with clear and thorough guidance on the secure implementation, configuration, and operation of its payment software.

- The distribution of the document is described in chapter Distribution
- Software installation is fully controlled by Nexi and the end user can’t do any SW installations to the terminal
- Terminals don’t have user accounts and user can’t control any security features of the terminal
- Software update mechanism is described in Software Updates
- Key management described in Key Management

| A.2.1 | The software vendor provides guidance to stakeholders regarding secure deletion of cardholder data after expiration of defined retention period(s).

- No user actions are needed for secure deletion of cardholder data

| A.2.2 | The software provides features to restrict or otherwise mask all displays of PAN to the minimum number of digits required.

- No user input or interaction is required to configure PAN masking features and options (A.2.2.c)

| A.2.3 | PAN is rendered unreadable anywhere it is stored (including data on portable digital media, backup media, and in logs)

- No user input or interaction is required to configure methods to render PAN unreadable when stored (A.2.3.b)

| B.2.8 | All software files are cryptographically signed to enable cryptographic authentication of the software files by the payment terminal firmware.

- Software for the terminal can only be signed by the terminal vendor or the software vendor (B.2.8.a)

| B.2.9 | The integrity of software prompt files is protected in accordance with Control Objective B.2.8.

- The prompt files are built in into the signed software package (B.2.9.b and B.2.9.c)

| B.5.1 | The software vendor provides implementation guidance on how to implement and operate the software securely for the payment terminals on which it is to be deployed

- This document provides the relevant information

| B.5.1.1 | Implementation guidance includes detailed instructions for how to configure all available security options and parameters of the software.

- Software doesn’t have any user configurable security options or parameters

| B.5.1.2 | Implementation guidance includes detailed instructions for how to securely configure the software to use the security features and functions of the payment terminal where applicable

- Software doesn’t have any user configurable security options or parameters

| B.5.1.3 | Implementation guidance includes detailed instructions for how to configure the software to securely integrate or use any shared resources provided by the payment terminal.

- There is no user accessible configuration for any shared resources provided by the payment terminal

| B.5.1.4 | Implementation guidance includes detailed instructions on how to cryptographically sign the software files in a manner that facilitates the cryptographic authentication of all such files by the payment terminal.

- Software packages can’t be signed by the customers, only by Nexi and the terminal doesn’t allow installation of unsigned software packages

| B.5.1.5 | Implementation guidance includes instructions for stakeholders to cryptographically sign all prompt files.

- Prompt files are only signed by Nexi

| B.5.2 | Implementation guidance adheres to payment terminal vendor guidance on the secure configuration of the payment terminal.

- Software doesn’t have any user configurable security options or parameters