neopost

*PSD MODEL C22,C28,C35, C56*

# SECURITY POLICY

Version 8.0

## Distribution List

| Reference of the document (identification and localization): PSD – OTH – SecurityPolicy - Canada.doc (eRoom)R&D (Bagneux)  > Technical Architecture  > Infogard > Security Policy > PSD | | | |
|---|---|---|---|
| **Writer** | | | |
| ROLE | NAME | DATE | VISA |
| Technical Architect | Nathalie Tortellier/Setonnougbo Hodonou/Adriana Rosca/Yves Latu | | |
| **Validation List** | | | |
| ROLE | NAME | DATE | VISA |
| Technical Architect | Setonnougbo Hodonou | | |
| PSD Software Manager | Cubilier Christophe | | |
| Project Manager | Dany Ray | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| **Diffusion List** | | | |
| Validation list, plus : | | | |
| ROLE | NAME(S) | | |
| | | | |
| | | | |

# List of Changes

| REVISION | DATE | REVISION CAUSE, MODIFICATION |
|----------|------|------------------------------|
| 1.0 | 01/04/2009 | Creation |
| 2.0 | 15/09/2009 | Update after review with InfoGard |
| 3.0 | 16/03/2010 | Update the PSD firmware version: 22.12.2 |
| 4.0 | 06/11/2010 | Add new model number C28 & C35 |
| 5.0 | 10/02/2011 | Update the PSD firmware version: 23.6 |
| 6.0 | 07/06/2011 | Append a new PSD PCB part number |
| 7.0 | 29/07/2011 | Add new model number C56 |
| 8.0 | 27/03/2012 | Update the PSD firmware version: 23.8 |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| The writer of the document is responsible for sending the last validated revision to the whole distribution list. | | |

# Contents

# Figures

# 1   Introduction

## 1.1   *System Overview*

The Neopost Postal Secure Device (PSD) is a module embedded within the Alpha, Delta and Omega postal franking machines. Integrated within the PSD is a cryptographic sub function and postal services sub function.

The postal services relate to the ultimate objective of the PSD which is to store postage credit belonging to a customer until it is needed by the indicium dispensing system of the franking machine. The indicia are dispensed in the form of an image containing digitally signed data. This image is a unique bit pattern that can be determined to have originated from a particular PSD at a particular point in time.

The cryptographic functions are used to restrict access to postal services and to authenticate, where necessary, postal service output.

The module configuration under FIPS 140-2 validation is:

**Canadian PSD Configuration:**

- o   Hardware
  - ▪   P/N 4129955LD
  - ▪   P/N 4150859LB
- o   Firmware
  - ▪   P/N 4151948VB Version 23.08
  - ▪   P/N 4151948VA Version 23.06
  - ▪   P/N 4148747LA Version 22.12.2

## 1.2   *Purpose of the Document*

This document contains a statement of the security rules under which the PSD module must operate. A number of these rules are wholly or partially a consequence of the general franking machine environment in which the PSD is intended to be placed and for this reason a brief description of this environment is included.

## 1.3   *Reference Documents*

General Documents (procedures, guides, templates, manuals, etc.):

| Id | Title | Reference | Accountable |
|----|-------|-----------|-------------|
| 1 | Digital Meter Indicia Specification | Spec3457 V1.2 | Canada Post Corporation |
| 2 | Postage Meter Product Information Handling Requirement | V2.0 | Canada Post Corporation |
| 3 | Postage Server Product Information Handling Requirement | V2.0 | Canada Post Corporation |

| Id | Title | Reference | Accountable |
|---|---|---|---|
| 4 | Security Requirements for Cryptographic modules, Federal Information Processing Standards Publication | FIPS140-2 | NIST |
| 5 | The TLS Protocol Version 1.0 | RFC 2246 | IETF |
| 6 | Information Technology - Open Systems Interconnection - The Directory: Authentication Framework | ITU-T Recommandation X509 | Telecommunication Standardization Sector of ITU |
| 7 | Secure Hash Standard, Federal Information Processing Standards Publication | FIPS 180-2 | NIST |
| 8 | RSA Cryptography Standard | PKCS #1 v1.5 | RSA Laboratories |
| 9 | RSAES: RSA – Encryption Scheme | PKCS #1 v1.5 | RSA Laboratories |
| 10 | ANSI X9.31 - Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA) - Appendix A.2.4 (RNG using AES) | ANSI X9.31 | ANSI |
| 11 | Specification for the Advanced Encryption Standard (AES), Federal Information Processing Standards Publication | FIPS 197 | NIST |
| 12 | ANSI X9.62 Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA) | ANSI X9.62 | ANSI |
| 13 | Keyed-Hash Message Authentication Code - HMAC | FIPS 193 | NIST |

## 1.4 Terminology

| Abbreviation | Description |
|---|---|
| AES | Advanced Encryption Standard (Reference [11]) |
| CPC | Canada Post Corporation |
| CSP | Critical Security Parameter |
| EFP | Environmental Failure Protection |
| EFT | Environmental Failure Testing |
| EMI | Electromagnetic Interference |
| EMC | Electromagnetic Compatibility |

| Abbreviation | Description |
| --- | --- |
| FIPS | Federal Information Processing Standards Publication : US standards to be complied with to get access to US administrative and governmental markets |
| I/O | Input / Output |
| MTBF | Mean time between failures |
| NIST | National Institute of Standards and Technology : Part of US Department of Commerce in charge of publishing and managing FIPS standards |
| NVEM | Non Volatile Electronic Memory |
| PSD | Postal Secure Device |
| PKI | Public Key Infrastructure |
| RNG | Random number generator |
| RSA | An algorithm for encryption/decryption and signature/verification (Rivest Shamir Adleman) (Reference [8]) |
| SHA | Secure Hash Algorithm (Reference [7]) |
| ECDSA | Elliptic Curve Digital Signature Algorithm |

# 2  Security Level

The PSD is a multi-chip embedded cryptographic module as defined in FIPS140-2 (reference [4]). The PSD module shall meet the overall requirements for Level 3 security as defined in reference [4]. The following table shows the security level requirement, as defined in reference [4], for each area of the PSD:

| Security Requirements Section | Level |
|---|---|
| **Cryptographic Module** | 3 |
| **Cryptographic Module Ports and Interfaces** | 3 |
| **Roles, Services and Authentication** | 3 |
| **Finite State Machine** | 3 |
| **Physical Security** | 3 + EFP/EFT |
| **Operating System Security** | N/A |
| **Cryptographic Key Management** | 3 |
| **EMI/EMC** | 3 |
| **Self Tests** | 3 |
| **Design Assurance** | 3 |
| **Mitigation of Other Attacks** | 3 |

N/A = not applicable

# 3   PSD Module Overview



**Top**

**Bottom**

*Figure 1      – Postal Security Device module (PSD)*

The PSD (Figure 1) consists of a cryptographic sub function and postal services sub function sharing common hardware that is contained on a printed circuit board and enclosed within a hard, opaque, plastic enclosure encapsulating the epoxy potted module which is wrapped in a special tamper detection envelope with a response mechanism that causes the zeroization of plaintext CSPs. This enclosure constitutes the cryptographic physical boundary.

The PSD contains dual redundant non-volatile electronic memories, which enables both critical security parameters and postal related data items to be stored in duplicate if required. Duplicate storage is typically used to increase MTBF.

The PSD will input and output authenticated data which requires the services of the cryptographic sub function.  The PSD will also input and output certain other data that has no security implications and that is permitted to pass freely across the cryptographic physical boundary. This latter data relates to the general control and use of the franking machine in which the PSD is embedded.

## 3.1   I/O Port

To enable communication with a base, the module provides a 10-pin serial communication physical edge connector (RS232 communicating at 921,600 bits per second). Power input, data input, data output, control input, and status output interfaces are logically assigned. The base is the main function that controls a transport motor, an indicium dispensing system, a display and a keypad. Keys and CSPs are always input and output from the module encrypted through the serial interface. No plaintext CSPs are input or output from the module through this serial interface.

| PIN | Description | Interface Type |
|---|---|---|
| 1 | Ground | |
| 2 | Ground | |
| 3 | RX | Data In/Control |
| 4 | RX | Data In/Control |
| 5 | TX | Data Out/Status |
| 6 | TX | Data Out/Status |
| 7 | Power (5V – 17V) | Power |
| 8 | Power (5V – 17V) | Power |
| 9 | Ground | |
| 10 | Ground | |

*Table 1 – Interface*
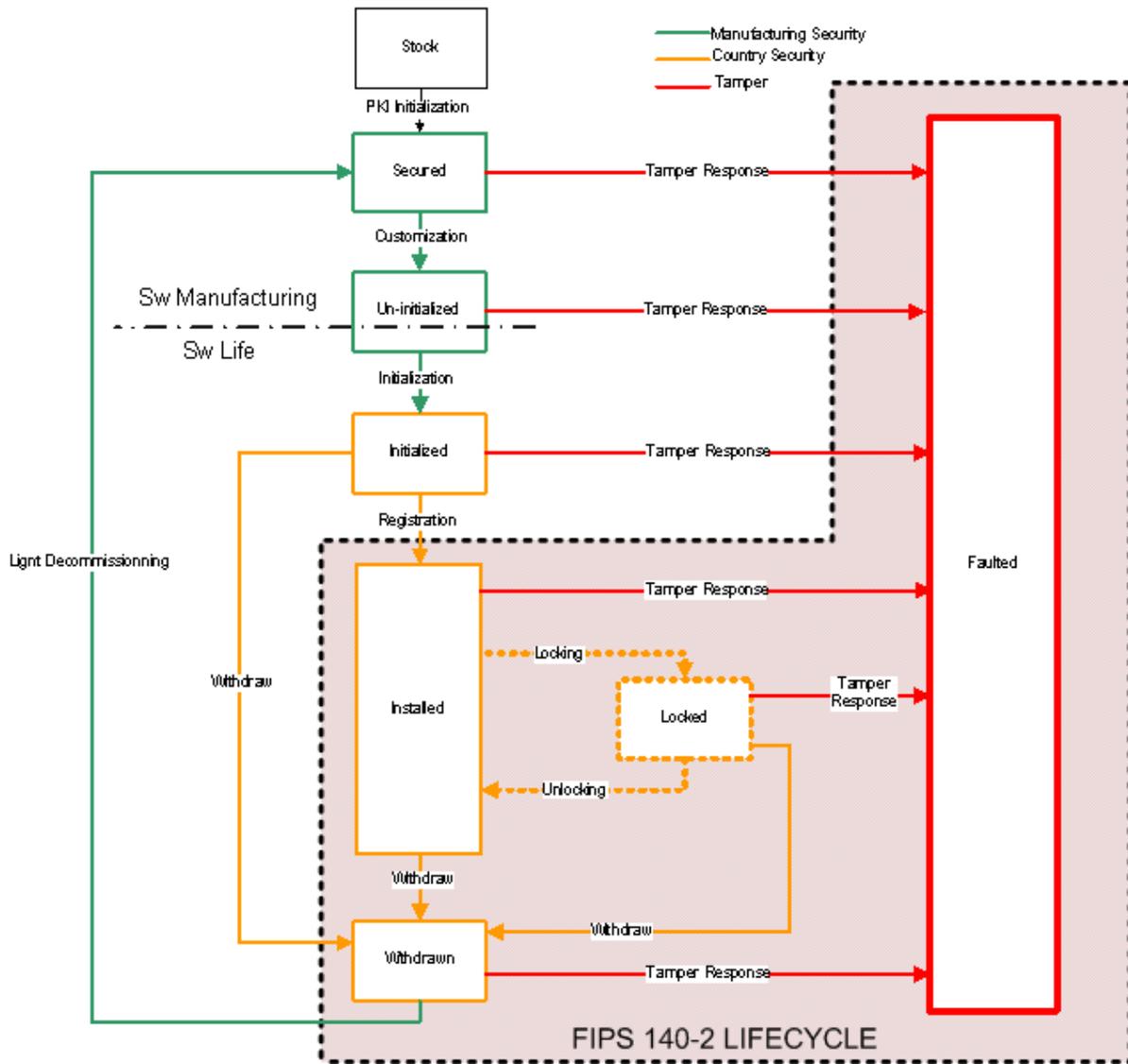
## 3.2 Lifecycle States



*Figure 2 – PSD Lifecycle*

The PSD assumes one of seven main overall states during its lifecycle. These states are relevant to the accessibility of cryptographic services. The states are:

- **Stock**

  This is the default at manufacture. The PSD does not contain the cryptographic parameters necessary to support interaction with the Neopost Infrastructure. A PKI-initialisation is required.

- **Secured**

  The PSD contains the cryptographic parameters necessary to support interaction with the Neopost Infrastructure but has not yet been configured for a country.

- **Uninitialized**

  The PSD has been configured for a country but does not contain the postal cryptographic parameters necessary to this country. An initialisation with the corresponding Neopost Postal Infrastructure is required.

- **Initialized**

  The PSD contains the cryptographic parameters necessary to this country but has not yet been registered with the corresponding Neopost Postal Infrastructure.

**Module's FIPS 140-2 Lifecycle States**

- **Installed**

  The PSD is registered with the corresponding Neopost Postal Infrastructure and may perform postal functions.

- **Locked**

  The PSD is temporarily locked and can't perform printing.

- **Withdrawn**

  The PSD is withdrawn from the corresponding Neopost Postal Infrastructure and may not perform postal functions. It cannot go back to the 'Initialised' state until it has undergone a factory process, which will reconfigure the contents of the PSD system memory.

- **Faulted**

  The PSD is faulted due to Zeroization service or from a tamper response and may not perform cryptographic functions. The only available function is Get Status in an un-authenticated way.

**In this document, only the FIPS lifecycle states of the PSD are described.**

# 4 Modes of Operation

## 4.1 *Approved mode of operation*

The cryptographic module supports the following FIPS Approved algorithms:

- RSA with variable key sizes (1536 or 2048 bit key sizes) for

  o Sign(1536 bit key size) / Verify(2048 bit key sizes) – Signature/Verification of signed X509 certificates used by TLS v1.0 Handshake protocol for authentication

  o Verify – verification of signed files imported into the module (1536 bit key size)

  o Key Wrapping of HMAC key output from the module (1536 bit key size)

  (See RSA Validation Certificate # 260)

- AES CBC with 128 key size for

  o Encryption/Decryption of:

    ▪ CSPs for storage within the module

    ▪ Data exchanged with the franking machine base or servers using the TLS protocol

  (See AES Validation Certificate # 563)

- HMAC-SHA-1

  o Authentication of indicia dispensed by the franking machine base

  (See HMAC Validation Certificate # 300)

- SHA-1, SHA-256 for hashing

  o Authentication of TLS messages

  (See Secure Hash Standard Validation Certificate # 629)

- ECDSA P192 for

  o Authentication of indicia dispensed by the base

  (See ECDSA P192 Validation Certificate # 62)

**Furthermore, the Alpha, Delta, Omega PSD module offers key generation services:**

  o RSA key pair generation

  o ECDSA key generation

  o HMAC key generation

For random value generation and generation of all cryptographic keys listed above, the PSD module relies on an implemented deterministic random number generator (DRNG) that is compliant with ANSI X9.31 with 16 bytes seed (externally generated by FIPS validated module and imported into the module in secure factory environment) which has 90-bits of entropy and based on AES as transition function. This DRNG is FIPS Approved; see Random Number Generator Validation Certificate # 328.

The Alpha, Delta, Omega PSD module also implements and uses the following non-Approved but allowed algorithms in FIPS mode:

  o RSA Key Wrapping (1536 bit key size) (key establishment methodology provides 96 bits of security of encryption strength)

  o Diffie Hellman used for key agreement of TLS master secret during TLS Handshake protocol (DH $Y_{Client/Server}$ = 1280 bits) (key establishment methodology provides 90 bits of security)

## *4.2   Non-FIPS mode of operation*

The module always operates in Approved mode of operation.

# 5   Roles, Services and Authentication

The PSD assures an identity-based authentication. The PSD shall support different identities. Each identity has one or several roles. The PSD authorizes transactions according to the state of the module and services according to the role. Transactions are defined for the rest of this document as an ordered sequence of services.

These allowable identified entities are:
- Expertise Tool
- Customer (Printing Base)
- Neopost Administrator (Field Server)
- File Signer Tool

Except for "Expertise Tool" which is unsecured, the identity-based authentication is a certificate-based authentication with TLS secure protocol.

The following table gives the link between the type of authentication and the identified entity for the PSD:

| IDENTIFIED ENTITY ▶<br><br><br>AUTHENTIFICATION ▼ | EXPERTISE TOOL | NEOPOST ADMINISTRATOR | CUSTOMER (PRINTING BASE) | FILE SIGNER TOOL |
|---|---|---|---|---|
| **No Authentication (Unsecured)** | ✔ | | | |
| **Two-Way TLS Authentication (Secured)** | | ✔ | ✔ | |
| **Authentication by signature and certificate chain (Secured)** | | | | ✔ |

The allowable roles of the system are:
- Printing Base
- Postal User
- Postal Crypto-Officer

- Field Crypto-Officer
- R&D Signer
- Region
- Root
- Unauthenticated User

The following table gives the link between the identity and role for the PSD:

| ROLE ▸ / IDENTIFIED ENTITY ▾ | PRINTING BASE | POSTAL USER | POSTAL CRYPTO OFFICER | FIELD CRYPTO OFFICER | R&D SIGNER | REGION | ROOT | UNAUTHENTICATED USER |
|---|---|---|---|---|---|---|---|---|
| **Expertise Tool** | | | | | | | | ✔ |
| **Customer (Printing Base)** | ✔ | | | | | | | |
| **Neopost Administrator** | | ✔ | ✔ | ✔ | | ✔ | ✔ | |
| **File Signer Tool** | | | | | ✔ | | | |

Where transactions have a state dependency, the PSD must be first placed into an appropriate FIPS lifecycle state. The relationship between PSD transactions and states is summarised in Section 11 - Appendix A: Relations transactions/states.

The relationship between PSD services and roles is summarised in Section 12.- Appendix B: Relation services/roles

## 5.1 Neopost Administrator
### (Crypto-Officer roles: Postal User, Region, Root, Postal Crypto-Officer and Field Crypto-Officer)

The Neopost Administrator has available some transactions necessary to control, sustain, and monitor the postal operation of a PSD – Postal Core Services (i.e. postage funding, usage auditing, withdrawal, etc.). The transactions require the PSD to be in an appropriate state (see Section 11 - Appendix A: Relations transactions/states).

The Neopost Administrator can have the following Crypto-Officer roles and associated services:

Postal User role:  The services available to the Postal User role are:

- Postal Core Services
  - o Resetting Transaction
  - o Update Registration Transaction
  - o Withdraw Transaction
  - o Rekey Transaction
- Read Status Data

Postal Crypto-Office role: The service available to the Postal Crypto-Officer is:

- Generate Stamp Key
- Set Stamp Info (Server Indicia Signature)

Field Crypto-Officer role: The services available to the Field Crypto-Officer role are:

- Generate PKI Key
- Get PKI Certificate
- Set PKI Certificate

Root role: The services available to the Root role are:

- Verify Region Certificate
- Verify Root Certificate

Region role: The service available to the Region role is:

- Verify Device Certificate

The Neopost Administrator has available some services (included in previous transactions) to initialize and maintain the parameters within the PSD that are necessary for interaction with the Neopost infrastructure.

For each transaction, the Neopost Administrator and the PSD are authenticating each other (with the TLS secure protocol).

### 5.1.1 Resetting

This transaction will carry out the following actions:

- The PSD sends its postal data
- The PSD sends its postal statistics
- The PSD generates the stamp keys if necessary (using the services "Generate Stamp Key" for ECDSA and HMAC keys and using the RSA Stamp Transport Public Key for HMAC key)
- Adjust the date and time of the PSD
- Set the funds to the PSD

### 5.1.2 Update Registration

This transaction will carry out the following actions:

- The PSD sends its postal data
- The PSD receive the stamp information (Server Indicia Signature)
- Update the information linked to the country or to the customer to the PSD

### 5.1.3 Withdraw

This transaction will carry out the following actions:

- Set the PSD state to WITHDRAWN
- The PSD sends its postal data to the server

### 5.1.4 Rekey

This transaction will carry out the following actions:

- If necessary, generate a new communication key (using the service "Generate PKI Key") (see Section 7 - Definition of Critical Security Parameters (CSP))

- Update the chain of certificates (using the service "Get PKI Certificates")
- If necessary, attach to a new region on PKI infrastructure (using the service "Set PKI Certificates")

## 5.2 *File Signer Tool (Other role: R&D Signer role)*

The File Signer Tool can have the following role and associated service:

R&D Signer role: The service available to the R&D Signer role is:
- Verify Files:  The file signer tool shall provide the service required to sign the files used by the franking machine. The PSD verifies the signature of the signed files.

## 5.3 *Expertise Tool (Other role: Unauthenticated User role)*

The expertise tool shall provide the services required to read postal data of the machine, in a unauthenticated way.

The Expertise Tool can have the following role and associated service:

Unauthenticated User role: The service available to the Unauthenticated User role is:
- Read Status Data
- Zeroize CSPs

### 5.3.1 Read status data

This service will carry out the following actions:

- The PSD sends its statistical data (these data are linked to the imprints or the errors)

- The PSD sends its registers

- The PSD sends its lifecycle state

### 5.3.2 Zeroize CSPs

This service will carry out the following actions:

- Overwrites with 0s the memory location of plaintext CSPs

  - Master Secret Key,

  - Random Number Seed and

  - Random Number Key

- Irreversibly moves the PSD into "Faulted" state

## 5.4 *Customer (User role: Printing Base role)*

The base shall provide the services required to use the franking machine.

The Customer can have the following role and associated services:

Printing Base role: The services available to the Printing Base role are:
- Initiate/End Postal Core Connection
- Initiate/End Rekey Connection
- Postal Indicium

- Other Base Services
- Read Status Data

For each of the services, the Printing Base and the PSD authenticate each other (with the TLS secure protocol).

### 5.4.1 Initiate/End Postal Core Connection

This service will carry out the following actions:
- Initiate Postal Core Connection: the customer initiates a transaction (Resetting, Update Registration, Withdraw); the PSD will connect to the server via the base (see §5.1)
- End Postal Core Connection: the base gets a connection result on the previous transaction to inform the customer.

### 5.4.2 Initiate/End Rekey Connection

This service will carry out the following actions:
- Initiate Rekey Connection: when the PSD needs to update its certificate chains, the base initiates a Rekey transaction; the PSD will connect to the server via the base (see §5.1)
- End Rekey Connection: the base gets a connection result on the previous rekey transaction to inform the customer.

### 5.4.3 Postal Indicium

This service will carry out the following actions:
- Send bitmap to print
- Send print authorization
- Send data to print

### 5.4.4 Other Base Services

This service will carry out the following actions:
- Time adjustment
- Download bitmaps
- Verify files signature
- Read date and time

### 5.4.5 Read Status Data

This service is the same as §5.3.1 Read status data but is accessible in an authenticated way.

# 6 Security Rules

## *6.1 Authentication Rules*

- The PSD shall provide identity-based authentication.

- The authentication of the user entity (Customer and Neopost Administrator) by the PSD is based on the TLS Secure Protocol using the "**TLS-DHE-RSA-WITH-AES-128-CBC-SHA**" cryptographic suite, with variable RSA key length (1536 to 2048 bits).

- The authentication of the PSD by the user entities (Customer and Neopost Administrator) is based on TLS Secure Protocol using of the "**TLS-DHE-RSA-WITH-AES-128-CBC-SHA**" cryptographic suite, with variable RSA key length (1536 to 2560 bits).

- The random number implementation employed shall be according to ANSI X9.31 using AES algorithm (Reference [10]).

- For any attempt to use the authentication mechanism, the probability that a random attempt will be accepted or that a false acceptance will occur will be at least 1 in $2^{96}$ (equivalent to at least 12 x $10^{28}$).

- The RSA key is 1536 bits and is considered to have 96 bits of strength. This is considerably more difficult to break than the 1 in 1,000,000 requirement.

- The maximum time to generate an authentication shall be 100ms.

- For multiple attempts to use the authentication mechanism then the probability that a random attempt will be accepted or that a false acceptance will occur will be 1 in $2^{96}$ divided by 600 (equivalent to 1 x $10^{26}$). This is considerably more difficult to break than the 1 in 100,000 requirement.

## *6.2 CSP/Key Overview*

There are three groups of keys and CSPs in the PSD.

**Communication keys:** These keys are used by the TLS protocol to authenticate the messages exchanged with the franking machine base or with the Neopost infrastructure (servers), and to establish a session secret keyset used for encrypting and authenticating the communication to and from the PSD as dictated by the TLS protocol.

**RNG keys:** These are keys used to initialize the approved pseudo-random number generator.

**Postal keys:** These are keys used by the PSD to provide and perform postal functionality.

(Please refer to Section 7.0 for further details).

## *6.3 TLS Configuration*

The PSD is always the client in the TLS negotiation. The server in the TLS negotiation is either the base, the Neopost Region server or the Neopost Root server. The TLS protocol is configured to use the following parameters during the handshake negotiation:

**Clent: ClientHello message**

- SessionID length = 0 (indicates that session-reuse is not allowed)

- ClientRandomValue (32 bytes) is generated by concatenation of the PSD time (4 bytes) and a random number generated by FIPS Approved PRNG.

- There is only one CipherSuite advertised to the server: **TLS-DHE-RSA-WITH-AES-128-CBC-SHA** to indicate that the server will use RSA to sign/verify Diffie Hellman parameters used between the client and server to agree on an premaster secret that will be used in turn to form the 128 bits AES encryption key (TLS Communication Secret Keyset) followed by a HMAC-SHA-1 for each message.

**Server: ServerHello message**

- ServerRandomValue (32 bytes) is generated by concatenation of server time (4 bytes) and random number generated by FIPS Approved PRNG.

- The CipherSuite field is selected from the set of ciphers advertised by the client: **TLS-DHE-RSA-WITH-AES-128-CBC-SHA**

**Server: Certificate message**

- The server will present to the client a chain of signed X509 certificates that will be used for authentication and selection of roles.

## *6.4 PSD Key Generation*

### 6.4.1 RSA Key Generation

RSA keys are used for authentication of the communication (communication keys of TLS protocol) with a length of 1536 bits. The PSD RSA Private keys shall be generated according to PKCS #1 v1.5 RSA Cryptography Standard

- The PSD RSA public key corresponding to the private key shall be calculated according to the relationship for derivation of a RSA public key defined in reference [8]
- During private/public key pair generation data output from the PSD shall be inhibited.

### 6.4.2 HMAC-SHA-1 Key Generation

The SHA1-HMAC Secondary Stamp key is used as an integrity mechanism within the indicia with a length of 160 bits.

- The PSD SHA1-HMAC secret key shall be a random number according to ANSI X9.31 – using the AES algorithm (Reference [10]).

- During secret key generation data output from the PSD shall be inhibited

### 6.4.3 ECDSA Key Generation

The ECDSA keys are used as the authentication mechanism within the indicia (stamp keys) with the curve P-192.

- This PSD ECDSA Private Stamp key shall be generated according the ECDSA algorithm reference [12]

- The PSD ECDSA public key corresponding to the private key shall be calculated according to the relationship for derivation of a ECDSA public key defined in reference [12]

- During private/public key pair generation data output from the PSD shall be inhibited

## *6.5 Conditional Self Test Rules*

- If either of the RSA or ECDSA key pair is invalid then all CSPs shall be zeroized and the PSD shall pass in the Faulted state. The validity of a key pair shall be determined by a pairwise consistency check, i.e. the encryption of a plaintext value and decryption of the cipher text value. This check shall be performed at the generation of each new key pair and at power up.

- Before generating the private/public keys and secret keys using the Approved random number generator, the PSD shall perform the continuous random number generator test, as defined in reference for conditional self-tests, for every number generated and shall pass in the Faulted state if it fails.

## *6.6 Power Up Self-Test Rules*

- The PSD shall test the operation of RAM areas used for secure operations at power up. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the contents of it's program memory area at power up by calculating the hash (SHA256) of the contents and comparing the result with a known answer. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the accessibility and validity of all keys and CSP values in NVRAM at power up. If any are not accessible (i.e. device failure) or contain erroneous data (16 bit EDC fails) then the PSD shall pass in the Faulted state.

- The PSD shall test the RSA algorithm at power up by performing a known answer test for both encryption / decryption (wrapping) and signature/verification using predetermined data embedded into the PSD firmware.

- The PSD shall test the DHE algorithm at power up by performing a known answer test for calculation of keys using predetermined data embedded into the PSD firmware. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the SHA-1 algorithm at power up by performing a known answer test for hashing predetermined data embedded into the PSD firmware. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the SHA-256 algorithm at power up by performing a known answer test for hashing predetermined data embedded into the PSD firmware. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the AES algorithm at power up by performing a known answer test for both encryption of a plaintext value and decryption of the cipher text value using predetermined data embedded into the PSD firmware. The PSD shall pass in the faulted state if the test fails.

- The PSD shall test the PRNG algorithm at power up by performing a known answer test for activating algorithm using predetermined data embedded into the PSD firmware. The PSD shall pass in the Faulted state if the test fails.

- The PSD shall test the accessibility and validity of all public keys values (communication keys, transport keys, stamp keys, certificates) in NVRAM at power up. If any are not accessible (i.e. device failure) or contain erroneous data then the PSD shall pass in the Faulted state.

- The PSD shall test the ECDSA P192 algorithm at power up by performing a known answer test for signature/verification using predetermined data embedded into the PSD firmware.

- The PSD shall test the HMAC algorithm at power up by performing a known answer test for computing keyed hash message authentication code using predetermined data embedded into the PSD firmware.

## 6.7  CSP Storage

The definition and the storage of CSP is described in Section 7 - Definition of Critical Security Parameters (CSP).

All CSPs (except Master Secret Key, Random Number Seed and Random Number Key stored in the RTC RAM) are stored encrypted by the Master Secret Key.

- The PSD shall detect data corruption of the value held for any particular CSP by the incorporation of 16 bit error detection data.

- Plain text CSPs (Master Secret Key, Random Number Seed and Random Number Key) shall be zeroized
    - o   by issuing "Zeroize CSPs" service or
    - o   by breaching the special flex circuit envelope that surrounds the module or
    - o   by bringing the module temperature above 84°C.

- Any CSP access failure shall cause the PSD to pass in faulted state.

## 6.8  Tamper Response

- In addition to the required FIPS 140-2 physical security protection mechanisms, the module incorporates a tamper detection and response mechanism in the form of a tamper detection envelope that protects the module from direct penetration attacks such as drilling or cutting the module.

- The RTC-RAM shall be erased to zero (the RTC-RAM contains all plaintext CSPs: Master Secret Key, Random Number Seed and Random Number Key) if the PSD physical cryptographic boundary is breached. At the same time the PSD shall pass in faulted state.

- The RTC-RAM shall be erased to zero (the RTC-RAM contains all plaintext CSPs: Master Secret Key, Random Number Seed, and Random Number Key) if the temperature inside the PSD covers exceeds 84 degrees Centigrade. At the same time the PSD shall pass in faulted state.

- The CSPs shall not be output from of the module in plain text under any circumstances.

## 6.9   Operational Environment

- The PSD was designed to securely operate when voltage supplied to the module is between +5V and +17V and the environmental temperature is between -30°C and 84°C. The module employs a high temperature fuse for the EFP circuitry such that when the module temperature exceeds 84°C, the module will zeroize all plaintext CSPs.

## 6.10  Status Indication

- The following 'module not ready' module states shall be indicated:
  - CSPs zeroed
  - Private/Public key pairs invalid (module not initialised)
  - Tamper mechanism tampered
  - Power Up tests error
  - RNG error
  - High temperature detected error

Indication will be via a unique text message output by the module suitable for viewing on an alphanumeric display device. The absence of one of these messages indicates that the module is in a 'ready' state.

## 6.11  Operators/Customers

Operators/customers shall be instructed to check for any errors, indicated by the status output, or for tamper evidence. Detection of any such errors or tamper evidence shall be required to be reported to Neopost such that the return of the PSD to the factory environment for withdrawal can be arranged.

# 7   Definition of Critical Security Parameters (CSP)

This section describes each CSP maintained by the PSD module.

| CSP NAME | DESCRIPTION | STORAGE | ZEROIZATION | KEY LENGTH |
|---|---|---|---|---|
| **Master Secret Key** | Master key of the PSD; used with AES algorithm to encrypt CSPs before their storage and decrypt the stored CSPs before usage. | **RTC-RAM** | **Zeroize CSPs service, breach of flex circuit or module temperature over 84°C.** | **128 bits** |
| **Random Number Seed** | Current status of the seed value is used by the random number generator. | **RTC-RAM** | **Zeroize CSPs service, breach of flex circuit or module temperature over 84°C.** | **128 bits** |
| **Random Number Key** | Key value is used by the random number generator. | **RTC-RAM** | **Zeroize CSPs service, breach of flex circuit or module temperature over 84°C.** | **128 bits** |
| **Communication Private Key** | The PSD private RSA key is used to authenticate (sign) messages and data output from the PSD during TLS handshake protocol. The current value and the future value of the key is always available | **Flash(stored encrypted by the AES Master Key)** | **Zeroization of the AES Master Key** | **1536 bits** |
| **TLS Communication Secret Keyset** | The 4 session keys generated by TLS Handshake protocol to ensure confidentiality and authenticity of messages exchanged between the client and the server or base. | **RAM only** | **At power off or end of TLS session.** | **4 x 128 bits** |

| | | | | |
|---|---|---|---|---|
| **Secondary Stamp Key** | The PSD stamp secret key (HMAC-SHA-1) is used to authenticate the stamp.<br><br>The current value and the future value of the key is always available | **Flash**<br><br>**(stored encrypted by the Master Secret Key)** | **Zeroization of the AES Master Key** | **160 bits** |
| **Stamp Key** | The PSD stamp private key (ECDSA – P192) is used to authenticate the stamp.<br><br>The current value and the future value of the key is always available | **Flash**<br><br>**(stored encrypted by the Master Secret Key)** | **Zeroization of the AES Master Key** | **192 bits** |

# 8   Definition of Public Parameters

This section describes public parameters maintained by the PSD.

| PUBLIC KEYS AND PARAMETERS | DESCRIPTION | STORAGE | KEY LENGTH |
|---|---|---|---|
| **Root Public Key**<br><br>**Neopost Root Certificate** | Signed X509 Certificate of the current Root Public key used for the verification of authenticated messages input from the Neopost server. | **EEPROM** | **2048 bits** |
| **Previous Root Public Key**<br><br>**Neopost Previous Root Certificate** | Signed X509 Certificate of the previous Root Public key used for the verification of authenticated messages input from the Neopost server. | **EEPROM** | **2048 bits** |
| **Region Public Key**<br><br>**Neopost Region Certificate** | Signed X509 Certificate of the Region Public Key used for the verification of authenticated messages input from the Neopost server. | **EEPROM** | **2048 bits** |
| **Communication Public Key**<br><br>**Communication Certificate** | The PSD public RSA key used to authenticate messages and data output from the PSD (TLS protocol)<br><br>The current value and the future value of the key is always available | **Flash**<br><br>**(stored encrypted by the master key)** | **1536 bits** |
| **Device Public Key**<br><br>**Neopost Device Certificate** | Signed X509 Certificate of the Device Public key used for the verification of authenticated messages input from the Neopost server. | **EEPROM** | **1536 bits** |
| **Neopost Diffie Hellman Parameters** | Diffie Hellman parameters used during TLS handshake to agree upon a TLS premaster secret. | **Flash** | **1280 bits** |
| **Stamp Transport Public Key** | The RSA public transport key is used to export the Secondary Stamp Key. The PSD encrypts the Stamp Key with the RSA Stamp Transport Public Key before sending it to the Neopost server. | **RAM only** | **1536 bits** |
| **Stamp Public Key** | The ECDSA – P192 public key is used to authenticate the stamp.<br><br>The current value and the future value of the key is always available | **Flash** | **392 bits** |

# 9   Definition of CSP Modes of Access

The section describes how CSPs are accessed by the services that can be activated by an operator. The modes of access are defined as follows:

- u    The data item will be read for internal use.
- e    The data item will be read and exported.
- w    The data item will be updated directly from an imported value.
- m    The data item will be modified to a value created by an internal process.
- z    The data item will be zeroized.
- s    The data item will be initialised to a starting value created by an internal process.
- i    The data item will be initialised to a benign value (typically zeroized).

<u>Note</u>:
The notation -/u means that in the first case, for an unauthenticated user, the parameter is not used and in the second case, for an authenticated user, the parameter will be read for internal use.
The notation m+e means that the key or CSP will be firstly modified to a value created by an internal process and then exported out of the module.
The notation m+z means that the parameter will be firstly modified to a value created by an internal process and then will be zeroized.

The first table summarises the relationship between generic CSPs maintained by the PSD and the services that access them.

The second table summarises the relationship between country specific CSPs maintained by the PSD and the services that access them.

| GENERIC CSP ▸<br><br>SERVICE ▾ | MASTER SECRET KEY | RANDOM NUMBER SEED | RANDOM NUMBER KEY | COMMUNICATION PRIVATE KEY | TLS COMMUNICATION SECRET KEYSET | STAMP KEY | SECONDARY STAMP KEY |
|---|---|---|---|---|---|---|---|
| **Read Status Data** | -/u | -/m | -/u | -/u | -/u | - | - |
| **Initiate Postal Core Connection** | u | m | u | u | m | - | - |
| **End Postal Core Connection** | u | m | u | u | m+z | - | - |
| **Initiate Rekey Connection** | u | m | u | u | u | - | - |

| End Rekey Connection | u | m | u | u | u | - | - |
|---|---|---|---|---|---|---|---|
| Postal Indicium | u | m | u | u | u | u | u |
| Other Base Services | u | m | u | u | u | - | - |
| Postal Core Services | u | m | u | u | u | - | - |
| Generate Stamp Key | u | m | u | u | u | m+e | m+e |
| Generate PKI Key | u | m | u | m | u | - | - |
| Get PKI Certificates | u | m | u | m | u | - | - |
| Set PKI Certificates | u | m | u | m | u | - | - |
| Zeroize CSPs / Tamper detection triggered / EFP High temp | z | z | z | - | - | - | - |

# 10 Definition of Public Parameters Modes of Access

The section describes how CSPs are accessed by the services that can be activated by an operator. The modes of access are defined as follows: -

u    The key or CSP item will be read for internal use.

e    The key or CSP item will be read and exported.

w    The key or CSP item will be updated directly from an imported value.

m    The key or CSP item will be modified to a value created by an internal process.

z    The key or CSP item will be zeroed.

s    The key or CSP item will be initialised to a starting value created by an internal process.

i    The key or CSP item will be initialised to a benign value (typically zeroed).

Note:
The notation -/u means that in the first case, for an unauthenticated user, the parameter is not used and in the second case, for an authenticated user, the parameter will be read for internal use.
The notation m+e means that the key or CSP will be firstly modified to a value created by an internal process and then exported out of the module.

The following table summarises the service relationships for generic public key parameters maintained by the PSD.

| Generic Public Parameter Name ▸<br><br>SERVICE ▾ | Root Public Key | Previous Root Public Key | Region Public Key | Communication Public Key | Device Public Key | Diffie Hellman Parameters | Stamp Transport Public Key | Stamp Public Key |
|---|---|---|---|---|---|---|---|---|
| Zeroize CSPs | - | - | - | - | - | - | - | - |
| Read Status Data | -/u | -/u | -/u | u | -/u | -/u | - | - |
| Initiate/End Postal Core Connection | u | u | u | u | u | u | - | - |
| Initiate/End Rekey Connection | u | u | u | u | u | u | - | - |
| Postal Indicium | u | u | u | u | u | u | - | - |
| Other Base Services | u | u | u | u | u | u | - | - |
| Postal Core Services | u | - | u | u | u | u | - | - |
| Generate Stamp Key | u | - | u | u | u | u | u | m+e |
| Generate PKI Key | - | - | - | m | - | - | - | - |
| Get PKI Certificates | u | u | u | u | u | u | - | - |
| Set PKI Certificates | w | w | w | u | w | u | - | - |

# 11 Appendix A: Relations transactions/states

The following table summarises the legality of transactions according to the prevailing lifecycle state of a PSD:

| PSD STATE ▸<br><br>TRANSACTION ▾ | INSTALLED | LOCKED | WITHDRAWN | FAULTED |
|---|---|---|---|---|
| Resetting | ✔ | ✔ | | |
| Update Registration | ✔ | ✔ | | |
| Withdrawal | ✔ | ✔ | ✔ | |
| Rekey | ✔ | ✔ | ✔ | |

A transaction is not permitted for a particular state unless indicated:

✔ = permitted

# 12 Appendix B: Relation services/roles

| SERVICE ▾ / ROLE ▸ | PRINTING BASE | POSTAL USER | POSTAL CRYPTO OFFICER | FIELD CRYPTO OFFICER | R&D SIGNER | UNAUTHENTICATED USER | REGION | ROOT |
|---|---|---|---|---|---|---|---|---|
| Zeroize CSPs | | | | | | ✔ | | |
| Read Status Data | ✔ | ✔ | | ✔ | | ✔ | | |
| Initiate/End Postal Core Connection | ✔ | | | | | | | |
| Initiate/End Rekey Connection | ✔ | | | | | | | |
| Postal Indicium | ✔ | | | | | | | |
| Other Base Services | ✔ | | | | | | | |
| Verify Files | | | | | ✔ | | | |
| Verify Device Certificates | | | | | | | ✔ | |
| Verify Region Certificates | | | | | | | | ✔ |
| Verify Root Certificates | | | | | | | | ✔ |
| Postal Core Services | | ✔ | | | | | | |
| Generate Stamp Key | | | ✔ | | | | | |
| Generate PKI Key | | | | ✔ | | | | |
| Get/Set PKI Certificates | | | | ✔ | | | | |

Service is not accessible to a particular entity unless specifically indicated:

✔ = can be accessed