Security Policy
PostagePlusÔ Client Communications Module

Prepared for:

Saranac Software Inc.
241 W. Fayette Street
Syracuse, NY 13202

&

Neopost Inc.
30955 Huntwood Avenue
Hayward, CA 94544-7084

August 28, 1998

Security Policy was consolidated by CygnaCom Solutions, Inc.

1 Introduction

The PostagePlus™ client is designed to perform all functions needed by a small business/home office relating to the acquisition and printing of legal US postage. The functioning of this product will be in direct accordance with the United State Postal Service Information Based Indicia Program (IBIP), as defined by a series of specifications released by the USPS and associated bodies.

The PostagePlus™ client will act as the Host System, and will display values of the Postal Security Device (PSD) to the user. The actual values of the user's PSD will be stored on the PostagePlus™ server. Postage will be acquired and displayed on the customer's local PC, and all printing of indicium will use standard Microsoft Windows™ compatible printers. The functions performed by the local PC, acting as Host, will be as follows:

- Secure purchase and local storage of postage
- Integration with existing word-processing applications
- Processing of mailing lists
- Implementation of ZIP+4 address cleansing
- Calculation of postage by mail-piece according to USPS rate tables
- Creation of authorized indicium (both FIM and bar-code)
- Formatting of mail-piece for proper printing
- Printing mail-piece on standard Windows™ printers

In addition, this product will be designed to allow other software manufacturers to access both the printing and transaction management pieces of this application through a controlled set of Application Program Interfaces (API).

This document contains the FIPS 140-1 compliant security policy for the **PostagePlus**™ Client Communication Module version 1.0. The document specifies the security requirements under which **PostagePlus**™ is designed, and addresses the requirements outlined in the FIPS 140-1 requirements.

The security requirements cover eleven areas related to the secure design and implementation of a Cryptographic Module. These areas include the following:

1. Cryptographic Module Design and Documentation
2. Module Interfaces
3. Roles and Services
4. Finite State Machine Model
5. Physical Security
6. Software Security
7. Operating System Security
8. Cryptographic Key Management
9. Cryptographic Algorithms
10. Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)
11. Self Tests

2 Applicable FIPS 140-1 Levels

Category Levels tested for the PostagePlusÔ Client Communication Module

Overall Level met by PostagePlus™ = 1

CATEGORY	LEVEL
Cryptographic Modules	1
Module Interfaces	1
Roles & Services	1
Finite State Machine Model	1
Physical Security	1
Software Security	1
Operating System Security	1
Key Management	1
Cryptographic Algorithms	1
EMI/EMC	3
Self-Tests	1

2.1 Cryptographic Module

The cryptographic boundary is defined to be the commsec.dll and three configuration files that are accessed by the comsec.dll:

- Useprofile.bin. This file contains a copy of all account information stored on the PostagePlus server. This data is stored for informational purposes only. The actual production data is stored on the server.
- User.kyr: This file contains the user's keyfile. The information contained in this file uniquely identifies a customer and is used to authenticate all requests sent to the server. This key file is DES encrypted and a password is required each time the file is accessed.
- Execrc: This file contains the DES MAC result that is expected during a software/firmware test.

The commsec.dll handles all requests for cryptographic functions such as hashing and encryption. The Commsec module performs communications encryption and manages session keys, which are generated at the beginning of each client/server session. It is also responsible for maintaining/verifying the integrity of all client/server messages. The commsec.dll also maintains message integrity by including a hash of all transactions within the message that is sent to the server.

The commsec.dll is also responsible for verifying the hash result of all messages received from the server. This is accomplished by decrypting the transaction and generating a hash on the data and comparing the result with the decrypted hash.

2.2 Module Interfaces

The module has all the four required logical interfaces: data input interface, data output

interface, control input interface, and status output interface.

Since this cryptographic module is a software module, all four logical interfaces are provided through the PostagePlus™ Client Communication module API. The input parameters for the API functions provide the data input to the module. Data output is logically provided through the output parameters of the API functions. Control input is provided by the actual API function calls themselves. Finally, status output is provided by the return codes from each function.

Also, data input and output are received/sent through the network card. The network card allows the client-server communication possible.

2.3 Roles and Services

The module has the following two roles: Crypto-Officer role and User role.

All users must authenticate to the module by providing an eight byte PIN to the Init(…) function.

The Crypto-Officer and User roles share all services provided by the PostagePlus™ Client Communication Module. Users have the capability to send and receive encrypted transactions, change user passwords, and initiate self-tests.

The following is the sequence of functions that must be called:

Order	Function Name	Description
1	VerifyDLLIntegrity(..)	Performs all power-up self-tests as described in section 2.11
2a	Init(…)	Performs user authentication
2b	ExecuteTransaction(…)	Performs a series of client-server communication transactions, this function can be called multiple times.
1a, 3	ChangePassword(…)	Allows user to change password, this can be called anytime after VerifyDLLIntegrity(…)

The cryptographic module will enforce the order as described above.

2.4 Finite State Machine

The PostagePlus™ module was developed with reference to the Finite State Machine. The finite state machine is proprietary to Saranac Software and Neopost Inc.

2.5 Physical Security

The PostagePlus™ Client Communication Module is a software only module. However, the module was tested using a general PC. The general PC ran under Windows 95 with a network card and met all the FIPS 140-1 Level 1 requirements for physical security.

2.6 Software Security

The PostagePlus™ module was written in the C++ language.

2.7 Operating System Security

The cryptographic module has been tested and is supported on the Windows 95 and 98 operating systems.

All cryptographic software shall be installed only as executable code on the PostagePlus host system.

Upon powerup and after performing the Memory, and Cryptographic self-tests, PostagePlus Client software shall check the integrity of the client executables. If the integrity has been compromised, the PostagePlus will enter the Error State.

Execution of cryptographic functions first requires the user to enter a password. If this password is found to be valid, cryptographic processing is continued. If it is not validated the Cryptographic Module enters the Error State.

2.8 Cryptographic Key Management

There are three sets of keys that the PostagePlus™ module uses: User public/private keys (RSA), server public key (RSA), and two session keys.

Key Generation
The module only generates DES session keys through both the user and crypto-officer roles. The key generation method used is compliant with a FIPS-approved key generation method (ANSI X9.17).

Key Distribution
DES session keys are distributed via a commercial public key-based distribution technique. All session keys distributed are encrypted (RSA) using the servers' public key.

Key Entry and Output
The module supports electronically entering and outputting DES session keys. These keys are entered and outputted in encrypted form.

Key Storage
All public and private keys in the user key file are stored in encrypted form. Once the user has authenticated to the module, the keys are decrypted into computer memory. These keys are not accessible outside the cryptographic module.

Key Destruction
All public and private keys loaded into computer memory are destroyed when the module enters an error state or when the module is powered off.

The session keys used by the module are zeroized after the ExecuteTransaction(…)

function has been completed.

2.9 Cryptographic Algorithms

2.9.1 SHA-1

The cryptographic Module shall employ SHA-1 for all hashing functions. Hashing is used to verify data integrity.

2.9.2 DES

The cryptographic Module shall employ DES to encrypt/decrypt all communications between the client and the server. The module also supports the use of triple DES to encrypt/decrypt all communications between the client and the server

2.10 EMI/EMC

The PostagePlus™ Client Communication module is a software only module. However, the product was evaluated on a standard PC that has been certified to meet the requirements specified in FCC Part 15, Subpart J, Class B.

2.11 Self Tests

PostagePlus™ performs self-tests to verify the proper operation of the Cryptographic Module. The following self-tests are performed each time PostagePlus™ powers up:

2.11.1 SHA-1

PostagePlus™ performs a "known-answer" test in which the Cryptographic Module generates a hash on an internally stored data field and compares the generated hash to a reference hash stored in memory. If the two hashes are identical, the test passes. If the hashes are not identical, the test fails and PostagePlus™ enters the Error State.

2.11.2 DES

PostagePlus™ performs a "known-answer" test in which the Cryptographic Module encrypts an internally stored data field and compares the encrypted value to a reference encrypted value stored in memory. If the two values are identical, the test passes. If the values are not identical, the test fails and PostagePlus™ enters the Error State. For decryption the known encrypted value will be decrypted and compared to the known decrypted value stored in memory. If the two values are identical, the test passes. If the values are not identical, the test fails and PostagePlus™ shall enter the Error State.

2.11.3 Continuos Random Number Generator

PostagePlus™ performs a comparative test to ensure that a newly generated random numbers do not equate in value to the immediate predecessors.

2.11.4 Code Integrity

The PostagePlus™ uses a FIPS-approved authentication technique (DES DAC) to ensure that code integrity is not violated.

2.11.5 Basic Security Check

Upon powerup and after performing the Memory, and Cryptographic self-tests, PostagePlus shall perform the following series of tests to determine if the content of non-volatile memory is valid.

PostagePlus shall check the CRC or checksum on the client executables. If the CRC or checksum does not verify, PostagePlus shall enter the Error State.