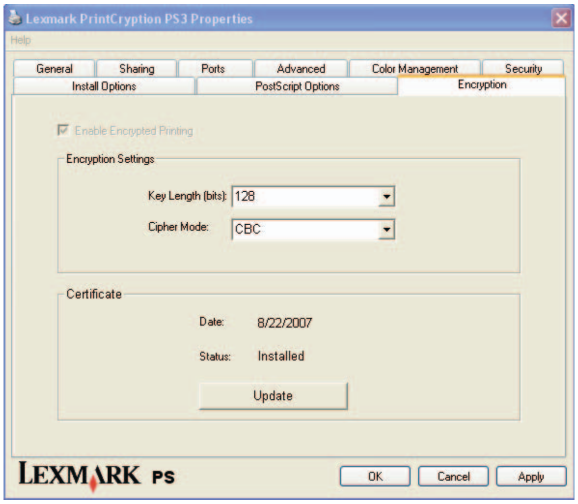
Lexmark International, Inc.

Lexmark Encryption Plug-In

(Software Version: 1.1)



☑ Enable Encrypted Printing

Encryption Settings

Key Length (bits): 128

Cipher Mode: CBC

Certificate

Date: 8/22/2007

Status: Installed

FIPS 140-2
Non-Proprietary Security Policy

Level 1 Validation

Document Version 0.3

Prepared for:

Lexmark International, Inc.
740 West New Circle Road
Lexington, KY 40550
Phone: 859-232-2000
Fax: 859-232-3120
http://www.lexmark.com

Prepared by:

Corsec Security, Inc.
10340 Democracy Lane, Suite 201
Fairfax, VA 22030
Phone: 703-267-6050
Fax: 703-267-6810
http://www.corsec.com

Revision History

Version	Modification Date	Modified By	Description of Changes
0.1	2007-11-26	Xiaoyu Ruan	Initial draft.
0.2	2008-04-17	Xiaoyu Ruan	Algorithm certificate numbers.
0.3	2008-07-21	Xiaoyu Ruan	Addressed CMVP comments.

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Purpose

This document is a non-proprietary Cryptographic Module Security Policy for the Lexmark Encryption Plug-In from Lexmark International, Inc. This Security Policy describes how the Lexmark Encryption Plug-In meets the security requirements of FIPS 140-2 and how to run the module in a secure FIPS 140-2 mode. This policy was prepared as part of the Level 1 FIPS 140-2 validation of the module.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2 – *Security Requirements for Cryptographic Modules*) details the U.S. and Canadian Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the National Institute of Standards and Technology (NIST) Cryptographic Module Validation Program (CMVP) website at: http://csrc.nist.gov/groups/STM/index.html.

In this document, the Lexmark Encryption Plug-In is referred to as "the module".

1.2 References

This document deals only with the operations and capabilities of the module in the technical terms of a FIPS 140-2 cryptographic module security policy. More information is available on the module from the following sources:

- The Lexmark website (http://www.lexmark.com) contains information on the full line of products from Lexmark.
- The CMVP website (http://csrc.nist.gov/groups/STM/index.html) contains contact information for answers to technical or sales-related questions for the module.

1.3 Document Organization

The Security Policy document is one document in a FIPS 140-2 submission package. In addition to this document, the submission package contains:

- Vendor evidence
- Finite state machine
- Other supporting documentation as additional references

This Security Policy and the other validation submission documentation was produced by Corsec Security, Inc. under contract to Lexmark. With the exception of this Non-Proprietary Security Policy, the FIPS 140-2 validation documentation is confidential and proprietary to Lexmark and is releasable only under appropriate non-disclosure agreements. For access to these documents, please contact Lexmark.

2 Lexmark Encryption Plug-In

2.1 Overview

Since the inception in 1991 as a spin-off of IBM, Lexmark has become a leading developer, manufacturer and supplier of printing and imaging solutions for offices and homes. Lexmark's products include laser printers, inkjet printers, multifunction devices, and associated supplies, services, and solutions.

The Lexmark Encryption Plug-In is a software cryptographic module implemented as a dynamic link library (DLL), *LMAA654A.dll*. The Lexmark Encryption Plug-In is a user space shared library. It does not modify or become part of the Operating System (OS) kernel. The module is installed as part of the Lexmark PrintCryption PS3 print driver on a General-Purpose Computer (GPC) running Windows 2000, Windows 2000 Server, Windows Server 2003, or Windows XP. The FIPS 140-2 testing was performed on Windows XP.

The Lexmark Encryption Plug-In supports FIPS-Approved algorithms including Advanced Encryption Standard (AES) encryption and ANSI[1] X9.31 Appendix A.2.4 Random Number Generator (RNG). In addition, the module implements keyed-Hash Message Authentication Code-SHA-1 (HMAC-SHA-1) for software integrity self-test and 2048-bit Rivest, Shamir, and Adleman (RSA) key wrapping. The module always operates in a FIPS-Approved mode of operation.

The module is validated at FIPS 140-2 section levels shown in Table 1 – Security Levels per FIPS 140-2 Section. Note that in Table 2, EMI and EMC stand for Electromagnetic Interference and Electromagnetic Compatibility, respectively, and N/A indicates "Not Applicable".

Table 1 – Security Levels per FIPS 140-2 Sections

Section	Section Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC	1
9	Self-Tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

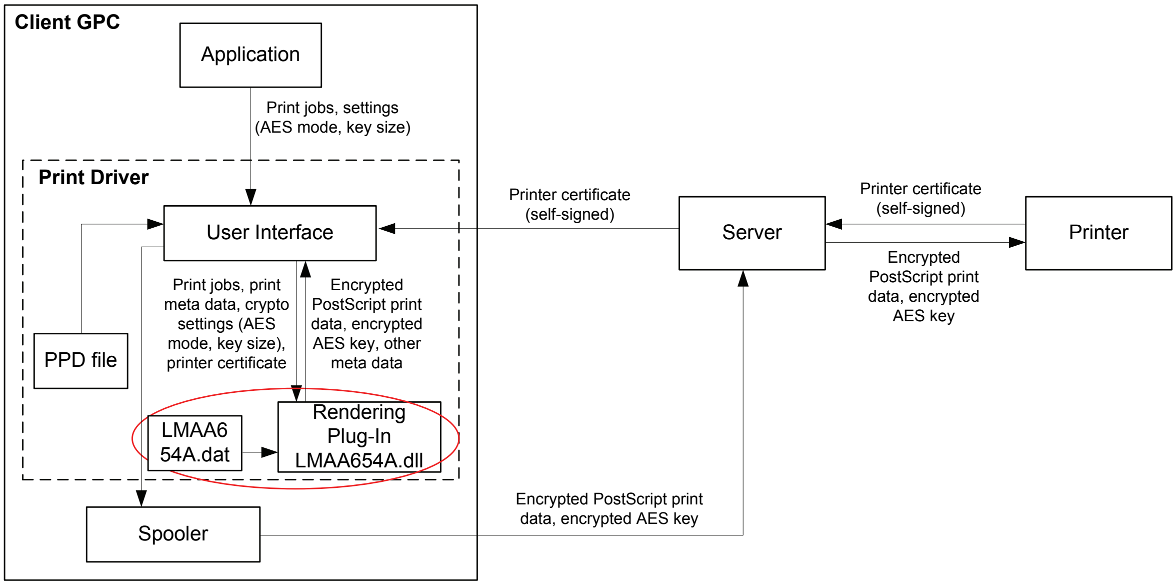
2.2 Module Interfaces

The logical cryptographic boundary of the module contains *LMAA654A.dll* and a supporting file, *LMAA654A.dat*, which stores the HMAC-SHA-1 signature of *LMAA654A.dll*. The Lexmark Encryption Plug-In is a software cryptographic module implemented as a single dynamic link library (DLL) named *LMAA654A.dll*. The module runs

[1] American National Standards Institute

on Windows 2000, Windows 2000 Server, Windows Server 2003, or Windows XP operating systems. The FIPS 140-2 testing was performed on Windows XP.

The Lexmark PostScript print driver encryption mechanism is depicted in Figure 1 – Lexmark PostScript Print Driver Encryption. The logical cryptographic boundary of the module contains *LMAA654A.dll* and a supporting file *LMAA654A.dat* that stores the HMAC-SHA-1 signature of *LMAA654A.dll*. The red ellipse in Figure 1 – Lexmark PostScript Print Driver Encryption shows the logical cryptographic boundary of the module.


Client GPC
Application
Print jobs, settings (AES mode, key size)
Print Driver
User Interface
PPD file
Print jobs, print meta data, crypto settings (AES mode, key size), printer certificate
Encrypted PostScript print data, encrypted AES key, other meta data
LMAA654A.dat
Rendering Plug-In LMAA654A.dll
Spooler
Printer certificate (self-signed)
Server
Printer certificate (self-signed)
Printer
Encrypted PostScript print data, encrypted AES key
Encrypted PostScript print data, encrypted AES key

- User Interface includes: UI plug-in DLL, Unidrvui.dll, and PScript5.dll
- PPD (PostScript Printer Description) file describes the entire set of features and capabilities available for the printer.
- Printer certificate contains printer's 2048-bit RSA public key.

⬭ FIPS cryptographic boundary

Figure 1 – Lexmark PostScript Print Driver Encryption

The print server sends the printer's 2048-bit RSA public key to the print driver running on a client GPC as part of the printer's self-signed X.509 digital certificate. When a print job is initiated, the print driver:

- generates an AES key (and an IV if CBC mode is specified by the User),
- encrypts the AES key (and IV if applicable) with printer's RSA public key,
- encrypts the print job with the AES key, and
- sends the encrypted print job and encrypted AES key (and IV if applicable) to the printer via the print server.

The plaintext and encrypted print job and all keys are zeroized as soon as the print job is accomplished. No plaintext data is left on the client GPC or the print server.

LMAA654A.dll, as a rendering plug-in, is responsible for generating AES key and IV, encrypting AES key and IV with printer's RSA public key, and encrypting print jobs with the AES key. In other words, all encryption work is carried out by *LMAA654A.dll*.

The module's interactions with surrounding components, including the Central Processing Unit (CPU), harddisk, memory, client application, and the OS, are demonstrated in Figure 2.

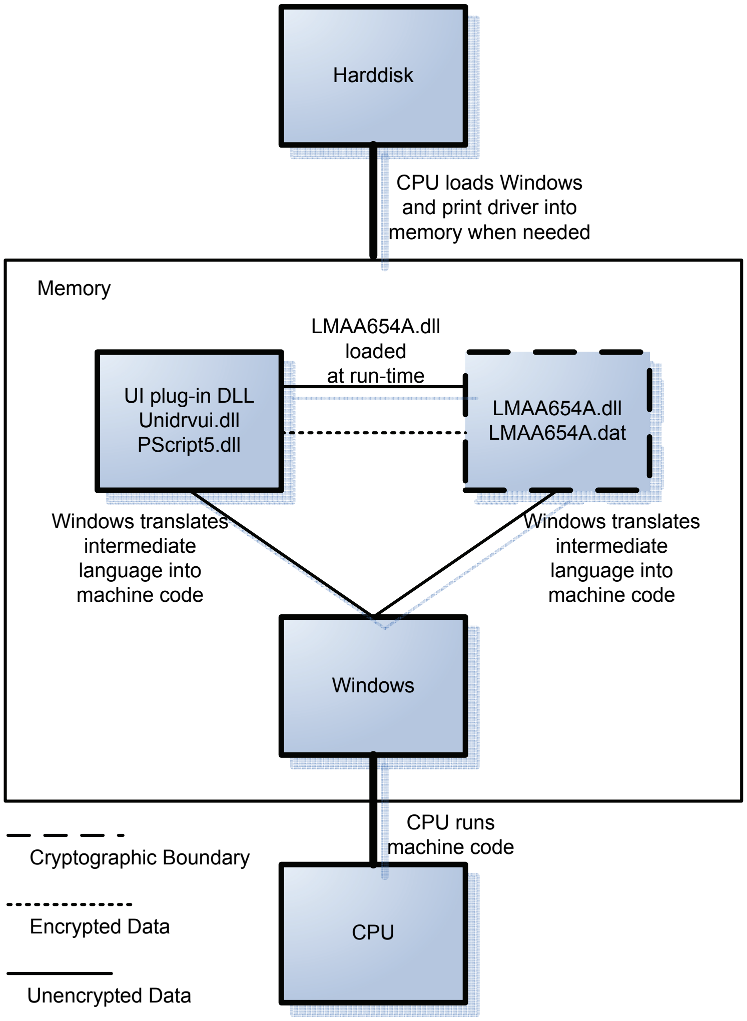

Harddisk
CPU loads Windows and print driver into memory when needed
Memory
LMAA654A.dll loaded at run-time
UI plug-in DLL Unidrvui.dll PScript5.dll
LMAA654A.dll LMAA654A.dat
Windows translates intermediate language into machine code
Windows translates intermediate language into machine code
Windows
CPU runs machine code
CPU
Cryptographic Boundary
Encrypted Data
Unencrypted Data

Figure 2 – Logical Cryptographic Boundary and Interactions with Surrounding Components

The module runs on a standard GPC running Windows 2000, Windows 2000 Server, Windows Server 2003, or Windows XP. The FIPS 140-2 testing was performed on Windows XP. In addition to the binaries, the physical device consists of the integrated circuits of the motherboard, the CPU, Random Access Memory (RAM), Read-Only Memory (ROM), computer case, keyboard, mouse, video interfaces, expansion cards, and other hardware components included in the GPC such as hard disk, floppy disk, Compact Disc ROM (CD-ROM) drive, power supply, and fans. The physical cryptographic boundary of the module is the hard opaque metal and plastic enclosure of the GPC. The block diagram for a standard GPC is shown in Figure 3. Note that in this figure, I/O means Input/Output, BIOS stands for Basic Input/Output System, PCI stands for Peripheral Component Interconnect, ISA stands for Instruction Set Architecture, and IDE represents Integrated Drive Electronics.

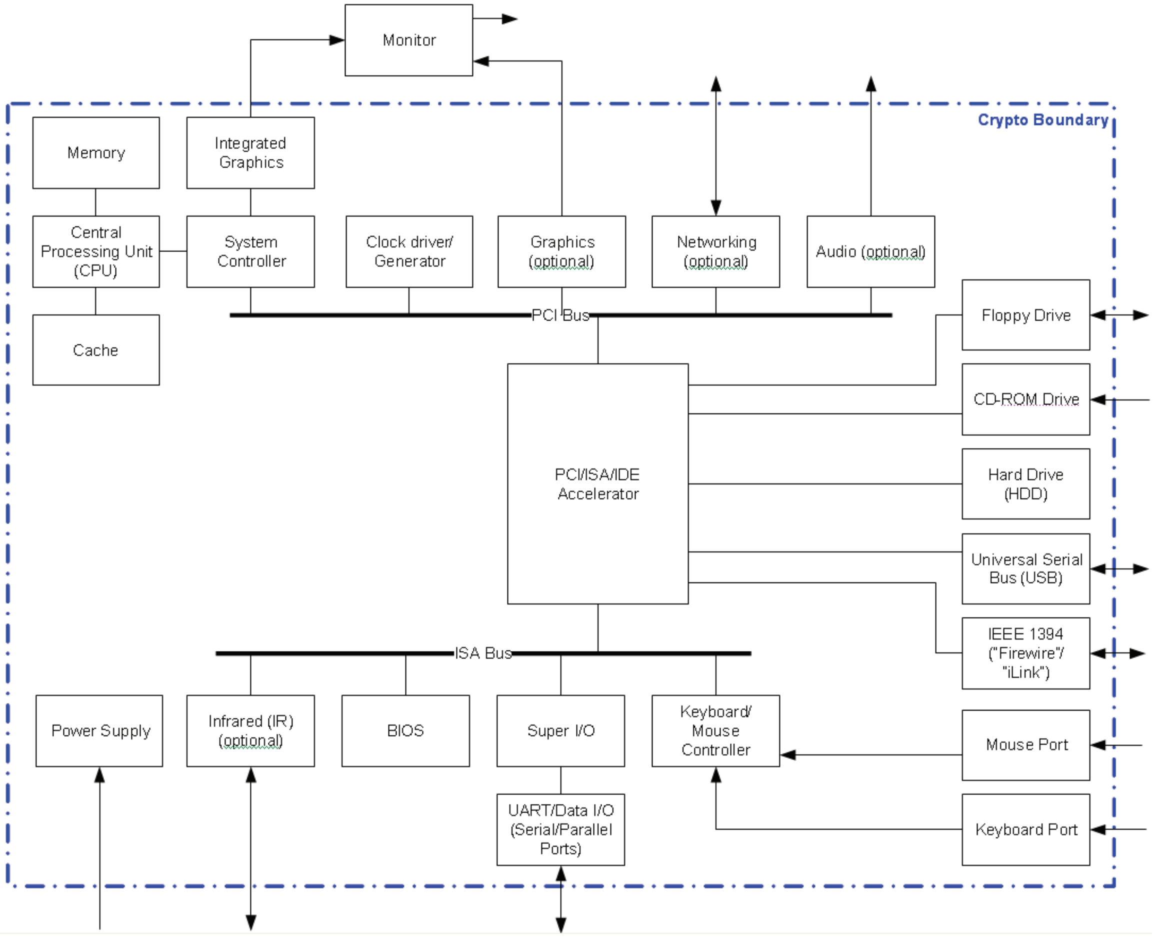


Figure 3 – Standard GPC Physical Block Diagram

All of the physical interfaces are mapped to logical interfaces (as defined by FIPS 140-2) as described in Table 2 – FIPS 140-2 Logical Interfaces.

Table 2 – FIPS 140-2 Logical Interfaces

Logical Interface	Physical Interface Mapping	Module Mapping
Data Input	Keyboard, mouse, CD-ROM, floppy disk, and serial/USB/parallel/network ports	Arguments for API calls that contain data to be used or processed by the module
Data Output	Hard disk, floppy disk, monitor, and serial/USB/parallel/network ports	Arguments for API calls that contain module response data to be used or processed by the caller
Control Input	Keyboard, mouse, and serial/USB/parallel/network port	API function calls
Status Output	Monitor and serial/USB/parallel/network ports	Arguments for API calls, function return values, error /status messages

2.3 Roles and Services

The module supports two authorized roles: a Crypto-Officer role and a User role. The Crypto-Officer role is defined to be a human operator that installs and uninstalls the module. The User role is defined to be the User Interface (UI) plug-in DLL, *Unidrvui.dll*, and *PScript5.dll* that invoke functions exported by the module.

The module does not implement authentication. Role selection is implicit. The operator of the module assumes either of the roles based on the operations performed without any authentication. When the operator is using services listed in Table 3 – Crypto-Officer Services, he implicitly assumes the Crypto-Officer role. When the operator is using services listed in Table 4 – User Services, he implicitly assumes the User role.

2.3.1 Crypto-Officer Services

Table 3 – Crypto-Officer Services describes the functions that the Crypto-Officer's can perform.

Table 3 – Crypto-Officer Services

Service	Description	Input	Output
Install module	Installs and configures the module.	Command	Status
Uninstall module	Removes the module from the OS.	Command	Status

2.3.2 User Services

The module only exports two functions, *DllGetClassObject()* and *DllCanUnloadNow()*. They are described in the first two rows of Table 4 – User Services. Other functions listed in the table are object members returned from call to the *DllGetClassObject()* function. "CSP" refers to Critical Security Parameters (see Table 5 – List of CSPs for details).

Table 4 – User Services

Service	Description	Input	Output	CSPs
DllGetClassObject[2]	Retrieves the class object from the module object handler or object application.	*rclsid, riid*	*ppv,* status	None
DllCanUnloadNow[3]	Determines whether the module is in use. If not, the caller can unload the module from memory.	None	Status	AES key, RSA public key, and ANSI X9.31 RNG seed – delete
QueryInterface[4]	Returns a pointer to a specified interface on an object to which a client currently holds an interface pointer.	*iid*	*ppvObject,* status	None
AddRef[5]	Increments the reference count for an interface on an object. It should be called for every new copy of a pointer to an interface on a given object.	None	Status	None

[2] See http://msdn2.microsoft.com/en-us/library/ms680760.aspx for details.
[3] See http://msdn2.microsoft.com/en-us/library/ms690368.aspx for details.
[4] See http://msdn2.microsoft.com/en-us/library/ms682521.aspx for details.
[5] See http://msdn2.microsoft.com/en-us/library/ms691379.aspx for details.

Service	Description	Input	Output	CSPs
Release[6]	Decrements the reference count for the calling interface on an object. If the reference count on the object falls to 0, the object is freed from memory.	None	Status	None
EnableDriver[7]	Allows the module to hook out some graphics Device Driver Interface (DDI) functions.	*driverVersion, cbSize, pded*	Status	None
EnablePDEV[8]	Allows the module to create its own *PDEV* structure.	*pdevobj, pPrinterName, cPatterns, phsurfPatterns, cjGdiInfo, pGdiInfo, cjDevInfo, pDevInfo, pded,*	*pDevOem,* status	None
ResetPDEV[9]	Allows the module Pscript5 to reset its *PDEV* structure.	*pdevobjOld, pdevobjNew*	Status	None
WritePrinter[10]	Enables the module to capture all output data generated by a Postscript driver.	*pdevobj, pBuf, cbBuffer*	*pcbWritten,* status	RSA public key – read
DisablePDEV[11]	Allows the module to delete the private *PDEV* structure that was allocated by its *EnablePDEV* method.	*pdevobj*	Status	None
DisableDriver[12]	Allows the module to free resources that were allocated by the plug-in's *EnableDriver* method.	None	Status	None
Command[13]	Used by the module for the Microsoft PostScript printer driver, in order to insert PostScript commands into the print job's data stream.	*pdevobj, dwIndex,*	*pdwResult,* status	None
DevMode[14]	Performs operations on private *DEVMODEW* members.	*dwMode, pOemDMParam*	Status	None
GetInfo[15]	Returns identification information.	*dwMode, pBuffer, cbSize, pcbNeeded*	Status	None
PublishDriverInterface[16]	Allows the module to obtain the Pscript5 driver's *IPrintOemDriverPS* interface.	*pIUnknown*	Status	None

[6] See http://msdn2.microsoft.com/en-us/library/ms682317.aspx for details.
[7] See http://msdn2.microsoft.com/en-us/library/bb734263.aspx for details.
[8] See http://msdn2.microsoft.com/en-us/library/bb734254.aspx for details.
[9] See http://msdn2.microsoft.com/en-us/library/bb725869.aspx for details.
[10] See http://msdn2.microsoft.com/en-us/library/bb725882.aspx for details.
[11] See http://msdn2.microsoft.com/en-us/library/bb734292.aspx for details.
[12] See http://msdn2.microsoft.com/en-us/library/bb725894.aspx for details.
[13] See http://msdn2.microsoft.com/en-us/library/bb725885.aspx for details.
[14] See http://msdn2.microsoft.com/en-us/library/bb725864.aspx for details.
[15] See http://msdn2.microsoft.com/en-us/library/bb734256.aspx for details.
[16] See http://msdn2.microsoft.com/en-us/library/bb734323.aspx for details.

Service	Description	Input	Output	CSPs
GetPDEVAdjustment[17]	Enables the module to override specific *PDEV* settings.	*pdevobj, dwAdjustType, pBuf, cbBuffer, pbAdjustmentDone*	Status	None
DrvStartDoc[18]	Called by Graphics Device Interface (GDI) when it is ready to start sending a document to the driver for rendering.	*pso, pwszDocName, dwJobId*	Status	AES key, RSA public key, and ANSI X9.31 RNG seed – write, read

After *DllGetClassObject()* is called, the module (*LMAA654A.dll)* expects certain functions to be called in a specific order in order to process a print job,.

- (Required) *EnableDriver()*
- (Required) *EnablePDev()*
- (Optional) *ResetPDev()*
- (Required) *DrvStartDoc()*[19], call once per job
- (Optional) *ResetPDev()*
- (Required) *WritePrinter()*[20], call once or multiple times per job
- (Required) *DrvEndDoc()*[21], call once per job
- (Required) *DisablePDev()*
- (Required) *DisableDriver()*

ResetPDev() may be called before or after *StartDoc()* to change printing parameters from the defaults, or to change parameters within a job (e.g. mixing portrait and landscape pages within a job).

2.4 Physical Security

The physical security requirements do not apply to this module since it is a software module. Although the module consists entirely of software, the FIPS 140-2 tested platform is a standard GPC, which has been tested for and meets applicable Federal Communication Commission (FCC) EMI and EMC requirements for business use as defined in Subpart B of FCC Part 15.

2.5 Operational Environment

The module runs on Windows 2000, Windows 2000 Server, Windows Server 2003, or Windows XP. The Windows OS being used must be configured for single user mode per NIST CMVP guidance. The module was tested on Windows XP. Single user mode configuration instructions can be found in Section 3 of this document.

The module is installed in its compiled form of a DLL binary named *LMAA654A.dll*. At module startup, a software integrity test using HMAC-SHA-1 with a 160-bit MAC and a 512-bit key is performed to ensure that the module has not been modified.

[17] See http://msdn2.microsoft.com/en-us/library/bb725887.aspx for details.
[18] See http://msdn2.microsoft.com/en-us/library/ms793389.aspx for details.
[19] Generation of an AES key (and IV if applicable) and encryption of the AES key (and IV if applicable) happen here.
[20] Encryption of the print job (using AES) happens here.
[21] Destroy of the AES key (and IV if applicable) happens here.

2.6 Cryptographic Key Management

The module implements the following FIPS-approved algorithms.

- RNG: ANSI X9.31 Appendix A.2.4 (certificate #441)
- AES: encryption, 128, 192, and 256 bits, in Electronic Codebook (ECB) and CBC modes (certificate #767)
- SHA-1 (certificate #774)
- HMAC-SHA-1 (certificate #420)

The module also implements the following non-Approved security functions.

- Non-Approved RNG for seeding the ANSI X9.31 Appendix 2.4 RNG
- RSA PKCS[22]#1 key wrapping: 2048-bit key wrapping providing 112-bit of encryption strength

The module supports the following CSPs.

Table 5 – List of CSPs

Key	Key Type	Generation / Input	Output	Storage	Zeroization	Use
AES key	128-, 192-, or 256-bit symmetric key	Generated by ANSI RNG	Encrypted with 2048-bit RSA	Plaintext in volatile memory	When *DrvEndDoc()* is called	Encrypt print jobs
RSA public key	2048-bit RSA public key	Input in X.509 certificate in plaintext	Never output	Plaintext in non-volatile memory	Zeroized when new X.509 certificate is retrieved from print server	Encrypt AES keys and IVs
ANSI X9.31 RNG seed	RNG seed	Generated by non-Approved RNG	Never output	Plaintext in volatile memory only	Zeroized when new seed value is fed	Generate AES keys and IVs

The AES key can be 128, 192, or 256 bits. The module uses the AES key to encrypt plaintext print job. The AES encryption is configured in either ECB or CBC mode. Generated internally by the ANSI X9.31 Appendix A.2.4 RNG, the AES key is output to the print server in ciphertext form. The AES key is encrypted with the printer's 2048-bit RSA public key. The module stores the AES key in volatile memory during the encryption. The AES key is zeroized when *DrvEndDoc()* is called.

The printer's 2048-bit RSA public key enters the module in plaintext form as part of the printer's X.509 certificate when the operator of the print driver retrieves the printer's certificate by clicking the "Update" button. See Figure 6 – Encryption Configuration. The RSA public key never leaves the module. It is stored in non-volatile memory in plaintext. When the operator retrieves a new certificate from the print server, the current RSA public key will be zeroized and replaced by the new RSA public key.

The seed of the ANSI X9.31 Appendix A.2.4 RNG is a 64-bit random data generated by a non-Approved RNG. The seed is stored in plaintext in volatile memory and never leaves the module. The seed is updated at each call to the ANSI X9.31 Appendix A.2.4 RNG.

[22] Public Key Cryptography Standards

2.7 Self-Tests

The module, *LMAA654A.dll*, only exports two functions, *DllGetClassObject()* and *DllCanUnloadNow()*. The only functionality of *DllCanUnloadNow()* is to unload the module if it is in use. The following power-up self-tests are performed when *DllGetClassObject()* is called.

- Software integrity test using HMAC-SHA-1
- Known Answer Test (KAT) on AES encryption/decryption
- KAT on RSA encryption/decryption (note: this KAT is not required by FIPS 140-2 and it is treated as a critical function test)
- KAT on ANSI X9.31 Appendix A.2.4 RNG

The module implements the following conditional self-tests.

- Continuous RNG test on the ANSI X9.31 Appendix A.2.4 RNG
- Continuous RNG test on the non-Approved RNG

If a self-test is failed, an exception will be thrown on the failure. The module will enter an error state and be unloaded. If the logging feature is turned on, then the error message will be recorded in a log file *LMAA654A.log* located in the driver installation directory (by default *C:\WINDOWS\system32\spool\drivers\w32x86\3*). See Section 3 of this document for details.

2.8 Mitigation of Other Attacks

This section is not applicable. The module does not claim to mitigate any attacks beyond the FIPS 140-2 Level 1 requirements for this validation.

3 Secure Operation

The Lexmark Encryption Plug-In meets Level 1 requirements for FIPS 140-2. This section describes how to set up and use the module.

The module always works in the FIPS-Approved mode of operation. The Lexmark PrintCryption PS3 print driver does not support bypass services. All print jobs handled by the driver must be encrypted. See Figure 6 – Encryption Configuration. The "Enable Encrypted Printing" option cannot be unchecked.

3.1 Configuring the Windows OS

FIPS 140-2 mandates that a cryptographic module be limited to a single user at a time. Before installing the module, the Crypto-Officer must have a standard GPC running on Windows 2000, Windows 2000 Server, Windows Server 2003, or Windows XP and configure the Windows OS for single user mode.

For example, to configure Windows XP for single user mode, the Crypto-Officer must ensure that all remote guest accounts are disabled in order to ensure that only one human operator can log into the OS at a time. The services that need to be turned off for Windows XP are:

- Fast-user switching (irrelevant if GPC is a domain member)
- Terminal services
- Remote registry service
- Secondary logon service
- Telnet service
- Remote desktop and remote assistance service

For procedures of configuring other Windows OS to single user mode, please refer to the appropriate Windows user manual. Once Windows has been properly configured, the Crypto-Officer can use the "Administrator" account to install software, uninstall software, and administer the module.

Please note that the module was tested on Windows XP and hence its FIPS 140-2 validation only covers Windows XP. When running on Windows 2000, Windows 2000 Server, or Windows Server 2003, the module does not require any source code modifications (e.g., changes, additions, or deletions of code). According to the *Implementation Guidance for FIPS PUB FIPS 140-2 and the Cryptographic Module Validation Program*[23] (dated 5/22/2008), Section G.5,

> *"The CMVP allows vendor porting and re-compilation of a validated software and firmware cryptographic module from the OS(s) and/or GPC(s) specified on the validation certificate to an OS(s) and/or GPC(s) which were not included as part of the validation testing. The validation status is maintained on the new OS(s) and/or GPC without re-testing the cryptographic module on the new OS(s) and/or GPC(s)."*

The FIPS 140-2 validation for the module is maintained on Windows 2000, Windows 2000 Server, and Windows Server 2003.

3.2 Installing the Lexmark PrintCryption PS3 Print Driver

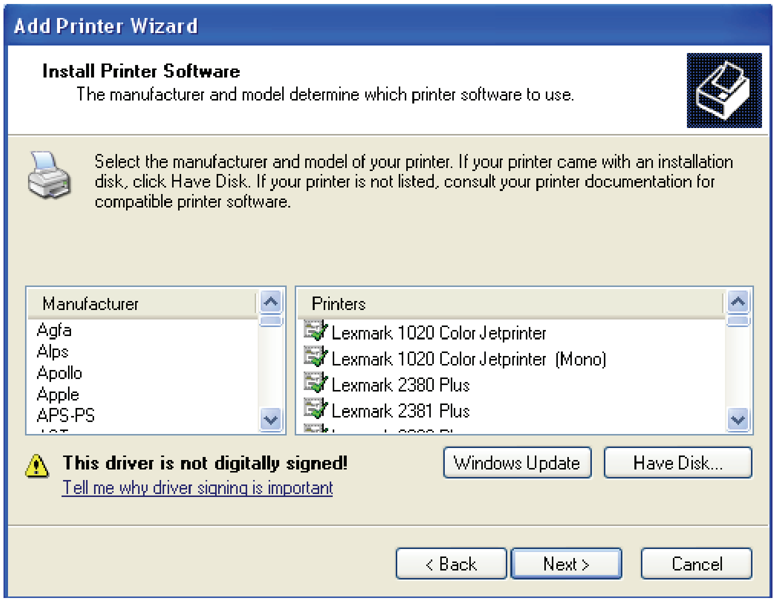
The module, *LMAA654A.dll*, is not a standalone software program. It will be provided to the operators by Lexmark along with the Lexmark PrintCryption PS3 print driver. The module is installed during installation of the print

[23] Available at http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf.

driver. The Crypto-Officer can follow Windows “Add Printer Wizard” to install the Lexmark PrintCryption PS3 print driver. The installation procedure is described below.

- Navigate to Windows “Printers and Faxes”.
- Click on “Add a printer”.
- Click on “Next” until the “Install Printer Software” dialog appears. See Figure 4 – “Install Printer Software” Dialog.
- Click on “Have Disk…”, and then browse to *LMA6540.inf*.
- Click on “Next” to complete the installation.

The driver installation directory is by default *C:\WINDOWS\system32\spool\drivers\w32x86\3*. The two files of the module, *LMAA654A.dll* and *LMAA654A.dat*, are located in this directory.



Add Printer Wizard

Install Printer Software

The manufacturer and model determine which printer software to use.

Select the manufacturer and model of your printer. If your printer came with an installation disk, click Have Disk. If your printer is not listed, consult your printer documentation for compatible printer software.

Manufacturer

Agfa
Alps
Apollo
Apple
APS-PS

Printers

Lexmark 1020 Color Jetprinter
Lexmark 1020 Color Jetprinter (Mono)
Lexmark 2380 Plus
Lexmark 2381 Plus

This driver is not digitally signed!
Tell me why driver signing is important

Figure 4 – “Install Printer Software” Dialog

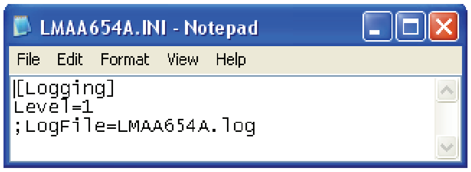
3.3 Configuring the Encryption Property

To configure the printer encryption property, perform the following steps.

- Navigate to Windows “Printers and Faxes”.
- Click on “Lexmark PrintCryption PS3”.
- Click on “Set printer properties”.
- Click on the “Encryption” tab. See Figure 6 – Encryption Configuration.
- Select the desired key length (128, 192, or 256 bits) and cipher mode (ECB or CBC).
- If update of the printer certificate is necessary, click on “Update”.
- Click on “OK” to complete the configuration.

3.4 Accessing the Log File

Create a text file named *LMAA654A.INI* using Windows Notepad with the contents showed in Figure 5 – *LMAA654A.INI*. Place *LMAA654A.INI* in the driver installation directory. A detailed log, *LMAA654A.log*, will be generated in the driver installation directory as the driver is working. To turn off the logging, simply remove *LMAA654A.INI* from the driver installation directory.



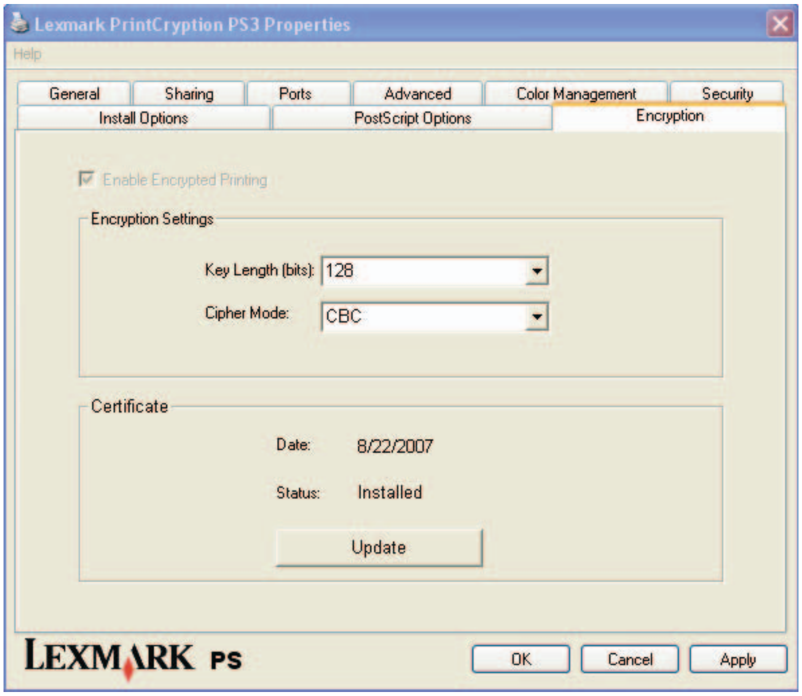
```
[Logging]
Level=1
;LogFile=LMAA654A.log
```

Figure 5 – *LMAA654A.INI*

3.5 Uninstalling the Lexmark PrintCryption PS3 Print Driver

To uninstall the Lexmark PrintCryption PS3 print driver, perform the following steps.

- Navigate to Windows "Printers and Faxes".
- Click on "Lexmark PrintCryption PS3".
- Click on "Delete this printer".
- Click on "Yes" to confirm the deletion.
- Delete the installation directory (by default *C:\WINDOWS\system32\spool\drivers\w32x86\3*).



☑ Enable Encrypted Printing

Encryption Settings

Key Length (bits): 128

Cipher Mode: CBC

Certificate

Date: 8/22/2007

Status: Installed

Figure 6 – Encryption Configuration

3.6 Zeroizing Keys and other CSPs

Zeroization of keys and other CSPs is controlled and performed by the print driver. Zeroization of ephemeral keys and other CSPs may be manually invoked by rebooting the computer on which the module is running. Uninstalling the Lexmark PrintCryption PS3 print driver results in zeroization of all keys and other CSPs.

4 Acronyms

Table 6 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
BIOS	Basic Input/Output System
CBC	Cipher Block Chaining
CD-ROM	Compact Disc Read-Only Memory
CFB	Cipher Feedback
CMVP	Cryptographic Module Validation Program
CPU	Central Processing Unit
CSP	Critical Security Parameter
DDI	Device Driver Interface
DLL	Dynamic Link Library
ECB	Electronic Codebook
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FCC	Federal Communication Commission
FIPS	Federal Information Processing Standard
GDI	Graphics Device Interface
GPC	General-Purpose Computer
HDD	Hard Disk
HMAC	Keyed-Hash Message Authentication Code
IDE	Integrated Drive Electronics
IR	Infrared
ISA	Instruction Set Architecture
IV	Initialization Vector
KAT	Known Answer Test
MAC	Message Authentication Code
N/A	Not Applicable
NIST	National Institute of Standards and Technology
OS	Operating System
PCI	Peripheral Component Interconnect
PKCS	Public Key Cryptography Standards
PPD	PostScript Printer Description

Acronym	Definition
RAM	Random Access Memory
RNG	Random Number Generator
ROM	Read-Only Memory
RSA	Rivest, Shamir and Adleman
SHA	Secure Hash Algorithm
UI	User Interface
USB	Universal Serial Bus