VMware, Inc.

VMware Java JCE (Java Cryptographic Extension) Module

Software Version: 1.0

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level: 1
Document Version: 1.0

Prepared for:

VMware, Inc.
3401 Hillview Ave
Palo Alto, CA 94304
United States of America

Phone: +1 (877) 486-9237
Email: info@vmware.com
http://www.vmware.com

Prepared by:

Corsec Security, Inc.
13135 Lee Jackson Memorial Hwy., Suite 220
Fairfax, VA 22033
United States of America

Phone: +1 (703) 267-6050
Email: info@corsec.com
http://www.corsec.com

Table of Contents

Table of Figures

List of Tables

1 Introduction

1.1 Purpose

This is a non-proprietary Cryptographic Module Security Policy for the VMware Java JCE (Java Cryptographic Extension) Module (Software Version: 1.0) from VMware, Inc. This Security Policy describes how the VMware Java JCE (Java Cryptographic Extension) Module meets the security requirements of Federal Information Processing Standards (FIPS) Publication 140-2, which 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) and the Communications Security Establishment (CSE) Cryptographic Module Validation Program (CMVP) website at http://csrc.nist.gov/groups/STM/cmvp.

This document also describes how to run the module in a secure FIPS-Approved mode of operation. This policy was prepared as part of the Level 1 FIPS 140-2 validation of the module. The VMware Java JCE (Java Cryptographic Extension) Module is referred to in this document as VMware JCE Module, crypto module, or the module.

1.2 References

This document deals only with 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 VMware website (http://www.vmware.com) contains information on the full line of products from VMware.
- The CMVP website (http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm) contains contact information for individuals to answer 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 document
- Finite State Model document
- Other supporting documentation as additional references

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

2 VMware Java JCE (Java Cryptographic Extension) Module

2.1 VMware Overview

VMware, Inc. is a global leader in virtualization and cloud infrastructure, delivering customer-proven solutions that accelerate IT by reducing complexity and enabling more flexible, agile service delivery. VMware enables enterprises to adopt a cloud model that addresses their unique business challenges. VMware's approach accelerates the transition to cloud computing while preserving existing investments and improving security and control.

2.1.1 VMware vCloud Networking and Security

One of the many virtualization products that VMware offers is vCloud Networking and Security (vCNS). vCNS provides software-defined networking and security built into their virtual infrastructure. vCNS is a networking and security solution which provides virtual firewalls, virtual private networks (VPNs), and load balancing to virtual datacenters and private cloud deployments. Three key Virtual Security Appliances are part of the vCNS solution:

- vShield Manager - A central management and reporting tool for vCNS components
- vShield Edge - A networking and security gateway for protecting virtual data centers
- vShield App - A virtual firewall to protect critical applications on virtual machines

vShield Manager is installed as a virtual appliance on any VMware vSphere ESXi host. Operators using vShield Manager are able to install, configure, and maintain vCNS components. Management of vCNS components through vShield Manager requires remote access to the virtual appliance via a secure HTTPS[1] connection. The web-based Graphical User Interface (GUI) is provided by vShield Manager as a Java Web Application hosted by an internal Tomcat server. The Tomcat server provides secure web connections to the vShield Manager Virtual Appliance, securely encrypts operator data, and generates keys and self-signed certificates. The Tomcat sever achieves this cryptographic functionality by utilizing the VMware Java JCE (Java Cryptographic Extension) Module.

2.1.2 VMware Java JCE (Java Cryptographic Extension) Module

The VMware Java JCE (Java Cryptographic Extension) Module (VMware JCE Module) is a software cryptographic module containing a set of cryptographic functions available to the Tomcat server located within vShield Manager. A well-defined Application Programming Interface (API) provides the module's cryptographic functions. These functions facilitate the secure transfer of information between vShield Manager's Tomcat server and an external web client. The Tomcat server uses the following FIPS-Approved security functions of the VMware Java JCE (Java Cryptographic Extension) Module:

- Symmetric key functions using AES[2] and Triple DES[3]
- Hashing functions using SHA[4]
- Asymmetric key functions using RSA[5] and DSA[6]

[1] HTTPS – Secure Hyper-Text Transfer Protocol
[2] AES – Advanced Encryption Standard
[3] DES – Data Encryption Standard
[4] SHA – Secure Hash Algorithm
[5] RSA – Rivest, Shamir, Adleman

- Random number generation using NIST SP[7] 800-90A Hash_DRBG[8]

Figure 1 provides a deployment scenario for the major components of the VMware vCloud Networking Security. Included are the interactions between vShield Manager, vShield Edge, and vShield App. The VMware Java JCE (Java Cryptographic Extension) Module is shown operating within the vShield Manager supporting the Tomcat server. The Tomcat server terminates external HTTPS[9] connections to host a Web Application for the vShield Manager. The server also leverages the module in order to provide encryption of user data before storing it to a database.

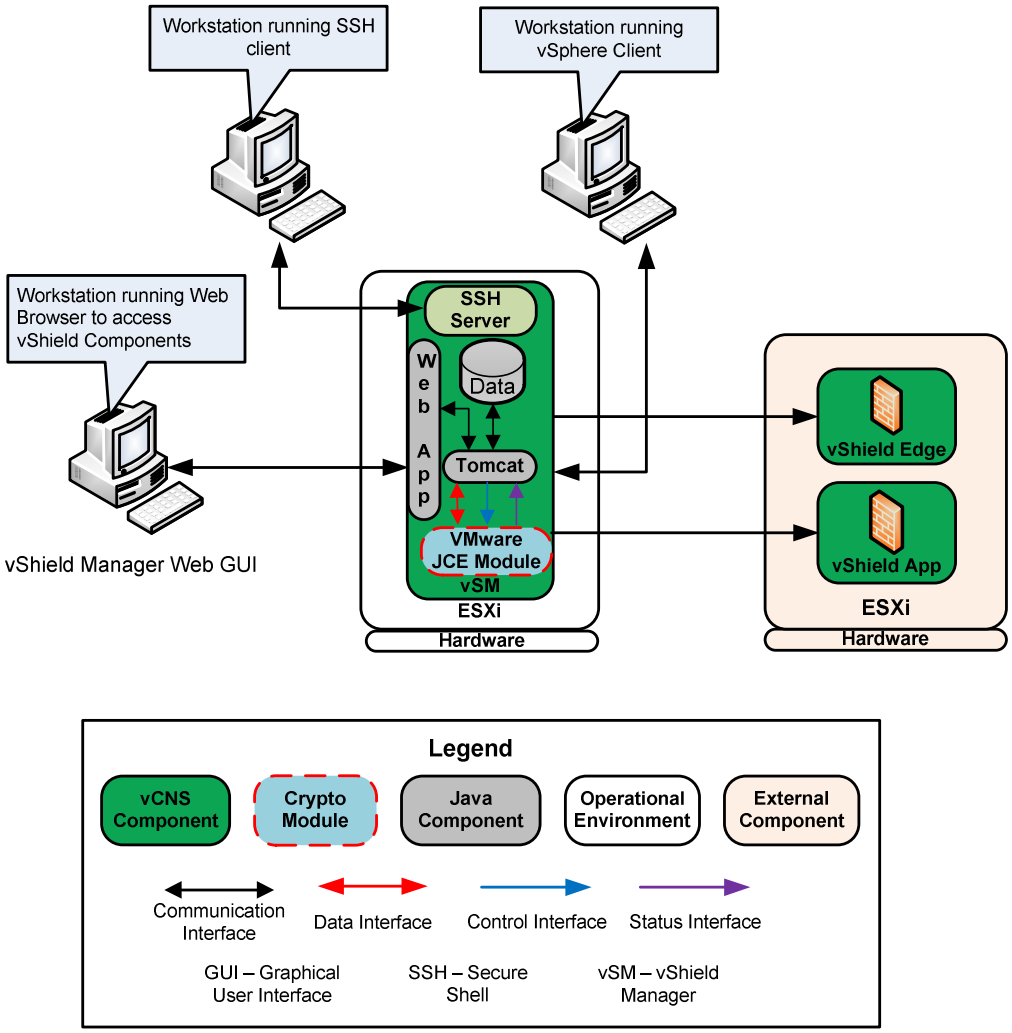

Figure 1 – vCloud Networking and Security Deployment Scenario

[6] DSA – Digital Signature Algorithm
[7] SP – Special Publication
[8] DRBG – Deterministic Random Bit Generator
[9] The HTTPS protocol has not be reviewed or tested by the CAVP and CMVP

The VMware Java JCE (Java Cryptographic Extension) Module is validated at the FIPS 140-2 Section levels shown in Table 1.

Table 1 – Security Level Per FIPS 140-2 Section

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[10]	1
9	Self-tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The VMware Java JCE (Java Cryptographic Extension) Module is a software cryptographic module with a multi-chip standalone embodiment. The overall security level of the module is 1. The module was tested and found to be FIPS 140-2 compliant on an HP ProLiant DL380e Gen8 Server running an Intel Xeon E5-2430 processor executing VMware vSphere Hypervisor (ESXi) 5.5 with VMware vCloud Networking and Security 5.5.0a vShield Manager Operating System (OS) running as the guest OS.

VMware, Inc. affirms that the VMware Java JCE (Java Cryptographic Extension) Module runs in its configured, Approved mode of operation on the following binarily compatible platforms executing VMware vSphere Hypervisor (ESXi) 5.5:

- A general purpose computing platform with an AMD Opteron x86 Processor executing VMware vCloud Networking and Security 5.5.0a vShield Manager OS
- A general purpose computing platform with an Intel Core i3, Core i5, Core i7, and Xeon x86 Processor executing VMware vCloud Networking and Security 5.5.0a vShield Manager OS

Because the VMware Java JCE (Java Cryptographic Extension) Module is defined as a software cryptographic module, it possesses both a physical cryptographic boundary and a logical cryptographic boundary. Sections 2.2.1 and 2.2.2 describe the physical and logical boundaries of the module.

2.2.1 Physical Cryptographic Boundary

As a software cryptographic module, there are no physical protection mechanisms implemented. Therefore, the module must rely on the physical characteristics of the host device. The hard enclosure around the host device is the physical boundary of the cryptographic module. The HP ProLiant Server is manufactured with production-grade material in order to provide physical security to the module. See Figure 2 for a block diagram of the host device.

[10] EMI/EMC – Electromagnetic Interference / Electromagnetic Compatibility

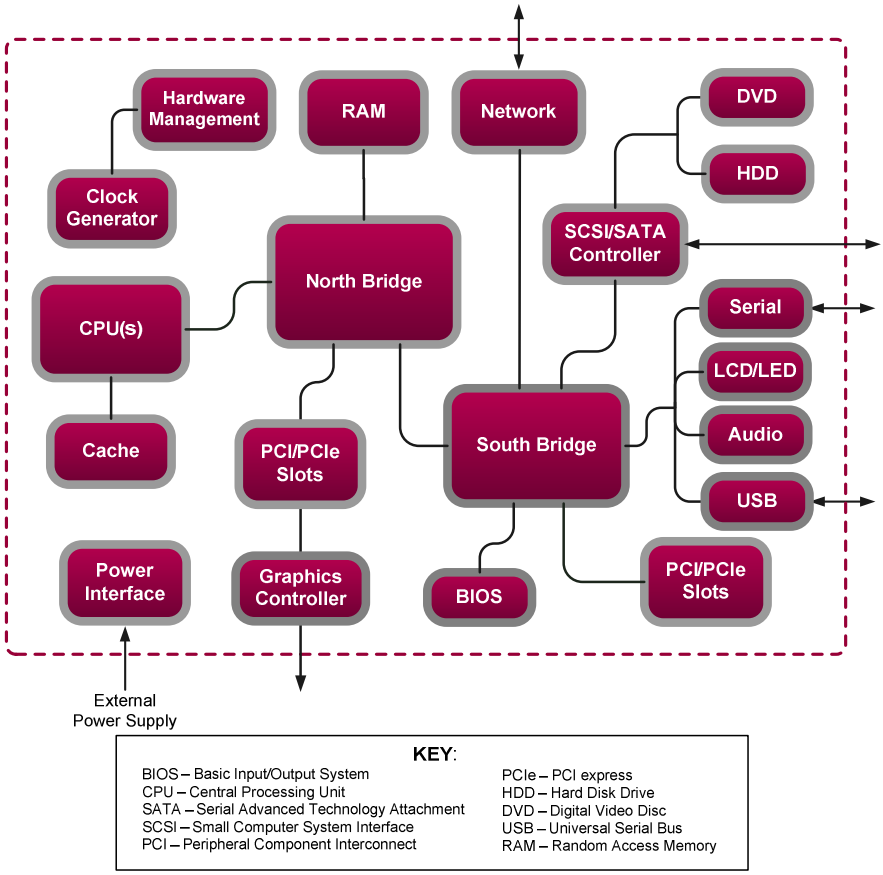

Figure 2 – HP ProLiant Server Block Diagram

2.2.2 Logical Cryptographic Boundary

Figure 3 shows a logical block diagram of the module and its surrounding software components, as well as the module's logical cryptographic boundary. The files and binaries that make up the cryptographic module are shown as the "VMware JCE Module" in Figure 3. The module is a cryptographic provider to the Java Runtime Environment (JRE). Java-based applications such as the Tomcat server call the module's services through the JRE. The module's logical boundary is a contiguous perimeter that surrounds all memory-mapped functionality provided by the module when loaded and stored in the host platform's memory.

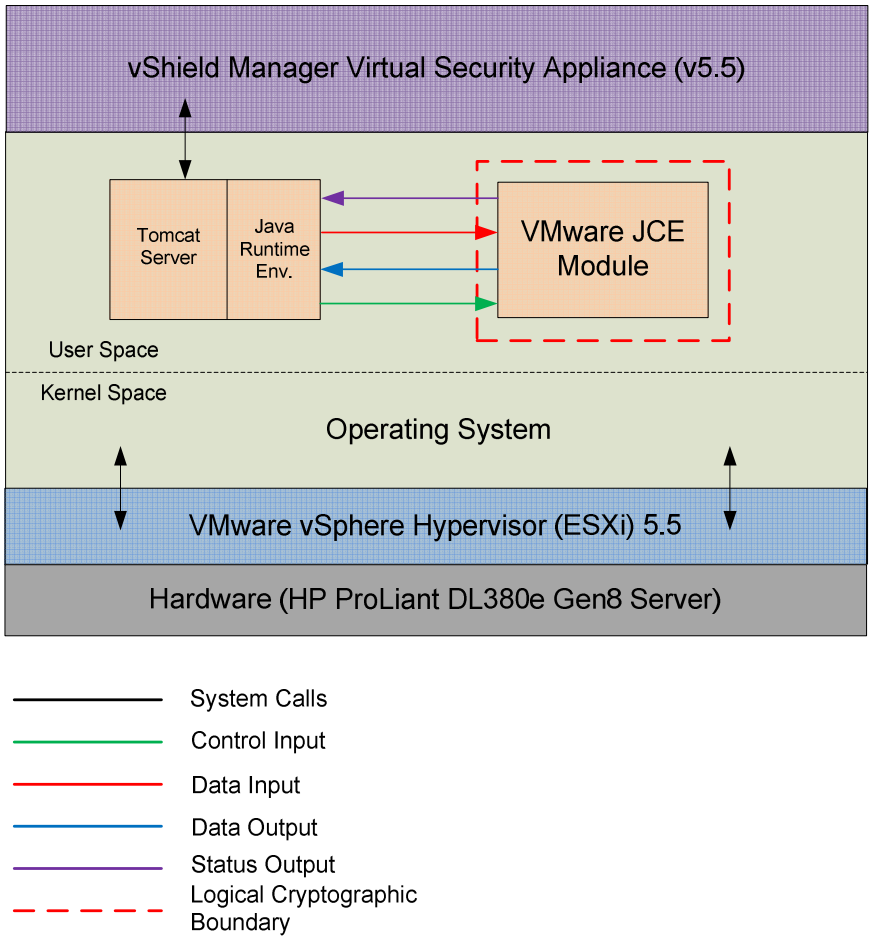

Figure 3 – VMware JCE Module Logical Cryptographic Boundary

2.3 Module Interfaces

The module's logical interfaces exist at a low level in the software as an API. Both the API and physical interfaces are categorized into the following interfaces defined by FIPS 140-2:

- Data input
- Data output
- Control input
- Status output
- Power input

As a software module, the module has no physical characteristics. Thus, the module's manual controls, physical indicators, and physical and electrical characteristics are those of the host device. A mapping of the FIPS 140-2 logical interfaces, the physical interfaces, and the module interfaces can be found in Table 2 below.

Table 2 – FIPS 140-2 Logical Interface Mappings

FIPS Interface	Physical Interface	Module Interface (API)
Data Input	Network port, Serial port, SCSI/SATA Controller, USB port	Method calls that accept, as their arguments, data to be used or processed by the module
Data Output	Network port, Serial port, SCSI/SATA Controller, USB port	Arguments for a method that specify where the result of the method is stored
Control Input	Network port, Serial port, USB port, Power button	Method calls utilized to initiate the module and the method calls used to control the operation of the module
Status Output	Network port, Serial port, USB port, Graphics controller	Thrown exceptions for method calls
Power Input	AC Power socket	Not applicable

2.4 Roles and Services

There are two roles in the module (as required by FIPS 140-2) that operators may assume: a Crypto Officer (CO) role and a User role. As the module does not support an authentication mechanism, roles are assumed implicitly through the execution of either a CO or User service. Each role and their corresponding services are detailed in the sections below. Please note that the keys and Critical Security Parameters (CSPs) listed in Table 3 and Table 4 below indicate the types of access required using the following notation:

- R – Read: The CSP is read.
- W – Write: The CSP is established, generated, modified, or zeroized.
- X – Execute: The CSP is used within an Approved or Allowed security function or authentication mechanism.

2.4.1 Crypto Officer Role

To assume the CO role, an operator of the module will perform one of the services listed in Table 3. The CO has the ability to enter and exit FIPS mode, run self-tests on demand, show status, and zeroize all keying material.

Table 3 – Crypto Officer Services

Service	Description	CSP and Type of Access
Initialize module	Performs integrity check and power-up self-tests	None
Show status	Returns the current mode of the module	None
Run self-tests on demand	Performs power-up self-tests	None
Zeroize keys	Zeroizes and de-allocates memory containing sensitive data	All keys – W

2.4.2 User Role

To assume the User role, an operator of the module will perform one of the services listed in Table 4. The User has the ability to generate random numbers, symmetric and asymmetric keys, and digital signatures.

Table 4 – User Services

Service	Description	CSP and Type of Access
Generate random number	Returns the specified number of random bits to calling application	DRBG Seed – WRX DRBG C Value – WRX DRBG V Value – WRX DRBG Entropy – WRX
Generate message digest	Compute and return a message digest using SHS algorithms	None
Generate keyed hash (HMAC)	Compute and return a message authentication code	HMAC[11] key – RX
Generate Cipher Hash (CMAC[12])	Compute and return a cipher message authentication code	AES CMAC Key – RX Triple-DES CMAC Key – RX
Generate symmetric key	Generate and return the specified type of symmetric key	Triple-DES Key – W
Symmetric encryption	Encrypt plaintext using supplied key and algorithm specification	AES key – RX Triple-DES key – RX
Symmetric decryption	Decrypt ciphertext using supplied key and algorithm specification	AES key – RX Triple-DES key – RX
Generate asymmetric key pair	Generate and return the specified type of asymmetric key pair	RSA private/public key – W DSA private/public key – W
Key Wrapping	Perform key wrap with RSA public key, AES key, and Triple-DES Key	RSA Public Key – RX AES Key – RX Triple-DES Key – RX
Key Unwrapping	Perform key unwrap with RSA private key, AES key, and Triple-DES Key	RSA Private Key – RX AES Key – RX Triple-DES Key – RX
Signature Generation	Generate a signature for the supplied message using the specified key and algorithm	RSA private key – RX DSA private key – RX
Signature Verification	Verify the signature on the supplied message using the specified key and algorithm	RSA public key – RX DSA public key – RX

2.5 Physical Security

The VMware Java JCE (Java Cryptographic Extension) Module is a software module, which FIPS defines as a multi-chip standalone cryptographic module. As such, it does not include physical security mechanisms. Thus, the FIPS 140-2 requirements for physical security are not applicable.

2.6 Operational Environment

The module was tested and found to be compliant with FIPS 140-2 requirements on an HP ProLiant DL380e Gen8 Server with an Intel Xeon E5-2430 processor running VMware vSphere Hypervisor (ESXi) 5.5 and VMware vCloud Networking and Security 5.5.0a vShield Manager OS. The cryptographic module

[11] HMAC – (keyed-) Hash-based Message Authentication Code
[12] CMAC – Cipher-based Message Authentication Code

will utilize the Java Virtual Machine (JVM) provided by Sun JRE v6.0. The JVM is responsible for relaying information from calling applications to the cryptographic module. All cryptographic keys and CSPs are under the control of the OS, which protects the module's CSPs against unauthorized disclosure, modification, and substitution. The module only allows access to CSPs through its well-defined API.

2.7 Cryptographic Key Management

2.7.1 FIPS-Approved Algorithms

The module implements the FIPS-Approved algorithms listed in Table 5 below.

Table 5 – FIPS-Approved Algorithm Implementations

Algorithm	Certificate Number
AES in ECB[13], CBC[14], CFB[15]-128, OFB[16], and CMAC modes encrypt/decrypt with 128-, 192- and 256-bit keys	2704
Triple-DES in ECB, CBC, CFB-8, CFB-64, and CMAC modes encrypt/decrypt; KO[17] 1	1623
RSA (FIPS 186-4) Key Generation with 2048- and 3072-bit key range	1402
RSA (PKCS[18] #1 v1.5) Signature Generation and Verification	1402
RSA (PSS[19]) Signature Generation and Verification	1402
DSA (FIPS 186-4) Key Generation with 2048- and 3072-bit keys	825
DSA Signature Generation and Verification	825
SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 hash	2271
HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 keyed hash	1685
SP 800-90A Hash_DRBG	446

The module employs the following key establishment methodologies, which are allowed for use in a FIPS-Approved mode of operation:

- RSA (key wrapping; key establishment methodology provides between 112 and 128 bits of encryption strength; non-compliant less than 112 bits of encryption strength)
- AES (Cert# 2704, key wrapping; key establishment methodology provides between 128 and 256 bits of encryption strength)
- Triple-DES (Cert# 1623, key wrapping; key establishment methodology provides 112 bits of encryption strength)

[13] ECB – Electronic Codebook
[14] CBC – Cipher Block Chaining
[15] CFB – Cipher Feedback
[16] OFB – Output Feedback
[17] KO – Keying Option
18 PKCS – Public-Key Cryptography Standards
[19] PSS – Probabilistic Signature Scheme

Caveats:

- Additional information concerning SHA-1, RSA, and DSA and specific guidance on transitions to the use of stronger cryptographic keys and more robust algorithms is contained in NIST Special Publication 800-131A.
- The module generates cryptographic keys whose strengths are modified by available entropy; No assurance of the minimum strength of generated keys.

2.7.2 Non-Approved Algorithms

The module employs non-Approved cryptographic algorithms, which are accessible by the operator of the module during module operation. The use of these algorithms with their associated services leads the module to operate in the non-Approved mode of operation. Their use, while operating in the FIPS-Approved mode, is strictly prohibited. Table 6 lists the non-Approved algorithms and their associated services.

Table 6 – Non-Approved Algorithm Implementations and Services

Algorithm	Service
RC2[20]	Encryption; Decryption
RC4	Encryption; Decryption
TWOFISH	Encryption; Decryption
IES[21]/ECIES[22]	Encryption; Decryption
DES	Encryption; Decryption
Triple-DES (2-key) [23]	Encryption; Decryption
MD2[24]/MD5	Hashing
RIPE MD	Hashing
TIGER	Hashing
ISO9797[25] Alg3 MAC	Hash-based Message Authentication Code
RSA	Key Generation; Signature Generation; Key Wrapping (Key size < 2048)
DSA	Key Generation; Signature Generation (Key size < 2048)
SHA-1	Signature Generation

[20] RC – Rivest Cipher
[21] IES – Integrated Encryption Scheme
[22] ECIES – Elliptic Curve IES
[23] To use the two-key Triple-DES algorithm to encrypt data or wrap keys in an Approved mode of operation, the module operator shall ensure that the same two-key Triple-DES key is not used for encrypting data (or wrapping keys) with more than 2^20 plaintext data (or plaintext keys). Please refer to Appendix A of SP 800-131A for restriction information regarding its use until December 31, 2015
[24] MD – Message Digest
[25] ISO – International Organization for Standards

2.7.3 Critical Security Parameters

The module supports the CSPs listed below in Table 7.

Table 7 – List of Cryptographic Keys, Cryptographic Key Components, and CSPs

CSP	CSP Type	Generation[26] / Input	Output	Storage	Zeroization	Use
AES key	AES 128-, 192-, 256-bit key	API call parameter	Output via GPC[27] INT path[28]	Plaintext in volatile memory	Reboot OS; Cycle host power	Encryption, Decryption
AES CMAC Key	AES CMAC 128-, 192, 256-bit key	API call parameter	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Message Authentication with AES
Triple-DES key	Triple-DES 168-bit secure key	API call parameter or internally generated	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Encryption, decryption
Triple-DES CMAC Key	Triple-DES CMAC 168-bit key	API call parameter	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Message Authentication with Triple-DES
HMAC key	160- to 512-bit HMAC Key	API call parameter	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Message Authentication with SHA-1 and SHA-2 family
RSA private key	RSA 2048-, 3072-bit key	API call parameter or internally generated	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Signature generation, key unwrapping
RSA public key	RSA 2048-, 3072-bit key	API call parameter or internally generated	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Signature verification, key wrapping

[26] The module complies with IG 7.8 Scenario 1 for symmetric key generation as well as the seed supplied to the algorithm for generating asymmetric keys
[27] GPC – General Purpose Computer
[28] GPC INT Path defined in Implementation Guidance Section 7.7

CSP	CSP Type	Generation[26] / Input	Output	Storage	Zeroization	Use
DSA private key	DSA 224-bit key	API call parameter or internally generated	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Signature generation
DSA public key	DSA 2048-bit key	API call parameter or internally generated	Output via GPC INT path	Plaintext in volatile memory	Reboot OS; Cycle host power	Signature verification
DRBG Seed	880-bit random value	API call parameter or Internally generated	Never	Plaintext in volatile memory	Reboot OS; Cycle host power	Seed input to SP 800-90 Hash_DRBG
DRBG Entropy	440-bit random value	API call parameter or Internally generated	Never	Plaintext in volatile memory	Reboot OS; Cycle host power	Entropy input to SP 800-90 Hash_DRBG
Hash DRBG V value	Internal hash DRBG state value	Internally generated	Never	Plaintext in volatile memory	Reboot OS; Cycle host power	Used for SP 800-90 Hash_DRBG
Hash DRBG C value	Internal hash DRBG state value	Internally generated	Never	Plaintext in volatile memory	Reboot OS; Cycle host power	Used for SP 800-90 Hash_DRBG

2.8 Self-Tests

Cryptographic self-tests are performed by the module when the module begins operation in the FIPS-Approved mode as well as when a random number or asymmetric key pair is created. The following sections list the self-tests performed by the module, the expected error status, and any error resolutions.

2.8.1 Power-Up Self-Tests

Power-up self-tests are automatically performed by the module when the module begins operation in the FIPS-Approved mode. The list of power-up self-tests that follows may also be run on-demand when the CO restarts the JRE or reboots the OS. The module will perform the listed power-up self-tests to successful completion. During the execution of self-tests, data output from the module is inhibited.

If any of the self-tests fail, the module will return an error to the JRE and enter an error state. After entering the error state, all subsequent calls to the module requiring cryptographic operation or data output will be rejected, ensuring that these abilities of the module are inhibited. In order to resolve a cryptographic self-test error, the JRE must unload the module and then reload it. If the error persists, the module must be reinstalled.

The VMware Java JCE (Java Cryptographic Extension) Module performs the following Power-up Self-tests:

- Software integrity check (HMAC SHA-1)
- Known Answer Tests (KATs)
    - AES KAT (Encrypt)
    - AES KAT (Decrypt)
    - Triple-DES KAT (Encrypt)
    - Triple-DES KAT (Decrypt)
    - RSA KAT (Signature Generation)
    - RSA KAT (Signature Verification)
    - HMAC with SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 KAT
    - SP 800-90A Hash_DRBG KAT
- DSA Pairwise Consistency Test

2.8.2 Conditional Self-Tests

Conditional self-tests are performed by the module whenever a new random number or a new asymmetric key pair is generated. If an error is encountered during an RSA or DSA pairwise consistency test, the module will return an error to the JRE and enter an error state. After entering the error state, all subsequent calls to the module requiring cryptographic operation or data output will be rejected. The JRE is responsible for resolving the error and returning the module to an operational state. This usually consists of unloading and reloading the module. No data will be returned by the module and the operation must be performed again. If the error persists, the module must be reinstalled.

The VMware Java JCE (Java Cryptographic Extension) Module performs the following conditional self-tests:

- SP 800-90A Hash_DRBG Continuous RNG Test
- RSA Pairwise Consistency Test for key pair generation
- DSA Pairwise Consistency Test for key pair generation

2.8.3 Critical Functions Tests

The SP 800-90A Hash_DRBG employed by the cryptographic module includes four critical functions. These critical functions include instantiation, generation, reseed, and uninstantiation. Each function is tested by the module during the module's power-up self-tests. If any of these critical functions fail, the

module will return an error to the JRE and will enter an error state. All subsequent calls to the module requiring cryptographic operation or data output will be rejected. The JRE will then proceed to unload and reload the module in order to reattempt these critical functions tests. If the error persists, the module must be reinstalled.

The VMware Java JCE (Java Cryptographic Extension) Module performs the following critical functions tests:

- DRBG Instantiate Critical Function Test
- DRBG Generate Critical Function Test
- DRBG Reseed Critical Function Test
- DRBG Uninstantiate Critical Function Test

2.9 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 VMware Java JCE (Java Cryptographic Extension) Module meets Level 1 requirements for FIPS 140-2. The sections below describe how to place and keep the module in FIPS-approved mode of operation.

3.1 Crypto Officer Guidance

Installation and operation of the VMware Java JCE (Java Cryptographic Extension) Module requires the proper installation of the vShield Manager Virtual Appliance. The sections below provide a brief summary of the installation procedures for vShield Manager and additional associated products. For a more comprehensive instruction set, please refer to the *vShield Installation and Upgrade Guide* provided by VMware. The VMware Java JCE (Java Cryptographic Extension) Module is preconfigured to always operate in the FIPS-Approved mode of operation.

All guides mentioned within in these instructions are freely available for download at http://www.vmware.com. These instructions assume that the CO is familiar with VMware vSphere 5.5 and VMware vShield 5.5 products.

3.1.1 Initial Setup

Prior to the secure installation of the vShield Manager Virtual Appliance, the CO shall prepare the virtual environment required to securely operate the virtual appliance. This includes installing the latest version of VMware vSphere 5.5 (see *vSphere Installation and Setup*). Included in this installation is the VMware vSphere Hypervisor (ESXi) 5.5, the vSphere 5.5 vSphere Client, and the vSphere 5.5 vCenter Server, all of which are prerequisites to installing the vShield Manager Virtual Appliance.

After installing the VMware vSphere 5.5 virtual environment, the CO shall log into the vCenter Server using the vSphere Client and follow the instructions provided in Chapter 3 of *vShield Installation and Upgrade Guide* to securely install and configure VMware vShield Manager Virtual Appliance.

3.1.2 Secure Installation

In order to install the VMware Java JCE (Java Cryptographic Extension) Module, the CO shall follow the installation instructions provided in Chapter 3 of *vShield Installation and Upgrade Guide* in order to securely install and configure the vShield Manager Virtual Appliance. A brief summary of the installation steps is provided:

- Obtain vShield Manager OVA[29] File
- Log into the vCenter Server using vSphere Client
- Determine which ESXi host the virtual appliance will be installed on[30]
- Create a new port group
- Deploy a new OVF[31] template, selecting the vShield Manager OVA file
- Complete the installation procedures using the installation wizard
- Complete the network configuration steps
- Link vShield Manager with vCenter Server

Successful completion of installing a vShield Manager Virtual Appliance will be indicated by the appearance of a vShield Manager Virtual Appliance on the ESXi host which the CO specified. Using the vSphere Client, the CO can determine that the virtual appliance is operational. Troubleshooting is available in *vShield Installation and Upgrade Guide*.

[29] OVA – Open Virtual Appliance
[30] This host should not be subject to downtime and multiple restarts
[31] OVF – Open Virtualization Format

3.1.3 VMware Java JCE (Java Cryptographic Extension) Module Secure Operation

Following the successful installation of the vShield Manager Virtual Appliance, the CO shall power on the virtual appliance. Powering on the vShield Manager Virtual Appliance will also power on the VMware Java JCE (Java Cryptographic Extension) Module. The VMware JCE Module is preconfigured to always operate in the FIPS-Approved mode of operation. The CO and User are incapable of operating the module in a non-Approved mode. The CO shall follow the guidelines offered in the *vShield Administration Guide* in order to securely configure and operate the VMware Java JCE (Java Cryptographic Extension) Module.

3.2 User Guidance

The VMware Java JCE (Java Cryptographic Extension) Module is designed for use by VMware security products. The user shall adhere to the guidelines of this Security Policy. The User does not have the ability to install or configure the module. Operators in the User role are able to use the services listed in Table 4. Users are responsible for reporting any irregular activity they notice to the CO. During operation, the User may check the status of the module by attempting to run any User service. If the service executes, the module is operating in FIPS Approved mode.

4 Acronyms

Table 8 provides definitions for the acronyms used in this document.

Table 8 – Acronyms

Acronym	Definition
AES	Advanced Encryption System
API	Application Programming Interface
BIOS	Basic Input/Output System
CBC	Cipher Block Chaining
CCM	Counter with CBC-MAC
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CO	Cryptographic Officer
CPU	Central Processing Unit
CSE	Communications Security Establishment
CSP	Critical Security Parameter
DES	Data Encryption Standard
DRBG	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
DVD	Digital Video Disc
ECB	Electronic Code Book
ECIED	Elliptic Curve IES
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FIPS	Federal Information Processing Standard
GPC	General Purpose Computer
GUI	Graphical User Interface
HDD	Hard Disk Drive
HMAC	(keyed-) Hash Message Authentication Code
HTTPS	Secure Hyper-Text Transfer Protocol
IES	Integrated Encryption Scheme
ISO	International Organization for Standards
JCE	Java Crypto Extension
JVM	Java Virtual Machine

Acronym	Definition
KAT	Known Answer Test
KO	Keying Option
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MD	Message Digest
NIST	National Institute of Standards and Technology
OFB	Output Feedback
OS	Operating System
OVA	Open Virtual Appliance
OVF	Open Virtualization Format
PCI	Peripheral Component Interconnect
PCIe	PCI express
PCT	Pairwise Consistency Test
PKCS	Public Key Cryptography Standard
PSS	Probabilistic Signature Scheme
RAM	Random Access Memory
RC	Rivest Cipher
RSA	Rivest Shamir and Adleman
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Interface
SHA	Secure Hash Algorithm
SP	Special Publication
USB	Universal Serial Bus
VPN	Virtual Private Network

Prepared by:
Corsec Security, Inc.

13135 Lee Jackson Memorial Highway, Suite 220
Fairfax, VA 22033
United States of America

Phone: +1 (703) 267-6050
Email: info@corsec.com
http://www.corsec.com