Blue Coat Systems, Inc.

Blue Coat Systems, Software Cryptographic Module

SW Version: 1.0

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level: 1
Document Version: 1.8

Prepared for:

Blue Coat Systems, Inc.
420 N. Mary Avenue
Sunnyvale, CA 94085
United States of America

Phone: +1 (801) 545-4100
Email: info@soleranetworks.com
http://www.soleranetworks.com

Prepared by:

Corsec Security, Inc.
13135 Lee Jackson Memorial Hwy
Suite 220
Fairfax, VA 22033

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 Blue Coat Systems, Software Cryptographic Module from Blue Coat Systems, Inc.. This Security Policy describes how the Blue Coat Systems, Software Cryptographic 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 Canada (CSEC) 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 Blue Coat Systems, Software Cryptographic Module is referred to in this document as the Software Cryptographic Module, cryptographic 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 Solera Networks, a Blue Coat company website (http://www.soleranetworks.com) contains information on the full line of products from Solera Networks, a Blue Coat company.
- 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
- Submission Summary
- Other supporting documentation as additional references

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

2 Blue Coat Systems, Software Cryptographic Module

2.1 Overview

Blue Coat develops tools that combine security intelligence and big data analytics to help organizations battle Advanced Persistent Threats (APTs) and Advanced Targeted Attacks (ATAs). This is accomplished through a combination of data collection, correlation, and enrichment that is made available to security analysts to help them stay ahead of today's evolving threat landscape. The Solera Networks DeepSee platform captures packets, indexes flows, and extracts files, from Layer 2 through Layer 7 data. This provides near real-time and retrospective visibility into security relevant network events.

The Software Cryptographic Module is incorporated in Solera DeepSee Software. Solera DeepSee Software acts as an unobtrusive network traffic recorder. Data crossing the network is captured and saved to storage. Once in storage, data can be played back to analysis applications, or can be sent to any other location on the network or to multiple applications and locations. It creates a complete record of network traffic (including both packet headers and payloads), facilitating regeneration, filtering, and playback for later analysis. Filtering and analysis tools can be applied during data capture or playback.

The major features of the Solera DeepSee Software are:

- Improved network security – Network administrators have comprehensive evidence to better protect against intruders, data leakage, and internal misuse.
- Flexible Deployment Technology – Solera DeepSee Software provides the flexibility to be deployed on any hardware platform allowing organizations of all sizes to benefit from deep packet capture and analysis to improve network performance and security.
- Increased network tool options – Solera DeepSee Software works with many management, analysis, and forensic tools (commercial, custom, and open source) to monitor, manage, and secure the network.

Figure 1 below shows the details of the typical deployment configuration of the Solera DeepSee Software. The following previously undefined acronyms appear in Figure 1:

- API – Application Programming Interface
- GBPS – Gigabits per second
- PCAP – Packet Capture

.

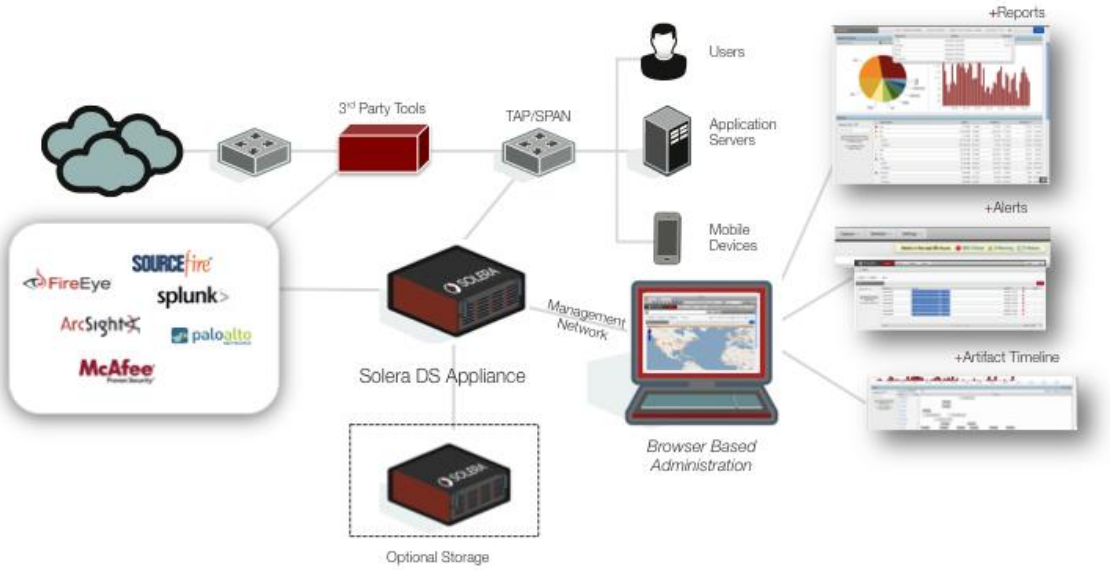

Figure 1 – Typical Deployment Configuration of Solera DeepSee Software

2.1.1 Software Cryptographic Module

The Blue Coat Systems, Software Cryptographic Module is a software shared library that is included with Solera DeepSee Software v6.5.0. It provides the primitive cryptographic services required by TLS[1] for secure communications. The module includes implementations of the following FIPS-Approved algorithms:

- Advanced Encryption Standard (AES)
- Triple Data Encryption Algorithm (TDES)
- Secure Hash Algorithm (SHA)
- Keyed-Hash Message Authentication Code (HMAC)
- Digital Signature Algorithm (DSA)
- RSA[2] signature generation and verification
- ANSI[3] X9.31 Pseudo Random Number Generator (PRNG)

The Software Cryptographic Module operates in a FIPS-Approved mode of operation when configured according to the Crypto-Officer guidance in this Security Policy and does not support a non-Approved mode of operation. It is validated at the FIPS 140-2 Section levels as indicated in Table 1 below.

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

[1] TLS – Transport Layer Security
[2] RSA – Rivest, Shamir, Adleman
[3] ANSI - American National Standards Institute

.

Section	Section Title	Level
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC[4]	1
9	Self-tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The Software Cryptographic Module is a software module with a multi-chip standalone embodiment. The overall security level of the module is 1. The Software Cryptographic Module is implemented in the C programming language and consists of a shared library that links to Solera DeepSee Software application components. The Solera DeepSee Software includes Solera Operating Environment v6.5.0 as its operating system. It is designed to execute on a host platform with a General Purpose Computer (GPC) hardware platform. The Blue Coat Systems, Software Cryptographic Module can also be installed on a supported virtual machine hypervisor. The cryptographic module was tested and found compliant on the VMware ESXi Server 5.0 on a Dell PowerEdge R720 and on a Dell PowerEdge R720 with dual Intel Xeon processors. The following sections define the physical and logical boundary of the Software Cryptographic Module.

While no claim can be made as to the correct operation of the module or the security strengths of the generated keys when ported to an operational environment which is not listed on the validation certificate, the vendor affirms that the module remains FIPS-complaint when executed on any of the supported platforms and environments:

- Dell model R620
- Dell model MD1200
- VMware Workstation
- VMware Player

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 platform. The physical boundary of the cryptographic module, whether running on a virtual hypervisor or on Dell R720 hardware, is defined by the hard enclosure around the host platform on which it runs. The module supports the physical interfaces of on the host platform. These interfaces include the integrated circuits of the system board, the CPU[5], network adapters, RAM[6], hard disk, device case, power supply, and fans. See Figure 2 for a Dell R720 block diagram.

[4] EMI/EMC – Electromagnetic Interference / Electromagnetic Compatibility
[5] CPU – Central Processing Unit
[6] RAM – Random Access Memory

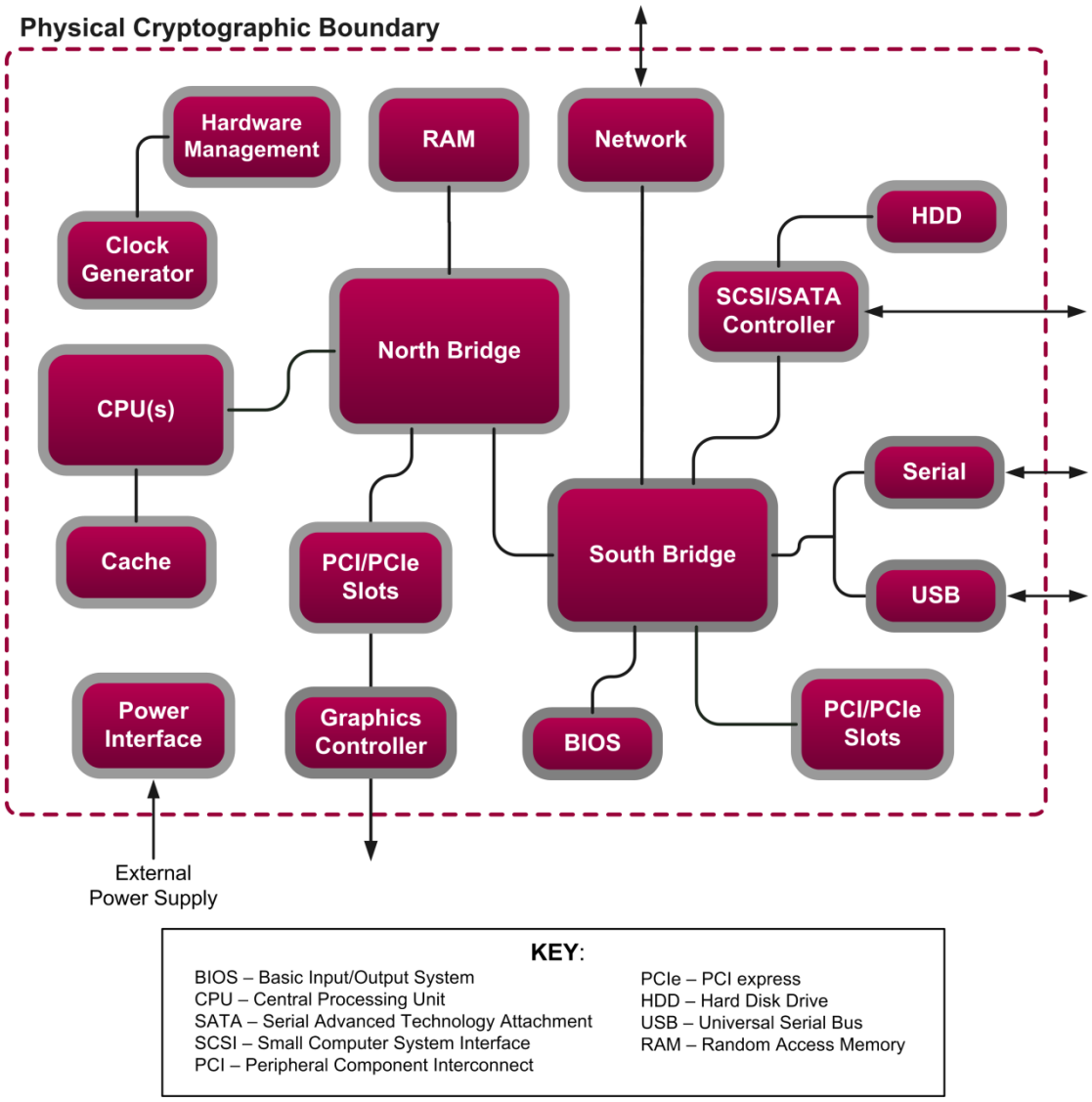

Figure 2 – Dell R720 Block Diagram

2.2.2 Logical Cryptographic Boundary

Figure 3 shows a logical block diagram of the module executing in memory and its interactions with surrounding software components when in the hardware configuration. Figure 3 also shows the module's logical cryptographic boundary. The module's services are designed to be called by other Solera software components.

.

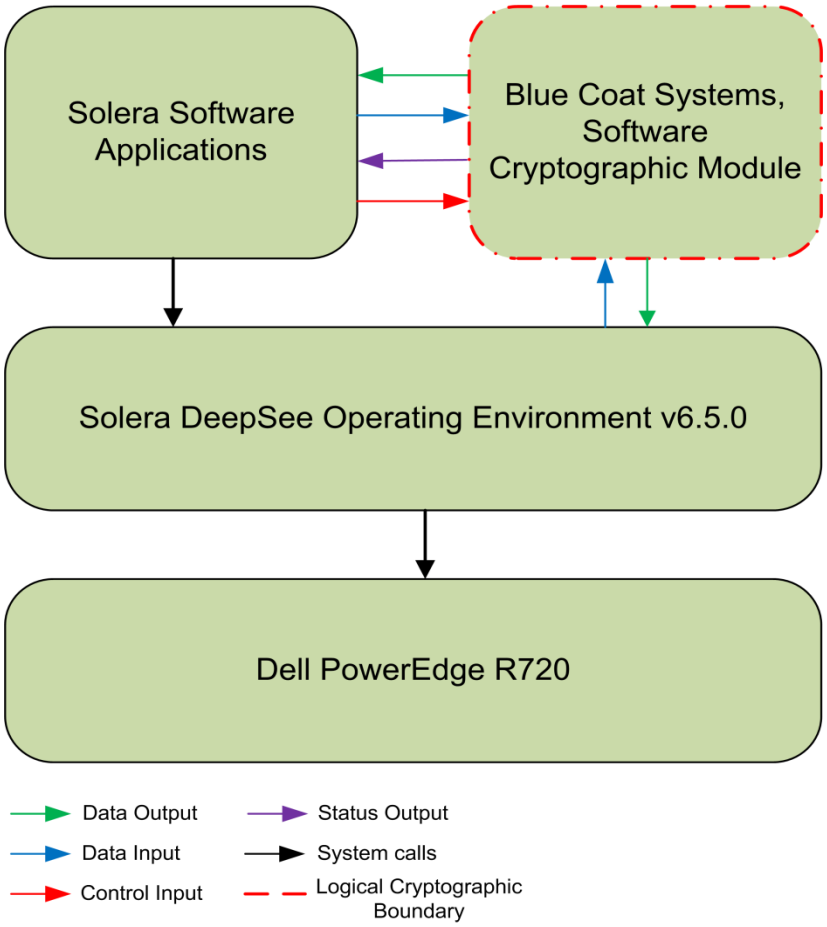

Figure 3 – Logical Block Diagram and Cryptographic Boundary for Hardware Configuration

Figure 4 shows a logical block diagram of the module executing in memory and its interactions with surrounding software components when in the virtual configuration. Figure 4 also shows the module's logical cryptographic boundary in the virtual configuration. The module's services are designed to be called by other Solera software components.

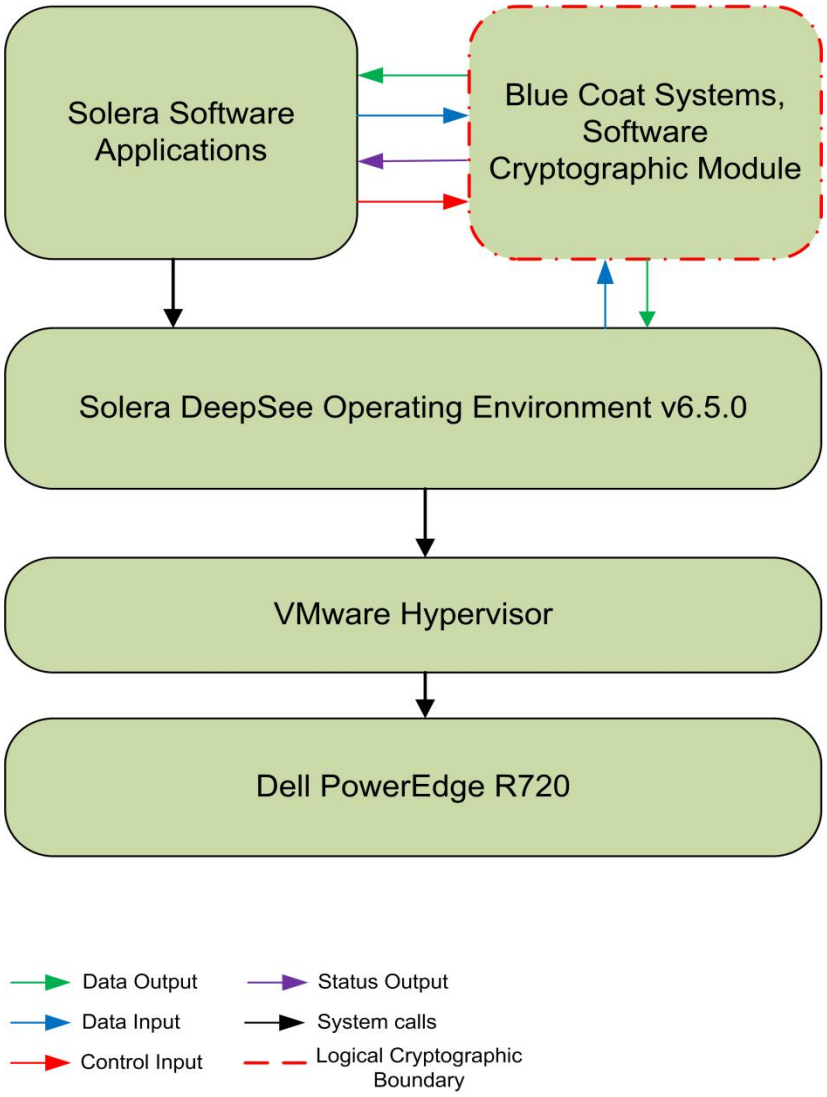

Figure 4 – Logical Block Diagram and Cryptographic Boundary for Virtual Configuration

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 can be categorized into the following interfaces defined by FIPS 140-2: Data Input, Data Output, Control Input, and Status Output. 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	USB ports (keyboard, mouse, data), network ports, serial ports, SCSI/SATA ports	Arguments for API calls that contain data to be used or processed by the module.

FIPS Interface	Physical Interface	Module Interface (API)
Data Output	Monitor, USB ports, network ports, serial ports, SCSI/SATA ports	Arguments for API calls that contain or point to where the result of the function is stored.
Control Input	USB ports (keyboard, mouse), network ports, serial ports, power switch	API Function calls and parameters that initiate and control the operation of the module.
Status Output	Monitor, network ports, serial ports	Return values from API function calls and error messages.
Power Input	Power Interface	N/A

2.4 Roles and Services

The Software Cryptographic Module supports the following two roles for operators, as required by FIPS 140-2: Crypto-Officer (CO) role and User role. As allowed by FIPS 140-2, the module does not perform authentication of any operators. Both roles are implicitly assumed when services are executed.

Note 1: Table 3 and Table 4 use the following definitions for entries in the "CSP[7] and Type of Access" column.

R – Read: *The plaintext CSP is read by the service.*
W – Write: *The CSP is established, generated, modified, or zeroized by the service.*
X – Execute: *The CSP is used within an Approved (or allowed) security function or authentication mechanism.*

Note 2: Input parameters of an API call that are not specifically a signature, hash, message, plaintext, ciphertext, or a key are NOT itemized in the "Input" column, since it is assumed that most API calls will have such parameters.

Note 3: The "Input" and "Output" columns are with respect to the module's logical boundary.

2.4.1 Crypto-Officer Role

The operator in the Crypto-Officer role installs, uninstalls, and administers the module via the Solera DeepSee Software interfaces. An operator assumes the CO role by invoking one of the following services:

Table 3 – Crypto-Officer Services

Service	Description	Input	Output	CSP and Type of Access
Initialize FIPS mode	Performs integrity checks and power-up self-tests. Sets the FIPS mode flag to on.	API call parameters	Status	Integrity check HMAC key, ANSI X9.31 PRNG seed, ANSI X9.31 PRNG seed key
Show status	Returns the current mode of the module (FIPS or non-FIPS).	None	Status	None

[7] CSP – Critical Security Parameter

.

Service	Description	Input	Output	CSP and Type of Access
Run self-tests on demand	Performs power-up self-tests.	None	Status	Integrity check HMAC key

2.4.2 User Role

The operator in the User role is a consumer of the module's security services. The role is assumed by invoking one of the following cryptographic services:

Table 4 – User Services

Service	Description	Input	Output	CSP and Type of Access
Generate random number (ANSI X9.31)	Returns the specified number of random bits to calling application.	API call parameters	Status, random bits	ANSI X9.31 RNG[8] seed – RWX ANSI X9.31 seed key – RX
Generate a symmetric key	Generate and return a symmetric key (AES, TDES).	API call parameters, ANSI X9.31, RNG seed	Status, key	ANSI X9.31 RNG seed – RX ANSI X9.31 seed key – RX AES – R,W TDES – R, W
Generate message digest (SHS[9])	Compute and return a message digest using SHS algorithms.	API call parameters, message	Status, hash	None
Generate keyed hash (HMAC)	Compute and return a message authentication code using HMAC-SHAx.	API call parameters, key, message	Status, hash	HMAC key – RX
Zeroize key	Zeroizes and de-allocates memory containing sensitive data.	Reboot or power cycle	Status	AES key – W TDES key – W HMAC key – W RSA private/public key – W DSA private/public key – W DH[10] components – W RNG seed – W
Symmetric encryption	Encrypt plaintext using supplied key and algorithm specification (TDES or AES)	API call parameters, key, plaintext	Status, ciphertext	AES key – RX TDES key – RX

[8] RNG – Random Number Generator
[9] SHS – Secure Hash Standard
[10] DH – Diffie-Hellman

.

Service	Description	Input	Output	CSP and Type of Access
Symmetric decryption	Decrypt ciphertext using supplied key and algorithm specification (TDES or AES)	API call parameters, key, ciphertext	Status, plaintext	AES key – RX TDES key – RX
Generate asymmetric key pair	Generate and return the specified type of asymmetric key pair (RSA or DSA)	API call parameters	Status, key pair	RSA private/public key – W DSA private/public key – W
RSA key wrapping	Wrap plaintext using RSA public key (used for key transport)	API call parameters, key, plaintext	Status, ciphertext	RSA public key – RX
RSA key unwrapping	Unwrap ciphertext using RSA private key (used for key transport)	API call parameters, key, ciphertext	Status, plaintext	RSA private key – RX
Diffie-Hellman primitive implementation*	Perform Diffie-Hellman primitive implementation	API call parameter	Status, key components	DH components – W
Signature Generation	Generate a signature for the supplied message using the specified key and algorithm (RSA or DSA)	API call parameters, key, message	Status, signature	RSA private key – RX, DSA private key – RX
Signature Verification	Verify the signature on the supplied message using the specified key and algorithm (RSA or DSA)	API call parameters, key, signature, message	Status	RSA public key – RX DSA public key – RX

*Diffie-Hellman primitive is implemented to perform in accordance with scenario 6 in section D.8 of FIPS 140-2 Implementation Guidance. This service is provided for calling process use and is not used to establish keys into the module.

2.4.3 Non-Approved Services

The following cryptographic services listed in Table 5 are not allowed in FIPS-Approved mode.

Table 5 – Non-Approved Services

Service	Cryptographic Function
Symmetric encryption/decryption	AES CFB1
Key establishment	Elliptic Curve Diffie-Hellman
Signature generation and verification	Elliptic Curve DSA

2.5 Physical Security

The Blue Coat Systems, Software Cryptographic 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 complaint on Solera Operating Environment v6.5.0, which is a proprietary OS and a Dell PowerEdge model R720 with dual Intel Xeon processors. The module was also tested and found complaint on Solera Operating Environment v6.5.0 running on VMware ESXi v5.0 on a Dell PowerEdge model R720 with dual Intel Xeon processors. All cryptographic keys and CSPs are under the control of the operating system, which protects the CSPs against unauthorized disclosure, modification, and substitution. The module only allows access to CSPs through its well-defined API. The tested operating system segregates user processes into separate process spaces. Each process space is an independent virtual memory area that is logically separated from all other processes. The Module functions entirely within the process space of the process that invokes it, and thus satisfies the FIPS 140-2 requirement for a single user mode of operation.

2.7 Cryptographic Key Management

The module implements the FIPS-Approved algorithms listed in Table 6 below.

Table 6 – FIPS-Approved Algorithm Implementations

Algorithm	Certificate Number
Symmetric Key Algorithm	
AES in ECB[11], CBC[12], CFB8[13], CFB128 and OFB[14] modes (128-,192-, 256-bits)	2153
Triple-DES in ECB, CBC, CFB8, CFB64, and OFB modes with 168-bit keys	1364
Asymmetric Key Algorithm	
RSA (ANSI X9.31) key generation (1024-, 1536-, 2048-, 3072-, 4096-bit keys) and signature generation/verification (1024-, 1536-, 2048-, 3072-, 4096-bit keys)	1108
RSA (PKCS[15] #1.5) signature generation/verification (1024-, 1536-, 2048-, 3072-, 4096-bit keys)	1108
RSA (PSS[16]) signature generation/verification (1024-, 1536-, 2048-, 3072-, 4096-bit keys)	1108
DSA signature generation/verification and key generation 1024-bit key	669
Secure Hashing Algorithm (SHA)	
SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	1873
Message Authentication Code (MAC)	
HMAC- SHA-1, -SHA-224, -SHA-256, -SHA-384, -SHA-512	1318
Pseudo Random Number Generation (PRNG)	

[11] ECB – Electronic Codebook
[12] CBC – Cipher-Block Chaining
[13] CFB – Cipher Feedback
[14] OFB – Output Feedback
[15] PKCS – Public-Key Cryptography Standards
[16] PSS – Probabilistic Signature Scheme

.

Algorithm	Certificate Number
ANSI X9.31 Appendix A.2.4 PRNG with AES 128-, 192-, and 256-bit keys	1101

NOTE: The following security functions have been deemed "deprecated" or "restricted" by NIST. Please refer to NIST Special Publication 800-131A for further details.

- two-key Triple DES for encryption
- ANSI X9.31 PRNG
- key lengths providing no more than 80 bits of security strength for digital signature generation

The module provides the following non-FIPS-Approved algorithms that are allowed in the FIPS-Approved mode:

- RSA key wrapping; key establishment methodology provides between 80 and 150 bits of encryption strength
- Diffie-Hellman primitive implementation provides between 80 and 219 bits of encryption strength

The module provides the following algorithms that are not allowed in the FIPS-Approved mode:

- Elliptic Curve Diffie-Hellman
- Elliptic Curve DSA
- AES CFB1

The module supports the CSPs listed below in Table 7.

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

Key	Key Type	Generation / Input	Output	Storage	Zeroization	Use
RSA Private key	1024-, 1536-, 2048-, 3072-, 4096-bit private keys	Internally generated	API call parameter	Plaintext in volatile memory	By API call, power cycle, or host reboot	Key exchange
		API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Signature generation, key unwrapping[1]
RSA Public Key	1024-, 1536-, 2048-, 3072-, 4096-bit public keys	Internally generated	API call parameter	Plaintext in volatile memory	By API call, power cycle, or host reboot	Key exchange
		API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Signature verification, key wrapping
Session Key	AES 128-, 192-, or 256-bit key in CBC, OFB,	API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Encryption, decryption

.

Key	Key Type	Generation / Input	Output	Storage	Zeroization	Use
	CFB, and ECB modes Triple DES 168-bit key in CBC, ECB, CFB, and OFB mode Keying Option 1 (Three-key)	Internally generated	API call parameter in encrypted form	Plaintext in volatile memory	By API call, power cycle, or host reboot	Encryption, decryption
ANSI X9.31 PRNG seed	128 bits of Random value[2]	Initialized by the module from an external NDRNG[17]	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Generate a random number
ANSI X9.31 PRNG seed key value	AES 128-, 192-, 256-bit key	Initialized by the module from an external NDRNG	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Generate a random number
HMAC Key	160-, 224-, 256-, 384-, and 512-bit keys	API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Message Authentication with SHA-1, -224, -256, -384, -512
DSA public key	DSA 1024-bit key	API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Signature Verification
DSA private key	DSA 160-bit key	API call parameter	Never exits the module	Plaintext in volatile memory	By API call, power cycle, or host reboot	Signature Generation
DH public key	public key	Internally generated	API call parameter	Plaintext in volatile memory	By API call, power cycle, host reboot	Key exchange
DH private key	private key	Internally generated	API call parameter	Plaintext in volatile memory	By API call, power cycle, host reboot	Key exchange

[1]RSA key wrapping provides between 80 and 150 bits of encryption strength.
[2]The entropy used by the FIPS-Approved ANSI X9.31 PRNG is acquired using an external, non-Approved NDRNG.

[17] NDRNG – Non-Deterministic Random Number Generator

2.7.1 Key Generation

The module uses an ANSI X9.31 Appendix A.2.4 PRNG implementation to generate cryptographic keys. This PRNG is FIPS-Approved as shown in Annex C to FIPS PUB 140-2 and complies with scenario one of IG 7.8. The ANSI X9.31 RNG is seeded using a seed key and seed gathered from a random pool filled with 48 bytes (estimated entropy strength on the tested system is 94 bits) of system data and internal resources such as time, user activity, and system activity. As the module's ANSI X9.31 RNG implementation generates random values of size 128 bits, it would take multiple calls to form a 256-bit key. Since no reseeding operation occurs, the total estimated strength for the two calls required to form a 256-bit key is 94 bits of entropy.

Caveat: The module generates cryptographic keys whose strengths are modified by available entropy – 94 bits.

2.7.2 Key Entry and Output

The cryptographic module itself does not support key entry or key output from its physical boundary. Keys are passed to the module as parameters from the applications resident on the host platform via the exposed APIs. Similarly, keys and CSPs exit the module via the well-defined exported APIs.

2.7.3 Key/CSP Storage and Zeroization

Symmetric, asymmetric, and HMAC keys are either provided by or delivered to the calling process, and are subsequently destroyed by the module at the completion of the API call. Keys and CSPs stored in RAM can be zeroized by a power cycle or a host platform reboot. The X9.31 PRNG seed and seed key are initialized by the module at power-up and remain stored in RAM until the module is uninitialized by a host platform reboot or power cycle. The HMAC key that is used to verify the integrity of the module is hard-coded within the module binary.

2.8 EMI/EMC

The Software Cryptographic Module is a software module. Therefore, the only electromagnetic interference produced is that of the host platform on which the Software Cryptographic Module resides and executes. FIPS 140-2 requires that the host platforms on which FIPS 140-2 testing is performed meet the Federal Communications Commission (FCC) EMI and EMC requirements for business use as defined in Subpart B, Class A of FCC 47 Code of Federal Regulations Part 15. However, all systems sold in the United States must meet these applicable FCC requirements.

2.9 Self-Tests

The modules implement two types of self-tests: power-up self-tests and conditional self-tests. Upon any self-test failure, the module reboots. Power-up self-tests can also be performed on demand by cycling the power on the host platform, calling the function FIPS_selftest(), or by reinitializing the module using the FIPS_mode_set() function.

2.9.1 Power-Up Self-Tests

The Software Cryptographic Module performs the following self-tests at power-up:

- Software integrity check
  - This test calculates an HMAC SHA-1 digest of the module and compares it to the pre-calculated digest stored in the module's associated digest file.
- Known Answer Tests (KAT)s
  - AES KAT for encryption
  - AES KAT for decryption
  - Triple-DES KAT for encryption
  - Triple-DES KAT for decryption
  - RSA KAT for signature generation
  - RSA KAT for signature verification
  - SHA-1 KAT
  - HMAC SHA-1 KAT
  - HMAC SHA-224 KAT
  - HMAC SHA-256 KAT
  - HMAC SHA-384 KAT
  - HMAC SHA-512 KAT
  - ANSI X9.31 PRNG KAT
- DSA pairwise consistency check

SHA-224, SHA-256, SHA-384, and SHA-512 do not have individual KATs, but are tested as part of their respective HMAC KATs. The RSA KAT test is a single function call that performs tests of both signature generation and signature verification. The DSA pairwise consistency check also is a single function that tests both signature generation and signature verification.

2.9.2 Conditional Self-Tests

The Software Cryptographic Module performs the following conditional self-tests:

- Continuous RNG test
- RSA pairwise consistency check for sign/verify and encrypt/decrypt
- DSA pairwise consistency check

If a conditional self-test fails, the module will enter an error state, during which cryptographic functionality and all data output is inhibited. To clear the error state, the CO must reinitialize the module.

.

2.10 Mitigation of Other Attacks

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

3 Secure Operation

The Blue Coat Systems, Software Cryptographic 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 Initial Setup

FIPS 140-2 mandates that a software cryptographic module at Security Level 1 be restricted to a single operator mode of operation. Solera DeepSee Software, by default configures the Solera Operating Environment for single-user mode.

The Software Cryptographic Module is installed as part of the installation of Solera DeepSee Software. The CO should follow the installation procedures found in the *Solera Networks DeepSee Administration Guide* Chapter 1: Installation and Configuration.

3.2 Secure Management

The following paragraphs describe the steps necessary to ensure that the Software Cryptographic Module is running in a FIPS-Approved manner.

3.2.1 Initialization

The Solera Software applications provided in Solera DeepSee Software v6.5.0 will ensure the module is configured in the FIPS-Approved mode prior use. An integrity check is performed for the module automatically when FIPS mode is set. This is achieved by calling a single initialization function *FIPS_mode_set()*. Upon initialization of the module, the module requires no set-up and runs its power-up self-tests automatically without operator interference. The power-up self-tests include a software integrity test that checks the integrity of the module using an HMAC SHA-1 digest. If the integrity check succeeds, then the module performs the remaining power-up self-tests. If the module passes all self-tests the function returns a value of "1", which indicates that the module is in a FIPS-Approved mode of operation. If the function returns "0" that indicates failure of the tests.

3.2.2 Management

The Crypto-Officer can call verify that the module is operating in a FIPS-Approved mode of operation with the *fips_mode* flag. Self-tests can be performed on demand by cycling the power on the host platform, or by the function call *FIPS_selftest()*.

3.2.3 Zeroization

The module does not persistently store any key or CSPs. All ephemeral keys used by the module are zeroized upon session termination. All keys can be zeroized by power cycling or rebooting the host platform.

3.2.4 User Guidance

The Software Cryptographic Module is designed for use by Solera DeepSee Software applications. The module does not input, output, or persistently store CSPs with respect to the physical boundary. As the module allows access to cryptographic services that are not FIPS-Approved or that provide less than the minimum NIST-recommended encryption strength, it is the responsibility of the calling application developer to ensure that only appropriate algorithms, key sizes, and key establishment techniques are applied. Users are responsible for using only the services that are listed in Table 4. Any use of the Blue Coat Systems, Software Cryptographic Module with non-FIPS-Approved cryptographic services or keys that provide less than 112 bits of encryption strength constitutes a departure from this Security Policy, and results in the module not being in its Approved mode of operation.

.

4 Acronyms

This section describes the acronyms.

Table 8 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
APT	Advanced Persistent Threats
ATA	Advanced Targeted Attacks
BIOS	Basic Input/Output System
CAST	Carlisle Adams and Stafford Tavares
CBC	Cipher Block Chaining
CFB	Cipher Feedback
CMVP	Cryptographic Module Validation Program
CO	Crypto-Officer
CPU	Central Processing Unit
CSEC	Communications Security Establishment Canada
CSP	Critical Security Parameter
DES	Data Encryption Standard
DH	Diffie-Hellman
DSA	Digital Signature Algorithm
DVD	Digital Video Disc
ECB	Electronic Codebook
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
GBPS	Gigabits per Second
GPC	General Purpose Computer
HDD	Hard Disk Drive
HMAC	Hash Message Authentication Code
IDEA	International Data Encryption Algorithm
IT	Information Technology
KAT	Known Answer Test

.

Acronym	Definition
KDF	Key Derivation Function
MD	Message Digest
MDC	Modification Detection Code
NDRNG	Non-Deterministic Random Number Generator
NIST	National Institute of Standards and Technology
OFB	Output Feedback
OS	Operating System
PCAP	Packet Capture
PCI	Peripheral Component Interconnect
PICe	PCI express
PKCS	Public-Key Cryptography Standard
PRNG	Psuedo Random Number Generator
PSS	Probabilistic Signature Scheme
RACE	Research and Development in Advanced Communications Technologies in Eurpoe
RAM	Random Access Memory
RC	Rivest Cipher
RipeMD	RACE Integrity Primitives Evaluation Message Digest
RNG	Random Number Generator
RSA	Rivest, Shamir, and Adleman
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Inteface
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
TDES	Triple Data Encryption Standard
TLS	Transport Layer Security
USB	Universal Serial Bus

Prepared by:
Corsec Security, Inc.

13135 Lee Jackson Memorial Highway
Suite 220
Fairfax, VA 22033

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