



IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Security Target

Version:	1.0
Status:	Final
Last Update:	2016-02-25

Trademarks

The following terms are trademarks and/or registered trademarks of atsec information security corporation in the United States and other countries:

- atsec®

The IBM logo is a trademark of International Business Machines Corporation in the United States and other countries.

The following terms are trademarks and/or registered trademarks of International Business Machines Corporation in the United States and other countries:

- BigFix®
- DB2®
- Fixlet®
- IBM®
- Relevance®

The following terms are trademarks and/or registered trademarks of Intel Corporation in the United States and other countries:

- Intel®
- Xeon®

The following terms are trademarks and/or registered trademarks of Linus Torvalds in the United States and other countries:

- Linux®

The following terms are trademarks and/or registered trademarks of Microsoft Corporation in the United States and other countries:

- Active Directory®
- Windows®

The following terms are trademarks and/or registered trademarks of Red Hat, Inc. in the United States and other countries:

- Red Hat®

Legal Notice

This document is provided AS IS with no express or implied warranties. Use the information in this document at your own risk.

This document may be reproduced or distributed in any form without prior permission provided the copyright notice is retained on all copies. Modified versions of this document may be freely distributed provided that they are clearly identified as such, and this copyright is included intact.

Revision History

Revision	Date	Author(s)	Changes to Previous Revision
1.0	2016-02-25	Scott Chapman	Final ST.

Table of Contents

1	Introduction	9
1.1	Security Target Identification	9
1.2	TOE Identification	9
1.3	TOE Type	9
1.4	TOE Overview	9
1.4.1	Required and optional non-TOE hardware and software	10
1.4.1.1	Air gapped network Operational Environment	10
1.4.1.2	Internet accessible network Operational Environment	10
1.4.2	Intended method of use	11
1.4.3	Major security features	11
1.5	TOE Description	11
1.5.1	TOE architecture	13
1.5.2	TOE security features	16
1.5.2.1	Security auditing	16
1.5.2.2	Cryptographic support	16
1.5.2.3	User data protection	16
1.5.2.4	Identification and authentication	16
1.5.2.5	Security management	17
1.5.2.6	Protection of the TSF	17
1.5.2.7	TOE access	18
1.5.2.8	Trusted path/channels	18
1.5.3	Physical boundary and delivery	18
1.5.4	Evaluated configuration	19
1.5.5	Operational Environment	19
2	CC Conformance Claim	20
2.1	Protection Profile tailoring and additions	20
3	Security Problem Definition	21
3.1	Threat Environment	21
3.1.1	Threats countered by the TOE	21
3.2	Assumptions	22
3.2.1	Intended usage of the TOE	22
3.3	Organizational Security Policies	22
4	Security Objectives	23
4.1	Objectives for the TOE	23
4.2	Objectives for the Operational Environment	23
4.3	Security Objectives Rationale	23
5	Extended Components Definition	24
6	Security Requirements	25
6.1	TOE Security Functional Requirements	25
6.1.1	Security audit (FAU)	27
6.1.1.1	Audit Data Generation (FAU_GEN.1)	27

6.1.1.2	User Identity Association (FAU_GEN.2)	29
6.1.1.3	External Audit Trail Storage (FAU_STG_EXT.1)	29
6.1.2	Cryptographic support (FCS)	29
6.1.2.1	Cryptographic Key Generation (for asymmetric keys) (FCS_CKM.1)	29
6.1.2.2	Cryptographic Key Zeroization (FCS_CKM_EXT.4)	29
6.1.2.3	Cryptographic Operation (for data encryption/decryption) (FCS_COP.1(1))	29
6.1.2.4	Cryptographic Operation (for cryptographic signature) (FCS_COP.1(2))	29
6.1.2.5	Cryptographic Operation (for cryptographic hashing) (FCS_COP.1(3))	30
6.1.2.6	Cryptographic Operation (for keyed-hash message authentication) (FCS_COP.1(4))	30
6.1.2.7	Extended: Cryptographic Operation (Random Bit Generation) (FCS_RBG_EXT.1)	30
6.1.2.8	Explicit: TLS (FCS_TLS_EXT.1)	30
6.1.2.9	Explicit: HTTPS (FCS_HTTPS_EXT.1)	30
6.1.3	User data protection (FDP)	31
6.1.3.1	Full Residual Information Protection (FDP_RIP.2)	31
6.1.4	Identification and authentication (FIA)	31
6.1.4.1	Password Management (FIA_PMG_EXT.1)	31
6.1.4.2	User Identification and Authentication (FIA_UIA_EXT.1)	31
6.1.4.3	Extended: Password-based Authentication Mechanism (FIA_UAU_EXT.2)	32
6.1.4.4	Protected Authentication Feedback (FIA_UAU.7)	32
6.1.5	Security management (FMT)	32
6.1.5.1	Management of TSF Data (for general TSF data) (FMT_MTD.1)	32
6.1.5.2	Specification of Management Functions (FMT_SMF.1)	32
6.1.5.3	Restrictions on Security Roles (FMT_SMR.2)	32
6.1.6	Protection of the TSF (FPT)	33
6.1.6.1	Extended: Protection of TSF Data (for reading of all symmetric keys) (FPT_SKP_EXT.1)	33
6.1.6.2	Extended: Protection of Administrator Passwords (FPT_APW_EXT.1)	33
6.1.6.3	Reliable Time Stamps (FPT_STM.1)	33
6.1.6.4	Extended: Trusted Update (FPT_TUD_EXT.1)	33
6.1.6.5	TSF Testing (FPT_TST_EXT.1)	33
6.1.7	TOE access (FTA)	34
6.1.7.1	TSF-initiated Session Locking (FTA_SSL_EXT.1)	34
6.1.7.2	TSF-initiated Termination (FTA_SSL.3)	34
6.1.7.3	User-initiated Termination (FTA_SSL.4)	34
6.1.7.4	Default TOE Access Banners (FTA_TAB.1)	34
6.1.8	Trusted path/channels (FTP)	34
6.1.8.1	Inter-TSF trusted channel (FTP_ITC.1)	34
6.1.8.2	Trusted Path (FTP_TRP.1)	34
6.2	Security Functional Requirements Rationale	35
6.2.1	Security requirements coverage	35

6.2.2	Security requirements dependency analysis	36
6.3	Security Assurance Requirements	38
6.3.1	Development (ADV)	38
6.3.1.1	Basic Functional Specification (ADV_FSP.1)	38
6.3.2	Guidance documents (AGD)	39
6.3.2.1	Operation User Guidance (AGD_OPE.1)	39
6.3.2.2	Preparative Procedures (AGD_PRE.1)	40
6.3.3	Tests (ATE)	40
6.3.3.1	Independent Testing - Conformance (ATE_IND.1)	40
6.3.4	Vulnerability assessment (AVA)	41
6.3.4.1	Vulnerability Survey (AVA_VAN.1)	41
6.3.5	Life-cycle support (ALC)	41
6.3.5.1	Labeling of the TOE (ALC_CMC.1)	41
6.3.5.2	TOE CM Coverage (ALC_CMS.1)	41
6.4	Security Assurance Requirements Rationale	42
7	TOE Summary Specification	43
7.1	TOE Security Functionality	43
7.1.1	Security auditing	43
7.1.1.1	Audit data generation	43
7.1.1.2	Audit trail storage	44
7.1.2	Cryptographic support	45
7.1.2.1	Cryptographic algorithms	45
7.1.2.2	Random bit generation	47
7.1.3	User data protection	48
7.1.4	Identification and authentication	48
7.1.5	Security management	49
7.1.6	Protection of the TSF	49
7.1.6.1	Prevent reading of all symmetric keys	49
7.1.6.2	Protection of administrator passwords	50
7.1.6.3	Reliable time stamps	50
7.1.6.4	Trusted update	50
7.1.6.5	Self-tests	52
7.1.7	TOE access	52
7.1.8	Trusted path/channels	53
8	Abbreviations, Terminology and References	54
8.1	Abbreviations	54
8.2	Terminology	56
8.3	References	57

List of Tables

Table 1: Air gapped network Operational Environment components	10
Table 2: Internet accessible network Operational Environment components	10
Table 3: SFRs for the TOE	25
Table 4: TOE security functional requirements and auditable events mapping	27
Table 5: Administrative password special characters	31
Table 6: Mapping of security functional requirements to security objectives	35
Table 7: TOE SFR dependency analysis	36
Table 8: SARs	38
Table 9: TOE audit records	43
Table 10: TOE key destruction	46
Table 11: TOE cryptographic algorithms and CAVP validation numbers	46
Table 12: Trusted path/channel connections	53

List of Figures

Figure 1: IBEM air gapped deployment	12
Figure 2: IBEM Fixlet management	13
Figure 3: TOE components	14

1 Introduction

1.1 Security Target Identification

Title: IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Security Target
Version: 1.0
Status: Final
Date: 2016-02-25
Sponsor: International Business Machines, Corporation
Developer: International Business Machines, Corporation
Validation Body: NIAP
Validation ID: VID10682
Keywords: NDPP, Common Criteria, BigFix Endpoint Manager, IBM

1.2 TOE Identification

The TOE is the IBM BigFix Endpoint Manager (IBEM) Common Criteria TOE Release 9.2.

1.3 TOE Type

The TOE type is an enterprise management network device for managing updates to networked computer systems (a.k.a. endpoints).

1.4 TOE Overview

The TOE is a single network server device comprised of hardware and all of the software components (applications and operating system) that reside on the hardware. The hardware model is:

- IBM System x3500 M5 server along with a monitor, keyboard, and mouse

The TOE software is comprised of the following components:

- IBEM¹ 9.2.3.101 Server software component (Linux only)
- IBEM 9.2.3.101 Client software component² (Linux only)
- Red Hat Enterprise Linux (RHEL) 6.6 Server (x86-64)

Additional details about the hardware model, operating system, and IBEM software components are specified in section 1.5.3.

The TOE is a content-driven messaging and management system that distributes the work of managing IT infrastructures out to managed devices (a.k.a. endpoints). The TOE and its managed endpoints reside on an air gapped network. The TOE utilizes a patented Fixlet® technology to identify vulnerable or misconfigured endpoints and allows authorized users to remediate identified issues across the network.

This Security Target (ST) is based on the "Protection Profile for Network Devices" ([NDPP][1](#)).

¹ The IBEM product has undergone multiple name changes. Previously, the product was named IBM Endpoint Manager (IEM). Prior to that, the product was named BigFix Enterprise Suite (BES).

² Although the IBEM Client software is supported by IBM on several platforms, only the Linux IBEM Client running inside the TOE was part of the evaluation. IBEM Clients running outside of the TOE were not part of the evaluation.

1.4.1 Required and optional non-TOE hardware and software

The TOE and a portion of its Operational Environment reside on an air gapped network. The other portion of the Operational Environment resides on an Internet accessible network. This section lists the Operational Environment components, both required and optional, for the two networks.

The IBEM product line contains software components that are *not* part of this evaluation, but they can be used in the Operational Environment with the TOE. These non-evaluated software components are:

- IBEM Console software component
- IBEM Relay software component
- IBEM Client software component installed on any device other than the TOE

1.4.1.1 Air gapped network Operational Environment

The items in Table 1 are part of the Operational Environment for the air gapped network and are either required or optional as indicated in the table. All items listed in Table 1 must reside on systems other than the TOE.

Required Qty	Description of air gapped network Operational Environment components
1 or more	IBEM Console software component
0 or more	IBEM Relay software component
0 or more	IBEM Client software component (installed on systems other than the TOE)
1	IBM DB2 database (the database application on the database server)
1	Syslog server (for storing audit records remotely)

Table 1: Air gapped network Operational Environment components

1.4.1.2 Internet accessible network Operational Environment

Fixlets can be downloaded from an IBM BigFix Fixlet Server over the Internet and then imported by the TOE. (Fixlets are described in more detail in section 1.5.) Since the TOE is on an air gapped network with no Internet access, a separate Microsoft Windows system with Internet access and with the IBEM Airgap tool (a.k.a. Windows BigFix Airgap utility, *BESAirgapTool.exe*) installed on the system is required to perform this download. Once the Fixlets are downloaded to this Microsoft Windows system, an administrator using portable media like a USB drive transfers the Fixlets to the TOE.

Required Qty	Description of Internet accessible network Operational Environment components
1	IBEM Airgap tool running on a Microsoft Windows system that has access to the Internet

Table 2: Internet accessible network Operational Environment components

1.4.2 Intended method of use

The TOE is designed to be used inside of an organization air gapped from the Internet (e.g., not connected to the Internet) in a non-hostile environment. The TOE must be located in a protected environment (e.g., server room) where only trusted administrators have access to the physical computers. No other software can reside on the TOE.

1.4.3 Major security features

As per the [NDPP][\[1\]](#), the TOE supports the following major security features:

- Security auditing
- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF (TOE security functionality)
- TOE access
- Trusted path/channels

1.5 TOE Description

The TOE is a device that identifies vulnerable or misconfigured endpoints and allows authorized users to remediate identified issues across the network. Figure 1 depicts an air gapped network (i.e., a network that is *not* connected to the Internet) deployment of the TOE.

The TOE resides on the air gapped network and communicates with other systems on the network to perform its tasks. In the deployment shown in Figure 1, the TOE is connected to the following devices in the Operational Environment:

- **Audit Server (syslog)**—Used by the TOE to remotely store the TOE's audit records.
- **Database Server**—Used by the TOE to store certain TSF data.
- **Endpoints**—Servers, PCs, laptops, etc. managed by the TOE.
- **Relays**—Endpoint aggregators used by the TOE to offload direct communications with endpoints. Relays are also endpoints.
- **Remote Consoles**—Remote computers used by administrators to manage the TOE and endpoints.

Basic IBEM Air Gapped Network Deployment

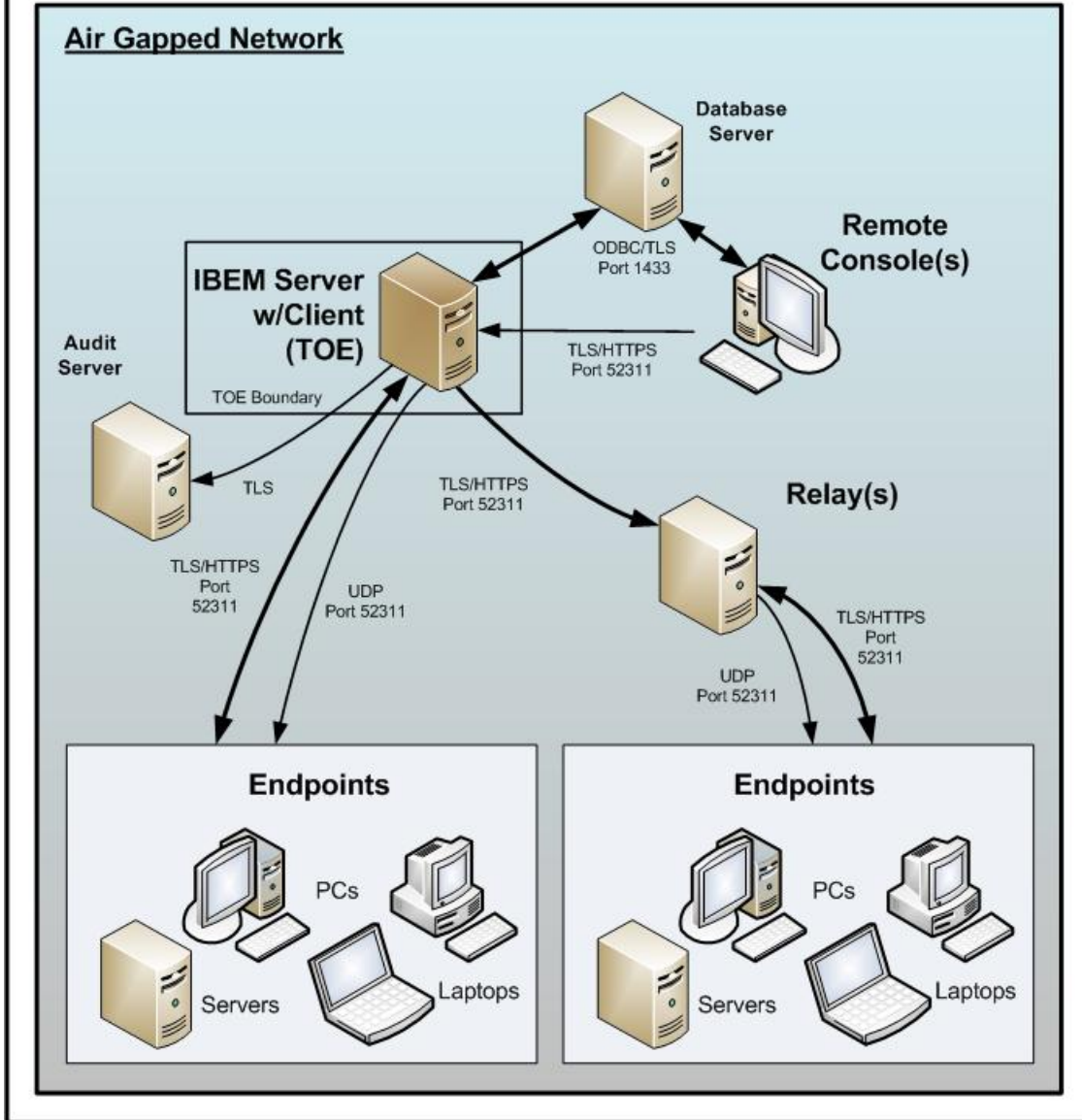


Figure 1: IBEM air gapped deployment

The TOE utilizes a patented Fixlet® technology to identify vulnerable or misconfigured endpoints and allows authorized users to remediate identified issues across the air gapped network. Fixlets are sent via Fixlet messages to the endpoints by the TOE.

Fixlets are available to administrators by subscribing to any of a number of Internet-based BigFix Fixlet Server Fixlet sites maintained by IBM. Each Fixlet site contains pretested, prepackaged Fixlet messages that provide out-of-the-box management solutions.

Since the TOE resides on an air gapped network, a separate Internet accessible network is required to obtain Fixlets from an IBM BigFix Fixlet Server. IBEM supplies an Airgap tool that runs on a Microsoft Windows OS and allows an administrator to download Fixlets from an IBM BigFix Fixlet Server and then transfer the Fixlets to the TOE using portable media such as a USB drive for the TOE to import and validate the Fixlets. Figure 2 depicts this configuration with both the Internet accessible network and the air gapped network.

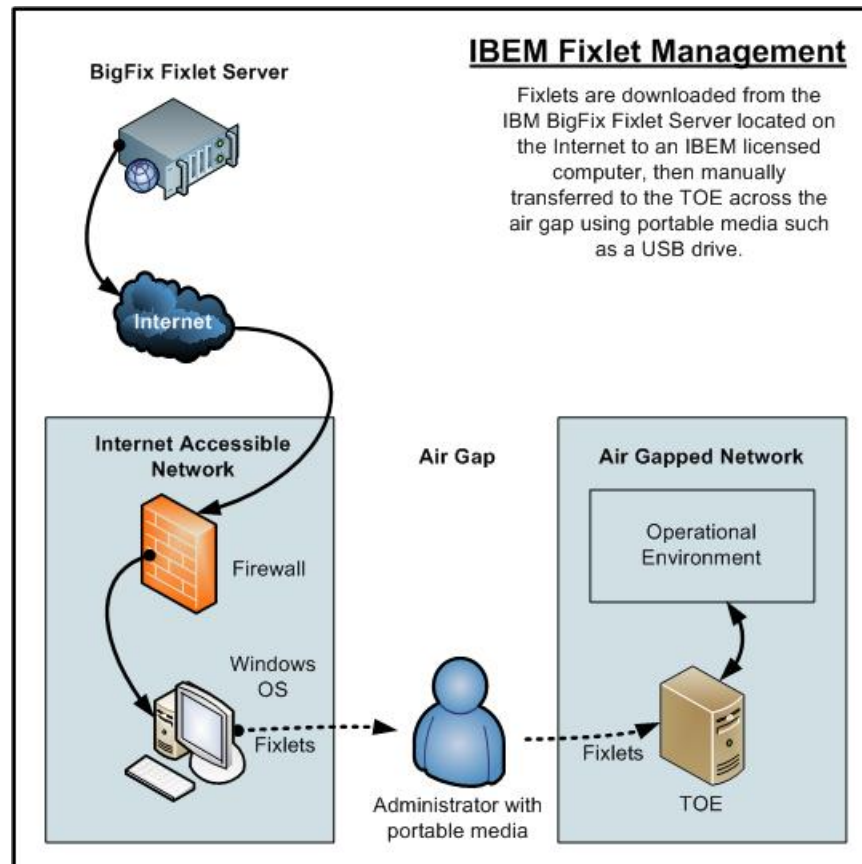


Figure 2: IBEM Fixlet management

Fixlet messages can also be developed in-house by administrators to address policy, configuration, and vulnerability concerns specific to the customer's environment. In-house fixes are known as Custom Fixlets and are developed by an authorized administrator to address specific situations. Both Fixlets and Custom Fixlets are supported by the TOE.

1.5.1 TOE architecture

The hardware for the TOE is an IBM System x3500 M5 server (Part Number 5464-AC1) containing:

- One Intel Xeon E5-2637 v3 series processor
- Quad-port Gigabit Ethernet

- One Gigabit management port
- One IBM 1.2TB 10K 6Gbps SAS 2.5" G3HS HDD
- ServeRAID M5210 SAS/SATA Controller for IBM System x
- 8GB High-speed 2133 MHz DDR4 SDRAM Registered DIMMs
- IBM HH DVD-ROM
- Seven Peripheral Component Interconnect Express (PCIe) 3.0 slots
- Two internal USB ports
- Six external USB ports (two front, four rear)
- One DB-15 VGA port

In addition, the following hardware components are required for the TOE:

- Keyboard
- Monitor
- Mouse

Figure 3 shows a layering of the software components on the TOE. The TOE includes and runs the RHEL 6.6 OS. The OS includes the RHEL kernel, a built-in firewall (not shown), and a set of RHEL applications such as the *stunnel* command.

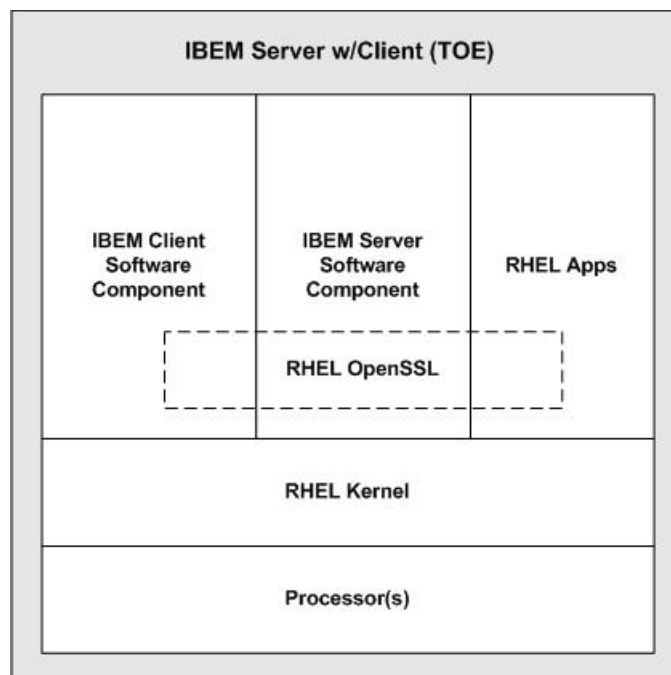


Figure 3: TOE components

The OS contains the RHEL OpenSSL package which includes a cryptographic library and Transport Layer Security (TLS) implementation. This package is used by the TOE in FIPS-enabled mode to protect the integrity and confidentiality of all trusted channel and trusted path communications.

The TOE also contains the IBEM Server software component and the IBEM Client software component. The IBEM Server software component manages and coordinates the flow of update information to and from the endpoints and stores the results on the database server. The IBEM Client software

component monitors the TOE for potential updates that apply to itself (i.e., the TOE is its own endpoint). The communication between the IBEM Server software component and IBEM Client software component is performed inside the TOE, so the communications is physically protected by the TOE.

The TOE supports the following local console for managing the TOE:

- Local OS console

The RHEL OS controls the local OS console interface. This interface is available to administrators from the attached monitor, keyboard, and mouse. The local OS console enforces the identification and authentication (I&A) of administrative users and allows the administrator to manage certain aspects of the TOE. The I&A for the local OS console is performed by the OS.

The TOE supports the following remote console type for managing the TOE:

- IBEM Console (remote GUI)

The TOE enforces the use of TLS/HTTPS (Hypertext Transfer Protocol Secure) to protect the communications channel of the IBEM Console. The TOE also enforces I&A of all remote console users through the use of user names and passwords. The account information for the IBEM Console is managed by the IBEM Server software component and is located remotely on the database server. Multiple IBEM Consoles can connect to the TOE simultaneously.

The TOE uses Open Database Connectivity (ODBC) to communicate with the database server. The network connection between the TOE and the database server is protected by the *stunnel* command which uses the TLS 1.0 protocol contained in the RHEL OpenSSL package. The TOE uses the database server to store and retrieve applicable Fixlets, TOE configuration data, EM administrative user account data, and other TOE data.

The TOE uses rsyslog to communicate to the audit server. The network connection between the TOE and the audit server is protected by the *stunnel* command which uses the TLS 1.0 protocol contained in the RHEL OpenSSL package.

In the evaluated configuration, Fixlets are downloaded by an administrator from an IBM BigFix Fixlet Server using a computer on an Internet accessible network. The administrator transfers and imports the downloaded Fixlets to the TOE residing on an air gapped network. Those Fixlets, as well as any Custom Fixlets, are stored on the database server and broadcast to each endpoint periodically by the TOE.

The TOE listens on a Transport Control Protocol (TCP) port for TLS/HTTPS messages from endpoints, relays, and IBEM Consoles. Data files containing Fixlets, Actions, or responses to Actions performed on endpoints are communicated between the TOE and endpoints using TLS/HTTPS protected messages (Fixlet messages). For the most part, the TOE just responds to TLS/HTTPS requests. The exception is that the TOE can issue User Datagram Protocol (UDP) messages to relays and endpoints when new content (e.g., a Fixlet) becomes available. Relays and endpoints are configured to periodically poll the TOE for new content, but if they receive a UDP message, they will retrieve the updated content from the TOE outside of their normal polling cycle.

1.5.2 TOE security features

1.5.2.1 Security auditing

The TOE generates audit records for the audit events specified by the [NDPP][\[4\]](#). The TOE stores its audit records in a local cache file on the TOE. When the cache file becomes full, the oldest records are overwritten by the newest records. The TOE also sends the audit records over a TLS-protected connection to an audit server located in the Operational Environment. This audit server serves as the external IT entity for storing audit records as required by the [NDPP][\[4\]](#).

Additional detail on the TOE's auditing functionality is provided in [section 7.1.1](#).

1.5.2.2 Cryptographic support

The TOE contains the following OpenSSL package for cryptography and TLS:

- RHEL OpenSSL package

The TOE uses the RHEL OpenSSL package in the Federal Information Processing Standards FIPS-enabled mode for TLS and TLS/HTTPS communications between itself and other non-TOE systems (i.e., endpoints, relays, remote consoles, the database server, the audit server), except for UDP communications.

The TOE uses Rivest-Shamir-Adleman (RSA) asymmetric key pairs for key establishment in the TLS and TLS/HTTPS protocols. The connections between the TOE and the database server and between the TOE and the audit server support TLS 1.0 in the evaluated configuration. All other TLS connections support TLS 1.0, TLS 1.1, and TLS 1.2 in the evaluated configuration.

The TLS and TLS/HTTPS protocols use the RSA algorithm with key sizes of 2048-bits or greater. These protocols use the Advanced Encryption Standard (AES) 128-bits and 256-bits cryptographic algorithms in cipher block chaining (CBC) mode for symmetric cryptography. These protocols also use the secure hash algorithms (SHAs) of SHA-1 and SHA-256 for cryptographic hashing and keyed-hash message authentication.

The TOE also supports several other block cipher modes of chaining for AES. A list of the modes is in [section 7.1.2](#). In addition, the TOE supports SHA-224, SHA-256, SHA-384, and SHA-512 for cryptographic hashing and keyed-hash message authentication.

The TOE overwrites all secret keys, private cryptographic keys, and critical security parameters once they are no longer required.

The RHEL OpenSSL package in FIPS-enabled mode contains a deterministic random bit generator (DRBG) that uses CTR_DRBG (AES). This DRBG is used by the TOE to fulfill various cryptographic keying requirements. For entropy, the DRBG uses a software-based noise source.

Additional details on the TOE's random bit generation and other cryptographic functions are provided in [section 7.1.2](#).

1.5.2.3 User data protection

The TOE ensures that packets sent through the TOE are made unavailable. The TOE does not provide pass-through capabilities for network packets.

1.5.2.4 Identification and authentication

The TOE's administrative interfaces are:

- IBEM Console (remote GUI)
- Local OS console (RHEL)

All administrative interfaces require each user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that user.

All administrative interfaces support password lengths of 15 characters or greater. For all administrative interfaces, the password composition includes upper case letters, lower case letters, numbers, and special characters. These interfaces provide obscured feedback of authentication data to the administrative user during the login process.

The authentication mechanisms for all administrative interfaces are local to the TOE. The TOE does not use remote authentication mechanisms such as LDAP (Lightweight Directory Access Protocol), Microsoft Active Directory, etc. The administrator account data used by the IBEM Server software component is stored on the database server. When an administrator logs in to the TOE from the IBEM Console interface, the IBEM Server software component retrieves the administrator's I&A data from the database server and performs the validation of the administrator's I&A data. When an administrator logs in to the TOE from the local OS console interface, the operating system identifies and authenticates the user using the operating system's local I&A files.

1.5.2.5 Security management

The TOE allows an authorized administrator to manage the TOE locally via the local OS console and remotely via the IBEM Console. Only authorized administrators are allowed to use these interfaces and manage the TSF data of the TOE.

The TOE provides security management of its trusted update feature via the IBEM Console. For additional information, see [section 1.5.2.6](#).

The TOE can associate a user with a single role. Note though that the IBEM Server software component uses a separate user authentication database than the RHEL operating system for defining user accounts and associating roles. The guidance documentation describes how to properly manage user accounts and roles between the IBEM Server software component and the operating system.


1.5.2.6 Protection of the TSF

The TOE conceals all symmetric keys and private keys from being viewed by administrators during normal usage.

The TOE hashes all administrator passwords before storing them so that no plaintext passwords are stored on the system.

The TOE includes a reliable time stamp mechanism for supporting the various security needs of the TOE (e.g., audit record time stamps).

For TSF testing, the TOE performs cryptographic module self-tests during its startup process.

For the trusted update process required by the [NDPP], the updates for the TOE are controlled by the TOE. One of the main functions of the TOE is to distribute updates to endpoints. The TOE is also an endpoint because of the existence of the IBEM Client software component on the TOE. The application of TOE updates is controlled by an administrator from the IBEM Console. The hash values for each update image are digitally signed by IBM with an X.509v3 certificate. This signature is verified by the TOE prior to initiating the update process from the IBEM Console. When an update

is applied by the TOE, the TOE first validates the hash values of each update image from the list of signed hash values, then applies the update images. If the hash values do not match, the update is not applied. Additional details are provided in section 7.1.6.4.

1.5.2.7 TOE access

The TOE provides an administrator-specified advisory notice and consent warning message on all interactive user interfaces prior to establishing an administrative user session. The TOE also allows administrators to terminate their own interactive sessions.

The TOE enforces session locking of all local interactive sessions after an administrator-specified period of inactivity.

The TOE enforces session termination of all remote interactive sessions after an administrator-specified period of inactivity.

1.5.2.8 Trusted path/channels

The TOE uses trusted communication channels between itself and other trusted IT products. The TOE initiates a connection to the database server which is protected using TLS. In addition, the TOE initiates a connection to the audit server (syslog server) which is protected using TLS.

The TOE uses trusted path communications between itself and the IBEM Console. The IBEM Console initiates and uses TLS/HTTPS to communicate to the TOE.

1.5.3 Physical boundary and delivery

IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 is comprised of the following physical software items:

- TOE:
 - ISO image of the TOE (rhel1) - This image contains the IBEM Server 9.2.3.101, IBEM Client 9.2.3.101, and RHEL 6.6.
 - Server installation binary (ServerInstaller_9.2.3.101-rhe6.x86_64.tgz) - This binary contains the IBEM Server 9.2.3.101 patch for the TOE.
 - RHEL OpenSSL RPM (openssl-1.0.1e-42.el6_7.2.x86_64.rpm) - This RPM Package Manager (RPM) file contains the RHEL OpenSSL patch for the TOE.
 - BES Support QA Masthead (BES Support QA.efxm) - This is the masthead data file for the TOE.
 - IBM BigFix 9.2 Common Criteria Guide V1.9
- Operational Environment:
 - ISO image 1 of 2 of the Operational Environment (rhel2) - This image contains the audit server, database server, and RHEL 6.6.
 - ISO image 2 of 2 of the Operational Environment (win7box) - This image contains the IBEM Console and Microsoft Windows OS.
 - Console binary (BESConsole.exe) - This binary is an updated IBEM Console binary for the win7box ISO image.

To obtain the above items, contact an IBM U.S. Federal Sales representative.

The hardware is obtainable from IBM:

- IBM System x3500 M5 Server

- Server component configuration is specified in [section 1.5.1](#)
 - Monitor
 - Keyboard
 - Mouse

The TOE includes the following guidance documents that are independently downloadable from the IBM website:

- "IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Action Guide"
- "IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 API Reference Guide"
- "IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Asset Discovery User's Guide"
- "IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Console Operator's Guide"
- "IBM BigFix Endpoint Manager Common Criteria TOE Release 9.2 Relevance Language Guide"

1.5.4 Evaluated configuration

The evaluated configuration consists of the software, hardware, and guidance documentation specified in [section 1.5.3](#). The evaluated configuration also imposes some limitations on the configuration of the product.

The specifications for configuring the TOE in the evaluated configuration are located in the guidance documentation listed in [section 1.5.3](#). The consumer must read, understand, and follow the guidance documentation provided as part of the TOE for the evaluated configuration.

The following restrictions apply to the evaluated configuration:

- The IBEM Server software component must be configured as an authenticating server.
- The communication encryption methods must be configured as per the "IBM BigFix 9.2 Common Criteria Guide V1.9."
- The database server is expected to be configured so that its ODBC interface and communications are protected in a manner appropriate to the environment in which it is being used. This includes the connection between the Server device and the database as well as the connection between the Console and the database server.
- Each user account can have only one role assigned to it.
- FTP must be disabled.
- SSH must be disabled.
- The Web Reports interface must be disabled.

1.5.5 Operational Environment

The Operational Environment for the TOE consists of the components specified in [section 1.4.1](#).

2 CC Conformance Claim

This Security Target is CC Part 2 extended and CC Part 3 conformant.

This Security Target claims conformance to the following Protection Profiles and PP packages:

- [\[NDPP\]](#): Protection Profile for Network Devices. Version 1.1 as of 2012-06-08; exact conformance.

Common Criteria [\[CC\]](#) version 3.1 revision 4 is the basis for this conformance claim.

2.1 Protection Profile tailoring and additions

This Security Target includes the "Security Requirements for Network Devices Errata #3" ([\[NDPP-ERR3\]](#)). In addition, this Security Target follows the NIAP definition of "exact compliance."

3 Security Problem Definition

3.1 Threat Environment

This section describes the threat model for the TOE and identifies the individual threats that are assumed to exist in the TOE environment.

The **assets** to be protected by the TOE are information or resources to be protected by the countermeasures of the TOE.

The **threat agents** having an interest in manipulating the data model can be categorized as either:

- Unauthorized individuals ("attackers") which are unknown to the TOE and its runtime environment
- Authorized users of the TOE (i.e., administrators) who try to manipulate data that they are not authorized to access

Threat agents originate from a well-managed user community within an organizations internal network. Hence, only inadvertent or casual attempts to breach system security are expected from this community.

TOE administrators, including administrators of the TOE environment, are assumed to be trustworthy, trained and to follow the instructions provided to them with respect to the secure configuration and operation of the systems under their responsibility. Hence, only inadvertent attempts to manipulate the safe operation of the TOE are expected from this community.

3.1.1 Threats countered by the TOE

T.ADMIN_ERROR

An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.

T.TSF_FAILURE

Security mechanisms of the TOE may fail, leading to a compromise of the TSF.

T.UNDETECTED_ACTIONS

Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.

T.UNAUTHORIZED_ACCESS

A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.

T.UNAUTHORIZED_UPDATE

A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.

T.USER_DATA_REUSE

User data may be inadvertently sent to a destination not intended by the original sender.

3.2 Assumptions

3.2.1 Intended usage of the TOE

A.NO_GENERAL_PURPOSE

It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

A.PHYSICAL

Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.

A.TRUSTED_ADMIN

TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

3.3 Organizational Security Policies

P.ACCESS_BANNER

The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

4 Security Objectives

4.1 Objectives for the TOE

O.PROTECTED_COMMUNICATIONS

The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.

O.VERIFIABLE_UPDATES

The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.

O.SYSTEM_MONITORING

The TOE will provide the capability to generate audit data and send those data to an external IT entity.

O.DISPLAY_BANNER

The TOE will display an advisory warning regarding use of the TOE.

O.TOE_ADMINISTRATION

The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.

O.RESIDUAL_INFORMATION_CLEARING

The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.

O.SESSION_LOCK

The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.

O.TSF_SELF_TEST

The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

4.2 Objectives for the Operational Environment

OE.NO_GENERAL_PURPOSE

There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

OE.PHYSICAL

Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

OE.TRUSTED_ADMIN

TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

4.3 Security Objectives Rationale

The security objectives rationale is provided in section 3 of the [NDPP][a](#).

5 Extended Components Definition

The Security Target draws upon the extended components implicitly defined in the [NDPP][\[4\]](#).

6 Security Requirements

6.1 TOE Security Functional Requirements

The operations performed on the security functional requirements in this Security Target are based on the security functional requirements as specified in the [NDPP].

The following table shows the SFRs for the TOE, and the operations performed on the components according to CC part 2: iteration (Iter.), refinement (Ref.), assignment (Ass.) and selection (Sel.).

Security functional group	Security functional requirement	Base security functional component	Source	Operations			
				Iter.	Ref.	Ass.	Sel.
FAU - Security audit	FAU_GEN.1 Audit Data Generation		NDPP	No	No	No	No
	FAU_GEN.2 User Identity Association		NDPP	No	No	No	No
	FAU_STG_EXT.1 External Audit Trail Storage		NDPP	No	No	No	Yes
FCS - Cryptographic support	FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)		NDPP	No	Yes	No	Yes
	FCS_CKM_EXT.4 Cryptographic Key Zeroization		NDPP	No	No	No	No
	FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)	FCS_COP.1	NDPP	Yes	No	Yes	Yes
	FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)	FCS_COP.1	NDPP	Yes	Yes	No	Yes
	FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)	FCS_COP.1	NDPP	Yes	No	No	Yes
	FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)	FCS_COP.1	NDPP	Yes	No	Yes	Yes
	FCS_RBG_EXT.1 Extended: Cryptographic Operation (Random Bit Generation)		NDPP	No	No	No	Yes
	FCS_TLS_EXT.1 Explicit: TLS		NDPP	No	No	No	Yes
	FCS_HTTPS_EXT.1 Explicit: HTTPS		NDPP	No	No	No	No
FDP - User data protection	FDP_RIP.2 Full Residual Information Protection		NDPP	No	No	No	Yes

Security functional group	Security functional requirement	Base security functional component	Source	Operations			
				Iter.	Ref.	Ass.	Sel.
FIA - Identification and authentication	FIA_PMG_EXT.1 Password Management		NDPP	No	Yes	Yes	Yes
	FIA_UIA_EXT.1 User Identification and Authentication		NDPP	No	No	No	Yes
	FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism		NDPP	No	No	No	Yes
	FIA_UAU.7 Protected Authentication Feedback		NDPP	No	No	No	No
FMT - Security management	FMT_MTD.1 Management of TSF Data (for general TSF data)		NDPP	No	No	No	No
	FMT_SMF.1 Specification of Management Functions		NDPP	No	No	No	Yes
	FMT_SMR.2 Restrictions on Security Roles		NDPP	No	No	No	No
FPT - Protection of the TSF	FPT_SKP_EXT.1 Extended: Protection of TSF Data (for reading of all symmetric keys)		NDPP	No	No	No	No
	FPT_APW_EXT.1 Extended: Protection of Administrator Passwords		NDPP	No	No	No	No
	FPT_STM.1 Reliable Time Stamps		NDPP	No	No	No	No
	FPT_TUD_EXT.1 Extended: Trusted Update		NDPP	No	No	No	Yes
	FPT_TST_EXT.1 TSF Testing		NDPP	No	No	No	No
FTA - TOE access	FTA_SSL_EXT.1 TSF-initiated Session Locking		NDPP	No	No	No	Yes
	FTA_SSL.3 TSF-initiated Termination		NDPP	No	No	No	No
	FTA_SSL.4 User-initiated Termination		NDPP	No	No	No	No
	FTA_TAB.1 Default TOE Access Banners		NDPP	No	No	No	No
FTP - Trusted path/channels	FTP_ITC.1 Inter-TSF trusted channel		NDPP	No	No	Yes	Yes

Security functional group	Security functional requirement	Base security functional component	Source	Operations			
				Iter.	Ref.	Ass.	Sel.
	FTP_TRP.1 Trusted Path		NDPP	No	No	No	Yes

Table 3: SFRs for the TOE

6.1.1 Security audit (FAU)

6.1.1.1 Audit Data Generation (FAU_GEN.1)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shut-down of the audit functions;
- b) All auditable events for the not specified level of audit; and
- c) All administrative actions;
- d) Specifically defined auditable events listed in Table 4.

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of Table 4.

TOE SFRs and auditable events mapping		
Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	
FAU_GEN.2	None.	
FAU_STG_EXT.1	None.	
FCS_CKM.1	None.	
FCS_CKM_EXT.4	None.	
FCS_COP.1(1)	None.	
FCS_COP.1(2)	None.	
FCS_COP.1(3)	None.	
FCS_COP.1(4)	None.	
FCS_RBG_EXT.1	None.	
FDP_RIP.2	None.	
FIA_PMG_EXT.1	None.	

TOE SFRs and auditable events mapping		
Requirement	Auditable Events	Additional Audit Record Contents
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	
FMT_MTD.1	None.	
FMT_SMF.1	None.	
FMT_SMR.2	None.	
FPT_SKP_EXT.1	None.	
FPT_APW_EXT.1	None.	
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FPT_TST_EXT.1	None.	
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.
FTA_TAB.1	None.	
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted channel functions.	Identification of the claimed user identity.
FCS_TLS_EXT.1	Failure to establish a TLS session.	Reason for failure.
	Establishment/Termination of a TLS session.	Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session.	Reason for failure.
	Establishment/Termination of a HTTPS session.	Non-TOE endpoint of connection (IP address) for both successes and failures.

Table 4: TOE security functional requirements and auditable events mapping

6.1.1.2 User Identity Association (FAU_GEN.2)

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

6.1.1.3 External Audit Trail Storage (FAU_STG_EXT.1)

FAU_STG_EXT.1.1 The TSF shall be able to **transmit the generated audit data to an external IT entity** using a trusted channel implementing the **TLS** protocol.

6.1.2 Cryptographic support (FCS)

6.1.2.1 Cryptographic Key Generation (for asymmetric keys) (FCS_CKM.1)

FCS_CKM.1.1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with

- a) **NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes [NIST800-56B]**

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

6.1.2.2 Cryptographic Key Zeroization (FCS_CKM_EXT.4)

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

6.1.2.3 Cryptographic Operation (for data encryption/decryption) (FCS_COP.1(1))

FCS_COP.1.1(1) The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm AES operating in **CBC, ECB, OFB, CFB 1, CFB 8, CFB 128, CTR** and cryptographic key sizes 128-bits and 256-bits that meets the following:

- FIPS PUB 197, "Advanced Encryption Standard (AES)"
- **NIST SP 800-38B, NIST SP 800-38C, NIST SP 800-38D, NIST SP 800-38E**

6.1.2.4 Cryptographic Operation (for cryptographic signature) (FCS_COP.1(2))

FCS_COP.1.1(2) The TSF shall perform cryptographic signature services in accordance with a **(1) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater, or**

that meets the following:

Case: RSA Digital Signature Algorithm

- ~~FIPS PUB 186-2 or FIPS PUB 186-3~~ *FIPS PUB 186-4*, "Digital Signature Standard"

6.1.2.5 Cryptographic Operation (for cryptographic hashing) (FCS_COP.1(3))

FCS_COP.1.1(3) The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm **SHA-1, SHA-224, SHA-256, SHA-384, SHA-512** and message digest sizes **160, 224, 256, 384, 512** bits that meet the following: FIPS Pub 180-3, "Secure Hash Standard."

6.1.2.6 Cryptographic Operation (for keyed-hash message authentication) (FCS_COP.1(4))

FCS_COP.1.1(4) The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm **HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512** key size **160, 224, 256, 384, 512** and message digest sizes **160, 224, 256, 384, 512** bits that meet the following: FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code," and FIPS Pub 180-3, "Secure Hash Standard."

6.1.2.7 Extended: Cryptographic Operation (Random Bit Generation) (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with **NIST Special Publication 800-90 using CTR_DRBG (AES)** seeded by an entropy source that accumulated entropy from **a software-based noise source**.

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of **256 bits** of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

6.1.2.8 Explicit: TLS (FCS_TLS_EXT.1)

FCS_TLS_EXT.1.1 The TSF shall implement one or more of the following protocols **TLS 1.0 (RFC 2246), TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)** supporting the following ciphersuites:

- Mandatory Ciphersuites:
 - TLS_RSA_WITH_AES_128_CBC_SHA
- Optional Ciphersuites:
 - **TLS_RSA_WITH_AES_256_CBC_SHA**
 - **TLS_RSA_WITH_AES_128_CBC_SHA256**
 - **TLS_RSA_WITH_AES_256_CBC_SHA256**

Application Note: *The stunnel command supports only TLS_RSA_WITH_AES_128_CBC_SHA and TLS_RSA_WITH_AES_256_CBC_SHA.*

6.1.2.9 Explicit: HTTPS (FCS_HTTPS_EXT.1)

FCS_HTTPS_EXT.1.1 The TSF shall implement the HTTPS protocol that complies with RFC 2818.

6.1.4.3 Extended: Password-based Authentication Mechanism (FIA_UAU_EXT.2)

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, **none** to perform administrative user authentication.

6.1.4.4 Protected Authentication Feedback (FIA_UAU.7)

FIA_UAU.7.1 The TSF shall provide only obscured feedback to the administrative user while the authentication is in progress at the local console.

NDPP Application Note:

"Obscured feedback" implies the TSF does not produce a visible display of any authentication data entered by a user (such as the echoing of a password), although an obscured indication of progress may be provided (such as an asterisk for each character). It also implies that the TSF does not return any information during the authentication process to the user that may provide any indication of the authentication data.

6.1.5 Security management (FMT)

6.1.5.1 Management of TSF Data (for general TSF data) (FMT_MTD.1)

FMT_MTD.1.1 The TSF shall restrict the ability to manage the TSF data to the Security Administrators.

6.1.5.2 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using **digital signature** capability prior to installing those updates;
- **No other capabilities.**

6.1.5.3 Restrictions on Security Roles (FMT_SMR.2)

FMT_SMR.2.1 The TSF shall maintain the roles:

- Authorized Administrator.

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions

- Authorized Administrator role shall be able to administer the TOE locally;
 - Authorized Administrator role shall be able to administer the TOE remotely;
- are satisfied.

6.1.6 Protection of the TSF (FPT)

6.1.6.1 Extended: Protection of TSF Data (for reading of all symmetric keys) (FPT_SKP_EXT.1)

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

NDPP Application Note:

The intent of the requirement is that an administrator is unable to read or view the identified keys (stored or ephemeral) through "normal" interfaces. While it is understood that the administrator could directly read memory to view these keys, do so is not a trivial task and may require substantial work on the part of an administrator. Since the administrator is considered a trusted agent, it is assumed they would not endeavour in such an activity.

6.1.6.2 Extended: Protection of Administrator Passwords (FPT_APW_EXT.1)

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

6.1.6.3 Reliable Time Stamps (FPT_STM.1)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

6.1.6.4 Extended: Trusted Update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1 The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3 The TSF shall provide a means to verify firmware/software updates to the TOE using a **digital signature mechanism, published hash** prior to installing those updates.

Application Note: *The TOE receives a digitally signed list of hash values for each image contained in an update. The TOE validates the signature and uses the list of hash values to validate the update images.*

6.1.6.5 TSF Testing (FPT_TST_EXT.1)

FPT_TST_EXT.1.1 The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

6.1.7 TOE access (FTA)

6.1.7.1 TSF-initiated Session Locking (FTA_SSL_EXT.1)

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, **lock the session - disable any activity of the user's data access/display devices other than unlocking the session, and requiring that the administrator re-authenticate to the TSF prior to unlocking the session;** after a Security Administrator-specified time period of inactivity.

6.1.7.2 TSF-initiated Termination (FTA_SSL.3)

FTA_SSL.3.1 The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

6.1.7.3 User-initiated Termination (FTA_SSL.4)

FTA_SSL.4.1 The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

6.1.7.4 Default TOE Access Banners (FTA_TAB.1)

FTA_TAB.1.1 Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

6.1.8 Trusted path/channels (FTP)

6.1.8.1 Inter-TSF trusted channel (FTP_ITC.1)

FTP_ITC.1.1 The TSF shall use **TLS** to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, **database server** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2 The TSF shall permit the TSF, or the authorized IT entities to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for **audit server, database server**.

6.1.8.2 Trusted Path (FTP_TRP.1)

FTP_TRP.1.1 The TSF shall use **TLS/HTTPS** to provide a trusted communication path between itself and remote administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and detection of modification of the communicated data.

FTP_TRP.1.2 The TSF shall permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

6.2 Security Functional Requirements Rationale

6.2.1 Security requirements coverage

The following table provides a mapping of SFR to the security objectives, showing that each security functional requirement addresses at least one security objective.

Security functional requirements	Objectives
FAU_GEN.1	O.SYSTEM_MONITORING
FAU_GEN.2	O.SYSTEM_MONITORING
FAU_STG_EXT.1	O.SYSTEM_MONITORING
FCS_CKM.1	O.PROTECTED_COMMUNICATIONS
FCS_CKM_EXT.4	O.PROTECTED_COMMUNICATIONS
FCS_COP.1(1)	O.PROTECTED_COMMUNICATIONS
FCS_COP.1(2)	O.PROTECTED_COMMUNICATIONS
FCS_COP.1(3)	O.PROTECTED_COMMUNICATIONS
FCS_COP.1(4)	O.PROTECTED_COMMUNICATIONS
FCS_RBG_EXT.1	O.PROTECTED_COMMUNICATIONS
FCS_TLS_EXT.1	O.PROTECTED_COMMUNICATIONS
FCS_HTTPS_EXT.1	O.PROTECTED_COMMUNICATIONS
FDP_RIP.2	O.RESIDUAL_INFORMATION_CLEARING
FIA_PMG_EXT.1	O.TOE_ADMINISTRATION
FIA_UIA_EXT.1	O.TOE_ADMINISTRATION
FIA_UAU_EXT.2	O.TOE_ADMINISTRATION
FIA_UAU.7	O.TOE_ADMINISTRATION
FMT_MTD.1	O.TOE_ADMINISTRATION
FMT_SMF.1	O.TOE_ADMINISTRATION
FMT_SMR.2	O.TOE_ADMINISTRATION
FPT_SKP_EXT.1	O.TOE_ADMINISTRATION
FPT_APW_EXT.1	O.TOE_ADMINISTRATION

Security functional requirements	Objectives
FPT_STM.1	O.PROTECTED_COMMUNICATIONS, O.SESSION_LOCK, O.SYSTEM_MONITORING
FPT_TUD_EXT.1	O.VERIFIABLE_UPDATES
FPT_TST_EXT.1	O.TSF_SELF_TEST
FTA_SSL_EXT.1	O.SESSION_LOCK
FTA_SSL.3	O.SESSION_LOCK
FTA_SSL.4	O.SESSION_LOCK
FTA_TAB.1	O.DISPLAY_BANNER
FTP_ITC.1	O.PROTECTED_COMMUNICATIONS
FTP_TRP.1	O.PROTECTED_COMMUNICATIONS

Table 6: Mapping of security functional requirements to security objectives

6.2.2 Security requirements dependency analysis

The following table demonstrates the dependencies of SFRs modeled in CC Part 2 and how the SFRs for the TOE resolve those dependencies.

Security functional requirement	Dependencies	Resolution
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1	FAU_GEN.1
	FIA_UID.1	FIA_UIA_EXT.1
FAU_STG_EXT.1	No dependencies.	
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1]	FCS_COP.1(2)
	FCS_CKM.4	FCS_CKM_EXT.4
FCS_CKM_EXT.4	No dependencies.	
FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Unresolved dependency because the [NDPP] does not include an FCS_CKM.1 for the cryptographic operations defined in FCS_COP.1(1)
	FCS_CKM.4	FCS_CKM_EXT.4
FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1
	FCS_CKM.4	FCS_CKM_EXT.4

Security functional requirement	Dependencies	Resolution
FCS_COP.1(3)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Unresolved dependency because keys are not required for the hashing algorithms defined in FCS_COP.1(3)
	FCS_CKM.4	Unresolved dependency because keys are not required for the hashing algorithms defined in FCS_COP.1(3)
FCS_COP.1(4)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Unresolved dependency because [NDPP] does not include an FCS_CKM.1 for the cryptographic operations defined in FCS_COP.1(4)
	FCS_CKM.4	FCS_CKM_EXT.4
FCS_RBG_EXT.1	No dependencies.	
FCS_TLS_EXT.1	No dependencies.	
FCS_HTTPS_EXT.1	No dependencies.	
FDP_RIP.2	No dependencies.	
FIA_PMG_EXT.1	No dependencies.	
FIA_UIA_EXT.1	No dependencies.	
FIA_UAU_EXT.2	No dependencies.	
FIA_UAU.7	FIA_UAU.1	FIA_UIA_EXT.1 FIA_UAU_EXT.2
FMT_MTD.1	FMT_SMR.1	FMT_SMR.2
	FMT_SMF.1	FMT_SMF.1
FMT_SMF.1	No dependencies.	
FMT_SMR.2	FIA_UID.1	FIA_UIA_EXT.1
FPT_SKP_EXT.1	No dependencies.	
FPT_APW_EXT.1	No dependencies.	
FPT_STM.1	No dependencies.	
FPT_TUD_EXT.1	No dependencies.	
FPT_TST_EXT.1	No dependencies.	
FTA_SSL_EXT.1	No dependencies.	
FTA_SSL.3	No dependencies.	
FTA_SSL.4	No dependencies.	

Security functional requirement	Dependencies	Resolution
FTA_TAB.1	No dependencies.	
FTP_ITC.1	No dependencies.	
FTP_TRP.1	No dependencies.	

Table 7: TOE SFR dependency analysis

6.3 Security Assurance Requirements

The security assurance requirements (SARs) for the TOE are the components defined in the evaluation assurance package NDPP.

The following table shows the SARs, and the operations performed on the components according to CC part 3: iteration (Iter.), refinement (Ref.), assignment (Ass.) and selection (Sel.).

Security assurance class	Security assurance requirement	Source	Operations			
			Iter.	Ref.	Ass.	Sel.
ADV Development	ADV_FSP.1 Basic Functional Specification	NDPP	No	No	No	No
AGD Guidance documents	AGD_OPE.1 Operation User Guidance	NDPP	No	No	No	No
	AGD_PRE.1 Preparative Procedures	NDPP	No	No	No	No
ATE Tests	ATE_IND.1 Independent Testing - Conformance	NDPP	No	No	No	No
AVA Vulnerability assessment	AVA_VAN.1 Vulnerability Survey	NDPP	No	No	No	No
ALC Life-cycle support	ALC_CMC.1 Labeling of the TOE	NDPP	No	No	No	No
	ALC_CMS.1 TOE CM Coverage	NDPP	No	No	No	No

Table 8: SARs

6.3.1 Development (ADV)

6.3.1.1 Basic Functional Specification (ADV_FSP.1)

Developer action elements:

ADV_FSP.1.1D The developer shall provide a functional specification.

ADV_FSP.1.2D The developer shall provide a tracing from the functional specification to the SFRs.

Content and presentation elements:

- ADV_FSP.1.1C** The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.
- ADV_FSP.1.2C** The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.
- ADV_FSP.1.3C** The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.
- ADV_FSP.1.4C** The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

Evaluator action elements:

- ADV_FSP.1.1E** The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.
- ADV_FSP.1.2E** The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

6.3.2 Guidance documents (AGD)

6.3.2.1 Operation User Guidance (AGD_OPE.1)

Developer action elements:

- AGD_OPE.1.1D** The developer shall provide operational user guidance.

Content and presentation elements:

- AGD_OPE.1.1C** The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.
- AGD_OPE.1.2C** The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
- AGD_OPE.1.3C** The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
- AGD_OPE.1.4C** The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
- AGD_OPE.1.5C** The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.
- AGD_OPE.1.6C** The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the Operational Environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

Evaluator action elements:

AGD_OPE.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

6.3.2.2 Preparative Procedures (AGD_PRE.1)

Content and presentation elements:

AGD_PRE.1.1D The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the Operational Environment in accordance with the security objectives for the Operational Environment as described in the ST.

Evaluator action elements:

AGD_PRE.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2E The evaluator *shall apply* the preparative procedures to confirm that the TOE can be prepared securely for operation.

6.3.3 Tests (ATE)

6.3.3.1 Independent Testing - Conformance (ATE_IND.1)

Developer action elements:

ATE_IND.1.1D The developer shall provide the TOE for testing.

Content and presentation elements:

ATE_IND.1.1C The TOE shall be suitable for testing.

Evaluator action elements:

ATE_IND.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2E The evaluator *shall test* a subset of the TSF to confirm that the TSF operates as specified.

6.3.4 Vulnerability assessment (AVA)

6.3.4.1 Vulnerability Survey (AVA_VAN.1)

Developer action elements:

AVA_VAN.1.1D The developer shall provide the TOE for testing.

Content and presentation elements:

AVA_VAN.1.1C The TOE shall be suitable for testing.

Evaluator action elements:

AVA_VAN.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2E The evaluator *shall perform* a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3E The evaluator *shall conduct* penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6.3.5 Life-cycle support (ALC)

6.3.5.1 Labeling of the TOE (ALC_CMC.1)

Developer action elements:

ALC_CMC.1.1D The developer shall provide the TOE and a reference for the TOE.

Content and presentation elements:

ALC_CMC.1.1C The TOE shall be labeled with its unique reference.

Evaluator action elements:

ALC_CMC.2.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

6.3.5.2 TOE CM Coverage (ALC_CMS.1)

Developer action elements:

ALC_CMS.1.1D The developer shall provide a configuration list for the TOE.

Content and presentation elements:

ALC_CMS.1.1C The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2C The configuration list shall uniquely identify the configuration items.

Evaluator action elements:

ALC_CMS.2.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

6.4 Security Assurance Requirements Rationale

The [NDPP] [\[1\]](#) specifies the security assurance requirements (SARs) individually. Any and all rationale for this protection profile's selection of SARs is provided by the protection profile. No modifications and no augmentations have been made to the protection profile's SARs.

7 TOE Summary Specification

7.1 TOE Security Functionality

As per the [NDPP][\[4\]](#), the TOE supports the following major security features:

- Security auditing
- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

The following subsections provide more detail on how the TOE meets the [NDPP][\[4\]](#) requirements.

7.1.1 Security auditing

7.1.1.1 Audit data generation

The TOE generates audit records for the audit events specified by the [NDPP][\[4\]](#). Table 9 contains a mapping of TOE audit records to the events specified in FAU_GEN.1. Each record contains the date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event. When an audit event is caused by an identified user, the audit record will contain the user's identity.

Requirement	Auditable Events	Additional Audit Record Contents
Start-up and shut-down of audit functions		
All administrative actions		
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.

Requirement	Auditable Events	Additional Audit Record Contents
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted channel functions.	Identification of the claimed user identity.
FCS_TLS_EXT.1	Failure to establish a TLS session.	Reason for failure.
	Establishment/Termination of a TLS session.	Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session.	Reason for failure.
	Establishment/Termination of a HTTPS session.	Non-TOE endpoint of connection (IP address) for both successes and failures.

Table 9: TOE audit records

This section maps to the following SFRs)

- FAU_GEN.1
- FAU_GEN.2

7.1.1.2 Audit trail storage

The TOE generates and immediately sends audit records to an audit server, specifically a syslog server, using rsyslog. The audit server is located in the Operational Environment and serves as the external IT entity for storing audit records as required by the [NDPP]. The network connection between the TOE and the audit server is protected by the *stunnel* command which uses the TLS 1.0 protocol contained in the RHEL OpenSSL package. The TOE also stores its audit records in a local 1GB syslog file on the TOE. When this local syslog file becomes full, the oldest records are overwritten by the newest records. This local syslog file allows audit record capture to continue when the network connection between the TOE and audit server is lost. The audit records generated by the TOE during the time that the connection is lost are never forwarded to the audit server. Only administrators who can log in to the local OS console can access the local syslog file.

When the connection between the TOE and audit server is lost, the TOE attempts to reestablish the connection each time a new audit record is generated. If there is an audit record in mid-transmission when the connection is lost, the audit record will exist in the local syslog file, but it will not exist on the remote syslog server.

This section maps to the following SFR:

- FAU_STG_EXT.1

7.1.2 Cryptographic support

7.1.2.1 Cryptographic algorithms

The TOE contains and uses the RHEL 6.6 OpenSSL package (Version 1.0.1e-42.el6_7.2) in FIPS-enabled mode for both TLS and TLS/HTTPS when communicating between itself and other non-TOE systems (i.e., endpoints, relays, remote consoles, the database server, the audit server), except for UDP communications. (Stated another way, both the IBEM Server and the *stunnel* command use the RHEL OpenSSL package in FIPS-enabled mode.)

The TOE generates an asymmetric key pair for use in key establishment with the TLS and TLS/HTTPS protocols. The TOE only supports TLS 1.0, TLS 1.1, and TLS 1.2 in the evaluated configuration corresponding to the standards defined in [FCS_TLS_EXT.1](#).

The IBEM Server supports TLS 1.0, TLS 1.1, and TLS 1.2 and the following TLS ciphersuites in the evaluated configuration (most importantly for the trusted path communication between the IBEM Console and the TOE):

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256

The *stunnel* command, used for the TOE's communication with the audit server and database server, supports TLS 1.0 and the following TLS ciphersuites in the evaluated configuration:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA

The TOE's RSA key establishment conforms to section 6.2 of [NIST800-56B]. The TOE's RSA key generation conforms to section 6.3 of [NIST800-56B] and provides at least 112 bits of key strength.

The TLS and TLS/HTTPS protocols use the [FIPS186-4] RSA algorithm with key sizes of 2048-bits and 3072-bits. These protocols use the AES 128-bits and 256-bits cryptographic algorithms in CBC mode for symmetric cryptography. These protocols also use SHA-1 and SHA-256 hash algorithms for cryptographic hashing and keyed-hash message authentication.

Also, for AES 128-bits and 256-bits, the TOE supports the following block cipher modes of chaining based on the standards defined in [FCS_COP.1\(1\)](#):

- CBC (cipher block chaining)
- CFB 1 (cipher feedback)
- CFB 8
- CFB 128
- CTR (counter)
- ECB (electronic code book)
- OFB (output feedback)

In addition, the TOE supports SHA-224, SHA-256, SHA-384, and SHA-512 for cryptographic hashing corresponding to the standards defined in [FCS_COP.1\(3\)](#) and the hash-based message authentication code (HMAC) algorithms HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 for keyed-hash message authentication corresponding to the standards defined in [FCS_COP.1\(4\)](#).

The TOE overwrites secret keys (used for TLS symmetric encryption), private cryptographic keys (TLS RSA private key), and critical security parameters (DRBG seed values) that are in volatile memory once these values are no longer required. Table 10 lists these values and their destruction algorithms.

Secret type	Use	Storage location	Zeroization algorithm
TOE RSA private key (TLS)	Used for signing during the TLS handshake.	Persistent storage (hard disk drive).	Persists for the life of the product; thus, is never destroyed by the TOE.
		Volatile memory/RAM. Loaded into memory for TLS signing.	Single-pass overwrite using pseudo random numbers once the key is no longer required.
AES session keys (TLS)	AES keys created by the TOE during the TLS handshake and used during a session to encrypt session data.	Volatile memory/RAM	Single-pass overwrite using pseudo random numbers once the key is no longer required.
HMAC keys (TLS)	HMAC keys created by the TOE.	Volatile memory/RAM	Single-pass overwrite using pseudo random numbers once the key is no longer required.
TLS pre-master secret	Created by the TOE during the TLS handshake.	Volatile memory/RAM	Single-pass overwrite using pseudo random numbers once the key is no longer required.
TLS master secret	Created by the TOE during the TLS handshake.	Volatile memory/RAM	Single-pass overwrite using pseudo random numbers once the key is no longer required.
DRBG seed values (TLS)	Used to seed the DRBG.	Volatile memory/RAM	Single-pass overwrite using pseudo random numbers once the key is no longer required.

Table 10: TOE key destruction

Table 11 provides more detail on the supported TLS algorithms.

Algorithms	CAVP Validation #
SP800-56B Pair-Wise Key Establishment Schemes for RSA	See the RSA CAVP Validation # below.
FIPS 186-4 RSA 2048-bits and 3072-bits	RSA: <ul style="list-style-type: none"> IBM BigFix FIPS RHEL OpenSSL (AES-NI and AVX+SSSE3 for SHA) 64 bit: Val#1975

Algorithms	CAVP Validation #
AES-128 & AES-256 with CBC, ECB, OFB, CFB 1, CFB 8, CFB 128, & CTR	AES: <ul style="list-style-type: none"> IBM BigFix FIPS RHEL OpenSSL (AES-NI and AVX+SSSE3 for SHA) 64 bit: Val#3872
SHA-1, SHA-224, SHA-256, SHA-384, & SHA-512	SHA: <ul style="list-style-type: none"> IBM BigFix FIPS RHEL OpenSSL (AES-NI and AVX+SSSE3 for SHA) 64 bit: Val#3191
HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, & HMAC-SHA-512	HMAC: <ul style="list-style-type: none"> IBM BigFix FIPS RHEL OpenSSL (AES-NI and AVX+SSSE3 for SHA) 64 bit: Val#2514
CTR_DRBG (AES)	DRBG: <ul style="list-style-type: none"> IBM BigFix FIPS RHEL OpenSSL (AES-NI and AVX+SSSE3 for SHA) 64 bit: Val#1102
Entropy source: /dev/random	n/a

Table 11: TOE cryptographic algorithms and CAVP validation numbers

The TOE's HTTPS implementation complies with [RFC2818]. The IBEM Console verifies the validity of the TOE's certificate during the TLS handshake. Once the HTTPS session is established, the TOE uses the identification and authentication information provided by the administrator to identify and authenticate the IBEM Console user.

This section maps to the following SFRs:

- FCS_CKM.1
- FCS_CKM_EXT.4
- FCS_COP.1(1)
- FCS_COP.1(2)
- FCS_COP.1(3)
- FCS_COP.1(4)
- FCS_TLS_EXT.1
- FCS_HTTPS_EXT.1

7.1.2.2 Random bit generation

For deterministic random bit generation (DRBG), the RHEL OpenSSL package conforms to the CTR_DRBG (AES) specified in [NIST800-90A]. The TOE uses the RHEL /dev/random file to seed the DRBG.

The RHEL `/dev/random` file uses a software-based noise source that accumulates entropy from various sources such as from user input (e.g., keyboard, mouse), disk randomness, and interrupt randomness. The `/dev/random` file uses an entropy estimator that will block a request until enough entropy has been generated to meet the request. The seed used by the DRBG contains a minimum of 256 bits of entropy.

The DRBG information is included in Table 11.

This section maps to the following SFR:

- FCS_RBG_EXT.1

7.1.3 User data protection

The TOE does not perform packet pass-through of any packets (i.e., the TOE is not a router or switch network device). Because the TOE does not pass-through packets, no zeroization of through packets is performed.

This section maps to the following SFR:

- FDP_RIP.2

7.1.4 Identification and authentication

The TOE's administrative interfaces are:

- IBEM Console (remote GUI)
- Local OS console (RHEL)

All administrative interfaces require each user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that user.

All administrative interfaces support password lengths of 15 characters or greater. For all administrative interfaces, the password composition includes upper case letters, lower case letters, numbers, and special characters. Table 5 contains the list of special characters supported by each interface. These interfaces provide obscured feedback of authentication data to the administrative user during the login process.

The authentication mechanisms for all administrative interfaces are local, but different authentication databases are used. The local OS console uses the local operating system's authentication database. The administrative users of the IBEM Console are vetted against the Console user account table maintained in the database on the database server. In all cases, the TOE performs the actual identification and authentication.

To log in to the IBEM Console (remote GUI), the administrator starts the IBEM Console program. The GUI presents the "Warning Notice" dialog displaying a warning banner to which the administrator clicks OK to dismiss the dialog. The administrator is then presented with the "Login to IBM Endpoint Manager" dialog containing input fields for the "Server" (i.e., IP address of the TOE), the administrator's "User name," and the administrator's "Password." The administrator fills in all three fields and clicks Login. The GUI connects to the TOE using TLS/HTTPS and transmits the administrator's login credentials. When the login is successful, a window containing the full IBEM Console GUI appears. When the login is unsuccessful, a "Login Failed" dialog appears displaying "Incorrect username or password." The administrator clicks OK to dismiss the "Login Failed" dialog and is presented with the previous "Login to IBM Endpoint Manager" dialog.

To log in to the local OS console (RHEL), the administrator is prompted with a warning banner and a login prompt. The administrator enters a user name at the login prompt and presses enter, then the administrator enters a password at the password prompt and presses enter. When the login is successful, the administrator is placed into a shell program and presented with a command line prompt. When the login is unsuccessful, the administrator is re-prompted with the warning banner and the login prompt.

This section maps to the following SFRs:

- FIA_PMG_EXT.1
- FIA_UIA_EXT.1
- FIA_UAU_EXT.2
- FIA_UAU.7

7.1.5 Security management

The TOE allows an authorized administrator to manage the TOE locally via the local OS console interface and remotely via the IBEM Console interface. Only authorized administrators are allowed to use these interfaces and manage the TSF data of the TOE.

For security management of trusted update, see section 7.1.6.4.

The TOE can associate a user with a single role. Note though that the IBEM Console and local OS console interfaces use independent user account and role files/tables for defining user accounts and associating roles. The guidance documentation describes how to properly manage user accounts and roles.

This section maps to the following SFRs:

- FMT_MTD.1
- FMT_SMF.1
- FMT_SMR.2

7.1.6 Protection of the TSF

7.1.6.1 Prevent reading of all symmetric keys

The IBEM Console interface does not provide the ability for the administrator to view any keys stored on the TOE.

The local OS console interface, once the TOE is installed and configured, restricts the login to be a non-root account by disabling the root (superuser) account. Since the TOE's TLS private key is stored in an unencrypted, Base64-formatted file only readable by root, non-root accounts cannot read or view the key. In order to view the private key, an administrator must take the system offline and boot the system in maintenance mode. The symmetric keys, generated by the TLS protocol, are ephemeral. The local OS console does not provide an interface to view ephemeral symmetric keys. Also, the TOE does not contain pre-shared keys.

This section maps to the following SFR:

- FPT_SKP_EXT.1

7.1.6.2 Protection of administrator passwords

For accounts created through the IBEM Server software component (i.e., for the IBEM Console interface), the passwords are hashed using SHA-256 and stored on the database server. The TOE does not store plaintext passwords.

The operating system stores a SHA-512 hashed version of all operating system user passwords. It does not store plaintext passwords.

This section maps to the following SFR:

- FPT_APW_EXT.1

7.1.6.3 Reliable time stamps

The TOE provides reliable time stamps for its own use. The operating system maintains the time stamp mechanism used by the TOE and provides the administrative commands for maintaining the time. The other aspects of the TOE use the operating system's time stamp mechanism; thus, providing a single reliable time source for all TOE functions that require a reliable time source.

The security functions that rely on this time stamp in the evaluated configuration are:

- Audit record generation
- Certificate validation
- Session locking
- Timeouts for remote sessions

This section maps to the following SFR:

- FPT_STM.1

7.1.6.4 Trusted update

For the trusted update process required by the [NDPP][\[1\]](#), the updates for the TOE (the IBEM Server software component, IBEM Client software component, and operating system) are controlled by an administrator via the IBEM Console. Getting the updates to the TOE requires several steps.

When a customer acquires the TOE, they must register with IBM for email notifications of TOE updates using a computer attached to the Internet. When there is an update to the TOE, IBM sends a signed email to the registered email address. This email serves as notice that an update is available. The email contains a link to where the update images are located on the Internet. The administrator downloads the update images from the Internet to a local computer. Since the TOE is on an air gapped network (i.e., not attached to the Internet), the administrator transfers the update images to the TOE using portable media such as a USB drive. The administrator must rename each update image file to be the SHA-1 hash value of the file. (Renaming each file to be the SHA-1 hash value provides file name uniqueness within the update directory.)

Independently, the update Fixlet corresponding to the update images must be obtained from the BigFix Fixlet Server. To do this, the administrator logs in to the TOE's local OS console and runs the Airgap tool on the TOE to generate a Fixlet request file. Once generated, the administrator transfers the Fixlet request file using portable media to a Microsoft Windows system that contains the IBEM Airgap tool and that is attached to the Internet. The administrator runs the Airgap tool on the Microsoft Windows system using the Fixlet request file from the TOE as input. The Airgap tool produces a Fixlet response file that is specific to the TOE from the BigFix Fixlet Server. The Fixlet response file is signed by IBM and contains the update information including the SHA-1 file name for each of the update images and the SHA-256 hash values for each of the update images. The

administrator transfers this Fixlet response file to the TOE using portable media and, using the TOE's local OS console, runs the TOE's Airgap tool providing the Fixlet response file as input to the Airgap tool. The TOE's Airgap tool validates the signature of the Fixlet response file and adds the Fixlet information to the TOE's database.

When the Airgap tool completes the Fixlet response file processing successfully (including the successful digital signature validation of the Fixlet response file), it displays the message "Operation successfully completed." When the Fixlet response file processing fails (including the unsuccessful digital signature validation of the Fixlet response file), the Airgap tool will display a different message based on the type of failure encountered.

Once both the Fixlet response file and the update images are on the TOE and the Fixlet response file successfully processed by the TOE's Airgap tool, the IBEM Console will indicate that updates are available for the TOE. From the IBEM Console, the administrator applies the updates to the TOE. When the updates are applied, the SHA-256 hash value of each update image is validated by the TOE's update process against the signed list of SHA-256 hash values contained in the Fixlet. If the SHA-256 hash values of the update images validate successfully, the updates are applied to the TOE; otherwise, the update process fails.

When the IBEM Consoles update operation completes successfully (including the successful hash value validation of all the update images), it displays the message "100% fixed (X of Y available computers)" where X and Y are computer count values relative to the deployment environment. When the update operation fails (including the unsuccessful hash value validation of one or more update images), the IBEM Console will display a different message based on the type of failure encountered.

Each Fixlet in the response file is digitally signed by IBM with an X.509v3 certificate using RSA with a key size of 2048-bits and SHA-1. These signatures are verified by the TOE's Airgap tool using the RHEL OpenSSL library in FIPS-enabled mode. The hash values contained in the Fixlet of the update images are SHA-256 hashes. The TOE calculates each update image's hash value using the RHEL OpenSSL library in FIPS-enabled mode.

The certificate used to sign the updates as the authorized source contains the following fields:

- Subject:
 - CN = Fixlet Site Admin
 - O = BigFix, Inc.
 - OU = Product Engineering
 - E = fixletsiteadmin@bigfix.com
- Issuer:
 - CN = EnterpriseSite Registrar:Dennis Goodrow
 - O = BigFix, Inc.
 - OU = Site Authorization

The certificate used to validate the digital signature comes with the TOE and is stored in the database on the database server.

An administrator can query the versions of the installed IBEM Server software component, IBEM Client software component, and the operating system via the TOE.

This section maps to the following SFRs:

- `FPT_TUD_EXT.1`
- `FCS_COP.1(2) - RSA`

- FCS_COP.1(3) - SHA-1 and SHA-256

7.1.6.5 Self-tests

The TOE, specifically the RHEL OpenSSL package, performs a series of power-up self-tests when invoked. First, the TOE performs an integrity test on the cryptographic module using HMAC-SHA-256. If the integrity test passes, then the TOE performs a set of Known Answer Tests (KATs) on individual algorithms. If the KAT tests pass, then the cryptographic module enters FIPS-enabled mode. The following cryptographic algorithms are tested during the KAT tests: AES, RSA, HMAC-SHA1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, and CTR_DRBG. The tested algorithms mentioned here are the algorithms critical to this protection profile. Other cryptographic algorithms supported by the cryptographic module are also tested during the power-up self-tests and can cause the power-up self-tests to fail, but these algorithms are not listed above because they are not used by the TOE in the evaluated configuration.

The power-up self-tests provide a sufficient set of tests for ensuring that the cryptographic algorithms used for trusted channel, trusted path, trusted update, and other cryptographic TOE operations will provide the proper protection.

This section maps to the following SFR:

- FPT_TST_EXT.1

7.1.7 TOE access

The TOE's interactive administrative interfaces are:

- IBEM Console (remote GUI)
- Local OS console (RHEL)

The IBEM Console communicates with the TOE using TLS/HTTPS. The supported TLS versions are specified in section 7.1.8. The local OS console is the interface provided by the TOE's attached monitor, keyboard, and mouse.

The TOE displays an administrator-specified advisory notice and consent warning message on all interactive user interfaces prior to establishing an administrative user session. The TOE also allows administrators to terminate their own interactive sessions.

The TOE enforces session locking of all local interactive sessions after an administrator-specified period of inactivity. This applies specifically to the local OS console interface.

The TOE enforces session termination of all remote interactive sessions after an administrator-specified period of inactivity. This applies specifically to the IBEM Console interface.

This section maps to the following SFRs:

- FTA_SSL_EXT.1
- FTA_SSL.3
- FTA_SSL.4
- FTA_TAB.1

7.1.8 Trusted path/channels

The TOE uses trusted communication channels between itself and other trusted IT products. The TOE initiates a connection to the database server which is protected using TLS 1.0 via the *stunnel* command. In addition, the TOE initiates a connection to the audit server which is protected using TLS 1.0 via the *stunnel* command. The *stunnel* command uses the RHEL OpenSSL package in FIPS-enabled mode.

Trusted path/channel	Channel	Secure protocol	Initiated by
Trusted channel	TOE to database server (ODBC)	TLS 1.0	TOE
	TOE to audit server (syslog server)	TLS 1.0	TOE
Trusted path	IBEM Console to TOE	TLS/HTTPS (TLS 1.0, 1.1, 1.2)	IBEM Console

Table 12: Trusted path/channel connections

The TOE uses trusted path communications between itself, specifically the IBEM Server, and the IBEM Console. The IBEM Console initiates and uses TLS/HTTPS to communicate to the TOE. The TOE uses the RHEL OpenSSL package in FIPS-enabled mode for the IBEM Console connection.

This section maps to the following SFRs:

- [FTP_ITC.1](#)
- [FTP_TRP.1](#)

8 Abbreviations, Terminology and References

8.1 Abbreviations

AES

Advanced Encryption Standard

CAVP

Cryptographic Algorithm Validation Program

CBC

Cipher Block Chaining

CFB

Cipher Feedback

CGI

Common Gateway Interface

CPU

Central Processing Unit

CSP

Critical Security Parameter

CTR

Counter

CVL

Component Validation List

DRBG

Deterministic Random Bit Generator

ECB

Electronic Code Book

FIPS

Federal Information Processing Standards

Gbps

Gigabit per second

HDD

Hard Disk Drive

HMAC

Hash-based Message Authentication Code

HTTP

Hypertext Transfer Protocol

HTTPS

Hypertext Transfer Protocol Secure

I&A

Identification and Authentication

ID

Identifier

IBEM

IBM BigFix Endpoint Manager

IP

Internet Protocol

KAT

Known Answer Test

LDAP

Lightweight Directory Access Protocol

NDA

Non-disclosure agreement

NDPP

Protection Profile for Network Devices

ODBC

Open Database Connectivity

OFB

Output Feedback

PC

Personal Computer

PCIe

Peripheral Component Interconnect Express

RAID

Redundant Array of Independent Disks

RBG

Random Bit Generator

RHEL

Red Hat Enterprise Linux

RHN

Red Hat Network

RPM

RPM Package Manager

RSA

Rivest-Shamir-Adleman

SAR

Security Assurance Requirement

SHA

Secure Hash Algorithm

SP

Special Publication

SSD

Solid-State Drive

SSL

Secure Sockets Layer

TCP

Transport Control Protocol

TLS

Transport Layer Security

TOE

Target of Evaluation

UDP

User Datagram Protocol

TSF

TOE Security Functionality

x86-64

Intel x86 64-bit processors

8.2 Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Action

An Action is a change applied to a system in order to remediate issues identified by Fixlets. They are typically scripts written in the BigFix Action Language.

Fixlet

A message that is the mechanism for targeting and describing a problematic situation on a computer and providing an automatic fix for it. For the purposes of this ST, the term Fixlet includes all of the different types of Fixlet messages to include Fixlets, Tasks, Analyses and Baselines.

IBEM Agent

Same as IBEM Client. The guidance documentation sometimes uses the term "IBEM agent," "BigFix agent," or "agent" when referring to the IBEM Client.

Threat Agent

Unauthorized individuals attacking the TOE or authorized users manipulating data they are unauthorized to access.

Trusted Agent

Someone or something trusted to administer the TOE, such as an administrator.

8.3 References

CC	Common Criteria for Information Technology Security Evaluation Version 3.1R4 Date September 2012 Location http://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R4.pdf Location http://www.commoncriteriaportal.org/files/ccfiles/CCPART2V3.1R4.pdf Location http://www.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R4.pdf
FIPS186-4	FIPS PUB 186-4: Digital Signature Standard Date July 2013 Location http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
NDPP	Protection Profile for Network Devices Version 1.1 Date 2012-06-08 Location https://www.niap-ccevs.org/pp/pp_nd_v1.1.pdf
NDPP-ERR3	Security Requirements for Network Devices Errata #3 Date 2014-11-03 Location https://www.niap-ccevs.org/pp/pp_nd_v1.1-err3.pdf
NIST800-56B	NIST Special Publication 800-56B: Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography Date August 2009
NIST800-90A	NIST Special Publication 800 - 90A: Recommendation for Random Number Generation Using Deterministic Random Bit Generators Date January 2012
RFC2818	HTTP Over TLS Author(s) E. Rescorla Date 2000-05-01 Location http://www.ietf.org/rfc/rfc2818.txt