vmware

Carbon Black
EDR

VMware Carbon Black Endpoint Detection and Response (EDR) Server 7.5

Security Target

ST Version: 1.1
July 27, 2021

VMware Carbon Black
1100 Winter Street
Waltham, MA 02451

Prepared By:

Booz | Allen | Hamilton
delivering results that endure

Cyber Assurance Testing Laboratory
1100 West Street
Laurel, MD 20707

Table of Contents

Table of Figures

Table of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation.

1.1.1 ST Identification

ST Title:	VMware Carbon Black Endpoint Detection and Response (EDR) Server 7.5 Security Target
ST Version:	1.1
ST Publication Date:	July 27, 2021
ST Author:	Booz Allen Hamilton

1.1.2 Document Organization

Chapter 1 of this document provides identifying information for the ST and TOE as well as a brief description of the TOE and its associated TOE type.

Chapter 2 describes the TOE in terms of its physical boundary, logical boundary, exclusions, and dependent Operational Environment components.

Chapter 3 describes the conformance claims made by this ST.

Chapter 4 describes the threats, assumptions, objectives, and organizational security policies that apply to the TOE.

Chapter 5 defines extended Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs).

Chapter 6 describes the SFRs that are to be implemented by the TSF.

Chapter 7 describes the SARs that will be used to evaluate the TOE.

Chapter 8 provides the TOE Summary Specification, which describes how the SFRs that are defined for the TOE are implemented by the TSF.

1.1.3 Terminology

This section defines the terminology used throughout this ST. The product-specific terminology used throughout this ST is defined in Table 1. Technology terms that are related to the security functionality claimed by the TOE are defined in the introductory materials of the claimed Protection Profile. These tables are to be used by the reader as a quick reference guide for terminology definitions.

Table 1: Customer Specific Terminology

Term	Definition
Administration Workstation	Any general-purpose computer that is used by an enterprise administrator to manage the TOE remotely via a web browser.
Application Management	Management of the Sensors by an enterprise administrator using the operational environment's VMware Carbon Black EDR Server.
Endpoint System	A device or set of devices, such as a laptop or desktop, with a Windows operating system, that has the host sensors installed.
Endpoint User	An individual who has access to the Endpoint System but is not able to manage its behavior.
Enterprise Administrator	The administrator who has system permissions to access sensitive data and perform management functionality on the VMware Carbon Black EDR Server (TOE).
Host Sensor	Generic term for VMware Carbon Black EDR Windows Sensor.
Sensor Group	Each host sensor is associated with a sensor group that defines its configuration and security characteristics. A sensor group must contain at least one host sensor and can contain many host sensors. However, a single host sensor can only belong to one sensor group. Sensor groups can be based on the security and organizational requirements. For example, one could base sensor groups on functional groupings/departments (such as marketing, customer service, or IT) or location.

Table 2: CC Specific Terminology

Term	Definition
Address Space Layout Randomization (ASLR)	An anti-exploitation feature which loads memory mappings into unpredictable locations. ASLR makes it more difficult for an attacker to redirect control to code that they have introduced into the address space of an application process.
Application (app)	Software that runs on a platform and performs tasks on behalf of the user or owner of the platform, as well as its supporting documentation. The terms *TOE* and *application* are interchangeable in this document.
Application Programming Interface (API)	A specification of routines, data structures, object classes, and variables that allows an application to make use of services provided by another software component, such as a library. APIs are often provided for a set of libraries included with the platform.
Credential	Data that establishes the identity of a user, e.g. a cryptographic key or password.
Data Execution Prevention (DEP)	An anti-exploitation feature of modern operating systems executing on modern computer hardware, which enforces a non-execute permission on pages of memory. DEP prevents pages of memory from containing both data and instructions, which makes it more difficult for an attacker to introduce and execute code.
Developer	An entity that writes application software. For the purposes of this document, vendors and developers are the same.

Operating System (OS)	Software that manages hardware resources and provides services for applications.
Personally Identifiable Information (PII)	Any information about an individual maintained by an agency, including, but not limited to, education, financial transactions, medical history, and criminal or employment history and information which can be used to distinguish or trace an individual's identity, such as their name, social security number, date and place of birth, mother's maiden name, biometric records, etc., including any other personal information which is linked or linkable to an individual.
Platform	The environment in which application software runs. The platform can be an operating system, hardware environment, a software-based execution environment, or some combination of these. These types platforms may also run atop other platforms.
Sensitive Data	Sensitive data may include all user or enterprise data or may be specific application data such as emails, messaging, documents, calendar items, and contacts. Sensitive data must minimally include PII, credentials, and keys. Sensitive data shall be identified in the TSS by the ST author.
Trusted Channel	An encrypted connection between the TOE and a system in the Operational Environment.
Trusted Path	An encrypted connection between the TOE and the application an Authorized Administrator uses to manage it (web browser, terminal client, etc.).
User	In a CC context, any individual who has the ability to manage TOE functions or data.
Vendor	An entity that sells application software. For purposes of this document, vendors and developers are the same. Vendors are responsible for maintaining and updating application software.

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 2. This table is to be used by the reader as a quick reference guide for acronym definitions.

Table 3: Acronym Definition

Acronym	Definition
AA	Assurance Activity
API	Application Programming Interface
ASLR	Address Space Layout Randomization
CB	Carbon Black
CC	Common Criteria
DEP	Data Execution Prevention
DRBG	Deterministic Random Bit Generator
EDR	Endpoint Detection and Response
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IT	Information Technology
LUKS	Linux Unified Key Setup
OS	Operating System
OSP	Organizational Security Policy

PII	Personally Identifiable Information
PP	Protection Profile
NIAP	National Information Assurance Partnership
NIC	Network Interface Card
RBG	Random Bit Generator
RHEL	Red Hat Enterprise Linux
SAR	Security Assurance Requirement
ST	Security Target
TD	Technical Decision
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function
UI	User Interface

1.1.5 Reference

[1] Protection Profile for Application Software, version 1.3 (App PP)
[2] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-001
[3] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-002
[4] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-003
[5] Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-004
[6] NIST Special Publication 800-56A Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography Rev. 3 April 2018
[7] FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard July 2013
[8] Red Hat Enterprise Linux 7.6 Security Target, Version 1.1, June 2020
[9] VMware Carbon Black EDR User Guide, VMware Carbon Black EDR 7.5
[10] VMware Carbon Black EDR Server 7.5 Supplemental Administrative Guidance for Common Criteria, Version 1.1

1.2 TOE Reference

The TOE is VMware Carbon Black Endpoint Detection and Response Server 7.5, which is an application residing on a Red Hat Enterprise Linux (RHEL) OS.

1.3 TOE Overview

The TOE is the VMware Carbon Black Endpoint Detection and Response Server 7.5 application, referred to as VMware CB EDR Server or TOE from this point forward. The VMware CB EDR Server product's primary functionality is receiving endpoint system event data from one or more host sensors for indexing, analyzing, and storing the event data. The VMware CB EDR Server also allows administrators to create and deploy sensor groups, configure host sensors, configure the data collection policy that each host sensor will enforce, update the host sensors, and uninstall host sensors. The VMware CB EDR Server and

host sensor communicate during each host sensor's periodic check-ins. VMware CB EDR Server's primary functionality of collecting, indexing, analyzing, and storing endpoint system event data as well as managing these features was not evaluated, except where the product's functionality relates to the Security Functional Requirements (SFRs) included within the scope of the evaluation.

In the evaluated configuration, as depicted in Figure 1, the TOE's evaluation scope is only the VMware CB EDR Server application and its configuration information. The TOE is an application that is installed on a RHEL 7.6 system with Linux Unified Key Setup (LUKS) encrypted partitioning enabled. The TOE is administered through a web user interface (web UI) via a web browser. Through the web UI, an administrator has the ability to configure the TOE and perform management for the product's primary functionality.

Supporting Environment Interfaces to the TOE:

- **TOE application to OS (1)** –The TOE leverages operating system callbacks for file system access, storage and indexing of received host sensor event data, and accessing network drivers. The TOE invokes the OS for network access and cryptographic support.
- **Web Browser to TOE Platform (2)** – A remote administrator uses a web browser from an administrative workstation to access the TOE's web UI. This transmission is secured using HTTPS session over TLS v1.2 (HTTPS/TLS). It provides a graphical interface to the TOE in order to perform remote administrative activities or to view reports or other graphical displays of system data that is collected by host sensor. Users authenticate to the web UI using username and password. The TOE platform is a HTTPS/TLS server for this interface.
- **Host Sensors Platform to TOE Platform (3)** – The TOE platform responds to a host sensor platform's HTTPS/TLS connection requests in order to establish a trusted communication channel for receiving the collected event data from host sensor and provide configuration changes and updates for the host sensors to pull. The host sensor platform (OS) is the HTTPS/TLS client and performs X.509 certificate validation in support of the non-mutually authenticated HTTPS session over TLS v1.2 communications. The TOE platform is a HTTPS/TLS server for this interface.

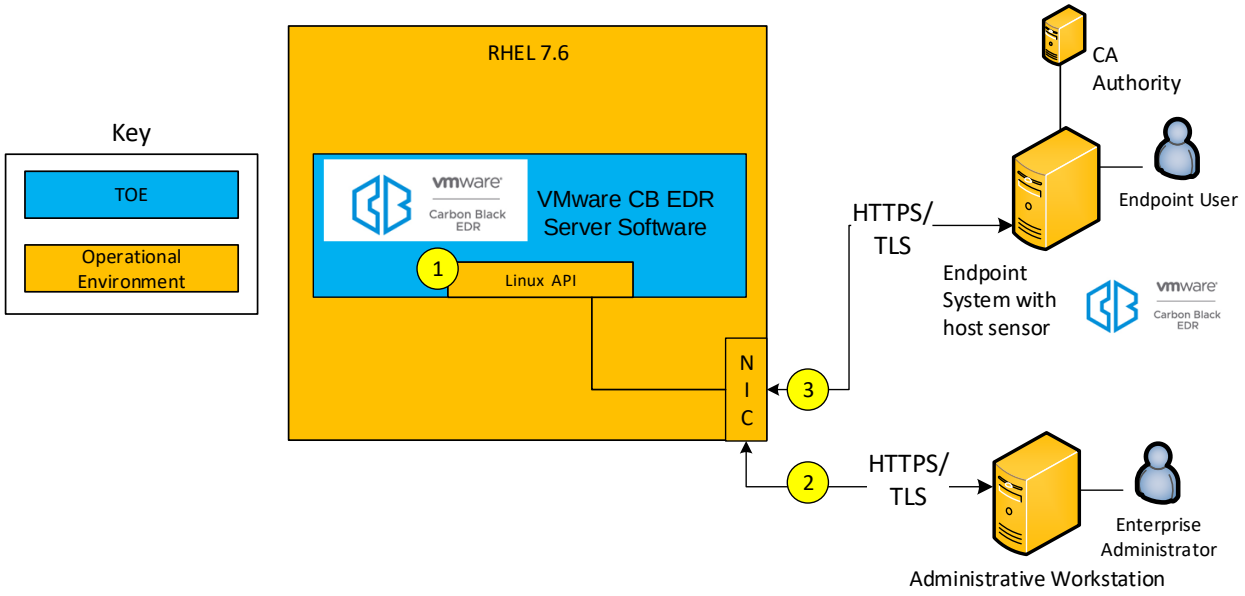

Figure 1: TOE Boundary

1.4 TOE Type

The TOE is application software that is deployed on a RHEL OS and provides administration to an enterprise-level system visibility of IT infrastructure. The [APP_PP] states the following:

"The application, which consists of the software provided by its vendor, is installed onto the platform(s) it operates on. It executes on the platform, which may be an operating system, hardware environment, a software based execution environment, or some combination of these. Those platforms may themselves run within other environments, such as virtual machines or operating systems, that completely abstract away the underlying hardware from the application. The TOE is not accountable for security functionality that is implemented by platform layers that are abstracted away. Some evaluation activities are specific to the particular platform on which the application runs, in order to provide precision and repeatability."

The Application Software TOE type is justified because the TOE is application software that provides administration to an enterprise-level system visibility for the IT infrastructure that is installed on a platform OS that communicates with an endpoint system that collects event data.

2 TOE Description

This section provides a description of the TOE in its evaluated configuration. This includes the physical and logical boundaries of the TOE.

2.1 Evaluated Components of the TOE

The following table describes the TOE components in the evaluated configuration:

Table 4: Evaluated Components of the TOE

Component	Definition
VMware CB EDR Server (TOE)	The TOE is a management server and central data storage for host sensors that have gathered and reported endpoint system event data to the TOE for storage and indexing.

2.2 Components and Applications in the Operational Environment

The following table lists components and applications in the environment that the TOE relies upon in order to function properly:

Table 5: Components of the Operational Environment

Component	Definition
Red Hat Enterprise Linux (RHEL) 7.6 server platform, Dell Inc. PowerEdge R630 with an Intel(R) Xeon(R) E5-2620v4 (Broadwell)	The host platform with the RHEL operating system environment that the TOE application is installed on.**
Endpoint System with Host Sensor (VMware CB EDR Windows Sensor)* *evaluated separately	An application that is installed on a Windows platform which collects event data from the host platform and reports the data back to the TOE.
Administration Workstation	Any general-purpose computer that is used by an enterprise administrator to manage the TOE remotely via a web browser.
Certificate Authority	The server deployed within the Operational Environment which confirms the validity and revocation status of certificates. This is only required for the Endpoint System with Host Sensor to validate TOE server certificate.

**NOTE: It is expected that the TOE is operating on a Common Criteria certified operating system and platform based on the Red Hat Enterprise Linux (RHEL) 7.6 evaluation (VID11039-2020).

2.3 Excluded from the TOE

The following optional products, components, and/or applications can be integrated with the TOE but are not included in the evaluated configuration. They provide no added security related functionality for the evaluated product. They are separated into three categories: not installed, installed but requires a separate license, and installed but not part of the TSF.

2.3.1 Not Installed

There are no optional components that are omitted from the installation process.

2.3.2 Installed but Requires a Separate License

There are no excluded components that are installed and require a separate license.

2.3.3 Installed Functionality Excluded from the Evaluation

The TOE only includes the functionality that are satisfied by the Security Functional Requirements in the claimed Protection Profile. Therefore, the following are considered out of scope of the evaluated configuration because there are no SFRs in the PP that allow this functionality to be claimed:

- Endpoint Detection and Response functionality (detection, analysis, and response)
  - Management of host sensors (create/modify/delete)
  - Management of host sensor certificates
  - Management of host sensor data collection policy
  - Analysis of data functionality
  - Response capabilities

2.4 Physical Boundary

2.4.1 Hardware

This is a software-only TOE. All hardware that is present is part of the TOE's Operational Environment.

2.4.2 Software

The physical boundary of the TOE software is the VMware CB EDR Server application and its configuration data.

2.5 Logical Boundary

The TOE is comprised of several security features. Each of the security features identified above consists of several security functionalities, as identified below.

1. Cryptographic Support
2. User Data Protection
3. Security Management
4. Privacy
5. Protection of the TSF
6. Trusted Path/Channel

2.5.1 Cryptographic Support

The TOE invokes the underlying platform to perform all cryptographic services including HTTPS sessions over TLSv1.2 (HTTPS/TLS) trusted communications and hashing user password credentials for storage.

2.5.2 User Data Protection

The TOE application restricts its access to the host system's network connectivity resources. Network activity is restricted to establishing HTTPS/TLS connections to remote management (via web UI) and

sensor check-in requests. During the host sensor check-in, the TOE receives sensor-collected endpoint system data for the host sensor as well as providing any configuration and software updates for the host sensors to pull during the check-in. The TOE requires LUKS encrypted partitioning to protect local sensitive data storage.

2.5.3 Security Management

The TOE does not provide any default credential used for initial authentication. The TOE uses the underlying platform's recommended methods for storing and setting configuration options. The TOE provides enterprise administrators with the ability to manage the TOE and host sensor through a web UI.

2.5.4 Privacy

The TOE does not transmit any personally identifiable information (PII) over the network.

2.5.5 Protection of the TSF

The TOE is packaged as separate software that is installed on the platform and can be uninstalled/removed if needed. The enterprise administrator can verify the software version from the web UI. All updates are downloaded and installed by an enterprise administrator using the OS software package manager. The digital signature of the update is verified by the platform during installation. Otherwise, the TOE does not download, replace, or modify its own binary code. The TOE implements anti-exploitation features, such as stack-based overflow protection, is compatible with security features provided by the OS, and only uses documented APIs and libraries.

2.5.6 Trusted Path/Channel

The TOE invokes the OS platform to act as a HTTPS/TLS v1.2 non-mutual authentication server for both the host sensor check-in and the remote administrative web browser communication channels.

3 Conformance Claims

3.1 CC Version

This ST is compliant with Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 5 April 2017.

3.2 CC Part 2 Conformance Claims

This ST and Target of Evaluation (TOE) is Part 2 extended to include all applicable NIAP and International interpretations through July 27, 2021.

3.3 CC Part 3 Conformance Claims

This ST and Target of Evaluation (TOE) is Part 3 extended to include all applicable NIAP and International interpretations through July 27, 2021.

3.4 PP Claims

This ST claims exact conformance to the following Protection Profile:

- Protection Profile for Application Software, version 1.3 [App PP]

3.5 Package Claims

The TOE claims exact conformance to the App PP version 1.3, which is extended with CC Part 3.

The TOE claims following Selection-Based SFRs that are defined in the appendices of the claimed PP:

- FCS_CKM.1(1)
- FCS_CKM.2
- FPT_TUD_EXT.2

This does not violate the notion of exact conformance because the PP specifically indicates these as allowable selections and options and provides both the ST author and evaluation laboratory with instructions on how these claims are to be documented and evaluated.

3.6 Package Name Conformant or Package Name Augmented

This ST and TOE are in exact conformance with the App PP.

3.7 Conformance Claim Rationale

The App PP states the following: “The requirements in this document apply to application software which runs on any type of platform. Some application types are covered by more specific PPs, which may be expressed as PP-Modules of this PP. Such applications are subject to the requirements of both this PP and the PP-Module that addresses their special functionality. PPs for some particularly specialized

applications may not be expressed as PP-Modules at this time, though the requirements in this document should be seen as objectives for those highly specialized applications.

Although the requirements in this document apply to a wide range of application software, consult guidance from the relevant national schemes to determine when formal Common Criteria evaluation is expected for a particular type of application. This may vary depending upon the nature of the security functionality of the application."

The TOE is a standalone application which runs on a RHEL 7.6 OS platform and is therefore considered to be relevant to the App PP. There are no PP-Modules to the App PP that are applicable to the product, so the TOE is characterized only as a software application.

3.8 Technical Decisions

Technical Decisions that effected the SFR wording have been annotated with a Footnote. The following is a complete list of Technical Decisions that apply to the App PP evaluation activities that must be performed during the evaluation of this TOE:

Table 6: Technical Decisions

TD #	Title	References	Changes			Analysis to this evaluation	
			SFR	AA	Notes	NA	Reason
TD0587	X.509 SFR Applicability in App PP	FIA_X509_EXT.1, FIA_X509_EXT.2, FTP_DIT_EXT.1	X	X	X	X	FIA_X509_EXT SFRS are not claimed as this is a non-mutual authentication HTTPS/TLS server only. FTP_DIT_EXT.1 has additional SFR option that is not claimed. Therefore, no footnote or changes were made to the claimed SFR.
TD0582	PP-Configuration for Application Software and Virtual Private Network (VPN) Clients now allowed	FDP_DAR_EXT,1	X		X	X	Update to conformance claims. Corrected SFR options that are not claimed. Therefore, no footnote or changes were made to the claimed SFR.
TD0561	Signature verification update	FPT_TUD_EXT.1.4, FPT_TUD_EXT.2	X	X			AA:TSS Footnotes: 3 and 4
TD0554	iOS/iPadOS/Android AppSW Virus Scan	AVA_VAN.1		X			AA: Test modified for iOS/Android platforms and update for AVA_VAN search and analysis applicable to all.
TD0548	Integrity for installation tests in AppSW PP 1.3	FPT_TUD_EXT.1.3		X			AA: Updated Test for iOS; Test wording applied to all other platforms
TD0544	Alternative testing methods for	FPT_AEX_EXT.1		X			AA: Test wording applied to all

	FPT_AEX_EXT.1.1						platforms
TD0543	FMT_MEC_EXT.1 evaluation activity update	FMT_MEC_EXT.1		X		X	Not claiming Windows platform. AA: Test modified for Windows
TD0540	Expanded AES Modes in FCS_COP	FCP_COP.1(1)	X	X		X	Not claiming Cryptography. All handled by OS. SFR modified; AES-CCM Tests modified
TD0519	Linux symbolic links and FMT_CFG_EXT.1	FMT_CFG_EXT.1.2		X			AA: Test modified for Linux
TD0515	Use Android APK manifest in test	FDP_DEC_EXT.1		X		X	Not claiming Android platform AA: Test modified for Android.
TD0510	Obtaining random bytes for iOS/macOS	FCS_RBG_EXT.1		X		X	Not claiming iOS or macOS platform AA: Test modified for iOS and macOS.
TD0498	Application Software PP Security Objectives and Requirements Rationale	Section 4.3 and Section 5.2 in PP			X		Updates PP rationale
TD0495	FIA_X509_EXT.1.2 Test Clarification	FIA_X509_EXT.1.2		X		X	Not claiming X509 AA: Test modified.
TD0473	Support for Client or Server TOEs in FCS_HTTPS_EXT	FCS_HTTPS_EXT.1, FCS_HTTPS_EXT.2	X	X	X	X	Not claiming HTTPS Modification to HTTPS SFR and AA activities
TD0465	Configuration Storage for .NET Apps	FMT_MEC_EXT.1		X		X	Not claiming .NET framework AA: Test
TD0445	User Modifiable File Definition	FPT_AEX_EXT.1.4		X	X		AA: Test User modifiable file definition clarity
TD0437	Supported Configuration Mechanism	FMT_MEC_EXT.1.1	X	X	X		AA: TSS, Tests support of file encryption Footnote 2
TD0435	Alternative to SELinux for FPT_AEX_EXT.1.3	FPT_AEX_EXT.1.3		X			AA: Test modified for Linux
TD0434	Windows Desktop Applications Test	FDP_DEC_EXT.1.1		X		X	Not claiming Window Platform AA: Test modified for Windows
TD0427	Reliable Time Source	A.Platform	X				Changes to wording in ST: Updated wording to Assumption

							Footnote 1
TD0416	Correction to FCS_RBG_EXT.1 Test Activity	FCS_RBG_EXT.1.1		X			AA: Test wording modified for invoking the platform

4 Security Problem Definition

4.1 Threats

This section identifies the threats against the TOE. These threats have been taken from the App PP.

Table 7: TOE Threats

Threat	Threat Definition
T.NETWORK_ATTACK	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may engage in communications with the application software or alter communications between the application software and other endpoints in order to compromise it.
T.NETWORK_EAVESDROP	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may monitor and gain access to data exchanged between the application and other endpoints.
T.LOCAL_ATTACK	An attacker can act through unprivileged software on the same computing platform on which the application executes. Attackers may provide maliciously formatted input to the application in the form of files or other local communications.
T.PHYSICAL_ACCESS	An attacker may try to access sensitive data at rest.

4.2 Organizational Security Policies

There are no Organizational Security Policies in the App PP.

4.3 Assumptions

The specific conditions listed in this section are assumed to exist in the TOE's Operational Environment. These assumptions have been taken from the App PP.

Table 8: TOE Assumptions

Assumption	Assumption Definition
A.PLATFORM[1]	The TOE relies upon a trustworthy computing platform with a reliable time clock for its execution. This includes the underlying platform and whatever runtime environment it provides to the TOE.
A.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.
A.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software in compliance with the applied enterprise security policy.

[1] TD0427

4.4 Security Objectives

This section identifies the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its environment in meeting the security needs.

4.4.1 TOE Security Objectives

This section identifies the security objectives of the TOE as defined by the App PP.

Table 9: TOE Objectives

Objective	Objective Definition
O.INTEGRITY	Conformant TOEs ensure the integrity of their installation and update packages, and also leverage execution environment-based mitigations. Software is seldom, if ever, shipped without errors. The ability to deploy patches and updates to fielded software with integrity is critical to enterprise network security. Processor manufacturers, compiler developers, execution environment vendors, and operating system vendors have developed execution environment-based mitigations that increase the cost to attackers by adding complexity to the task of compromising systems. Application software can often take advantage of these mechanisms by using APIs provided by the runtime environment or by enabling the mechanism through compiler or linker options.
O.QUALITY	To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs.
O.MANAGEMENT	To facilitate management by users and the enterprise, conformant TOEs provide consistent and supported interfaces for their security-relevant configuration and maintenance. This includes the deployment of applications and application updates through the use of platform-supported deployment mechanisms and formats, as well as providing mechanisms for configuration. This also includes providing control to the user regarding disclosure of any PII.
O.PROTECTED_STORAGE	To address the issue of loss of confidentiality of user data in the event of loss of physical control of the storage medium, conformant TOEs will use data-at-rest protection. This involves encrypting data and keys stored by the TOE in order to prevent unauthorized access to this data. This also includes unnecessary network communications whose consequence may be the loss of data.
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application.

4.4.2 Security Objectives for the Operational Environment

The TOE's operating environment must satisfy the following objectives:

Table 10: TOE Operational Environment Objectives

Objective	Objective Definition
OE.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This

	includes the underlying operating system and any discrete execution environment provided to the TOE.
OE.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.
OE.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

4.5 Security Problem Definition Rationale

The assumptions, threats, OSPs, and objectives that are defined in this ST represent the assumptions, threats, OSPs, and objectives that are specified in the Protection Profile to which the TOE claims conformance.

5 Extended Components Definition

5.1 Extended Security Functional Requirements

The extended Security Functional Requirements that are claimed in this ST are taken directly from the PPs to which the ST and TOE claim conformance. These extended components are formally defined in the PPs in which their usage is required.

5.2 Extended Security Assurance Requirements

The extended Security Assurance Requirement that is claimed in this ST is taken directly from the PP to which the ST and TOE claim conformance. This extended component is formally defined in the PP in which its usage is required.

6 Security Functional Requirements

6.1 Conventions

The CC permits four functional component operations—assignment, refinement, selection, and iteration—to be performed on functional requirements. This ST will highlight the operations in the following manner:

- **Assignment:** allows the specification of an identified parameter. Indicated with *italicized* text.
- **Refinement:** allows the addition of details. Indicated with **bold** text and italicized text.
- **Selection:** allows the specification of one or more elements from a list. Indicated with underlined text.
- **Iteration:** allows a component to be used more than once with varying operations. Indicated with a sequential number in parentheses following the element number of the iterated SFR (e.g., "(1)").

When multiple operations are combined, such as an assignment that is provided as an option within a selection or refinement, a combination of the text formatting is used.

If SFR text is reproduced verbatim from text that was formatted in a claimed PP (such as if the PP's instantiation of the SFR has a refinement or a completed assignment), the formatting is not preserved. This is so that the reader can identify the operations that are performed by the ST author as opposed to the PP author.

6.2 Security Functional Requirements Summary

The following table lists the SFRs claimed by the TOE:

Table 11: Security Functional Requirements for the TOE

Class Name	Component Identification	Component Name
Cryptographic Support	FCS_CKM_EXT.1	Cryptographic Key Generation Services
	FCS_CKM.1(1)	Cryptographic Asymmetric Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_RBG_EXT.1	Random Bit Generation Services
	FCS_STO_EXT.1	Storage of Credentials
User Data Protection	FDP_DAR_EXT.1	Encryption of Sensitive Application Data
	FDP_DEC_EXT.1	Access to Platform Resources
	FDP_NET_EXT.1	Network Communications
Security Management	FMT_CFG_EXT.1	Secure by Default Configuration
	FMT_MEC_EXT.1	Supported Configuration Mechanism
	FMT_SMF.1	Specification of Management Functions
Privacy	FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Information
Protection of the TSF	FPT_AEX_EXT.1	Anti-Exploitation Capabilities
	FPT_API_EXT.1	Use of Supported Services and APIs
	FPT_IDV_EXT.1	Software Identification and Versions
	FPT_LIB_EXT.1	Use of Third Party Libraries
	FPT_TUD_EXT.1	Integrity for Installation and Update
	FPT_TUD_EXT.2	Integrity for Installation and Update

Trusted Path/Channel	FTP_DIT_EXT.1	Protection of Data in Transit

6.3 Security Functional Requirements

6.3.1 Class FCS: Cryptographic Support

6.3.1.1 *FCS_CKM_EXT.1 Cryptographic Key Generation Services*

FCS_CKM_EXT.1.1

The application shall [

- invoke platform-provided functionality for asymmetric key generation

].

6.3.1.2 *FCS_CKM.1(1) Cryptographic Asymmetric Key Generation*

FCS_CKM.1.1(1)

The application shall [

- invoke platform-provided functionality

] to generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm [

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following FIPS PUB 186-4, "Digital Signature Standard (DSS), Appendix B.3",
- ECC schemes using "NIST curves" P-256, P-384 and [no other curves] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4

].

6.3.1.3 *FCS_CKM.2 Cryptographic Key Establishment*

FCS_CKM.2.1

The application shall [invoke platform-provided functionality] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method:

[

- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography",

].

6.3.1.4 *FCS_RBG_EXT.1 Random Bit Generation Services*

FCS_RBG_EXT.1.1

The application shall [

- invoke platform-provided DRBG functionality

] for its cryptographic operations.

6.3.1.5 *FCS_STO_EXT.1 Storage of Credentials*

FCS_STO_EXT.1.1

The application shall [

- invoke the functionality provided by the platform to securely store [*user credentials*] ,

] to nonvolatile memory.

6.3.2 Class FDP: User Data Protection

6.3.2.1 *FDP_DAR_EXT.1 Encryption of Sensitive Application Data*

FDP_DAR_EXT.1.1

The application shall [

- leverage platform-provided functionality to encrypt sensitive data,
- protect sensitive data in accordance with FCS_STO_EXT.1,

] in non-volatile memory.

6.3.2.2 *FDP_DEC_EXT.1 Access to Platform Resources*

FDP_DEC_EXT.1.1

The application shall restrict its access to [

- network connectivity

].

FDP_DEC_EXT.1.2

The application shall restrict its access to [

- no sensitive information repositories

].

6.3.2.3 *FDP_NET_EXT.1 Network Communications*

FDP_NET_EXT.1.1

The application shall restrict network communication to [

- respond to [*management web interface request, host sensor check-in request*]

].

6.3.3 Class FMT: Security Management

6.3.3.1 *FMT_CFG_EXT.1 Secure by Default Configuration*

FMT_CFG_EXT.1.1

The application shall provide only enough functionality to set new credentials when configured with default credentials or no credentials.

FMT_CFG_EXT.1.2

The application shall be configured by default with file permissions which protect the application binaries and data files from modification by normal unprivileged users.

6.3.3.2 *FMT_MEC_EXT.1 Supported Configuration Mechanism*

FMT_MEC_EXT.1.1[2]

The application shall [

- invoke the mechanisms recommended by the platform vendor for storing and setting configuration options

].

6.3.3.3 *FMT_SMF.1 Specification of Management Functions*

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions: [

- [*version check*]

].

6.3.4 Class FPR: Privacy

6.3.4.1 *FPR_ANO_EXT.1 User Consent for Transmission of Personally Identifiable Information*

FPR_ANO_EXT.1.1

The application shall [

- not transmit PII over a network

].

[2] TD0437

6.3.5 Class FPT: Protection of the TSF

6.3.5.1 *FPT_AEX_EXT.1 Anti-Exploitation Capabilities*

FPT_AEX_EXT.1.1

The application shall not request to map memory at an explicit address except for [*none*].

FPT_AEX_EXT.1.2

The application shall [

- not allocate any memory region with both write and execute permissions

].

FPT_AEX_EXT.1.3

The application shall be compatible with security features provided by the platform vendor.

FPT_AEX_EXT.1.4

The application shall not write user-modifiable files to directories that contain executable files unless explicitly directed by the user to do so.

FPT_AEX_EXT.1.5

The application shall be built with stack-based buffer overflow protection enabled.

6.3.5.2 *FPT_API_EXT.1 Use of Supported Services and APIs*

FPT_API_EXT.1.1

The application shall use only documented platform APIs.

6.3.5.3 *FPT_IDV_EXT.1 Software Identification and Versions*

FPT_IDV_EXT.1.1

The application shall be versioned with [[*major release number, minor release number, patch number, build date number, build time number*]].

6.3.5.4 *FPT_LIB_EXT.1 Use of Third Party Libraries*

FPT_LIB_EXT.1.1

The application shall be packaged with only [*third-party libraries listed in Table 14*].

6.3.5.5 *FPT_TUD_EXT.1 Integrity for Installation and Update*

FPT_TUD_EXT.1.1

The application shall [leverage the platform] to check for updates and patches to the application software.

FPT_TUD_EXT.1.2

The application shall [provide the ability, leverage the platform] to query the current version of the application software.

FPT_TUD_EXT.1.3

The application shall not download, modify, replace or update its own binary code.

FPT_TUD_EXT.1.4[3]

Application updates shall be digitally signed such that the application platform can cryptographically verify them prior to installation.

FPT_TUD_EXT.1.5

The application is distributed [as an additional software package to the platform OS].

6.3.5.6 *FPT_TUD_EXT.2 Integrity for Installation and Update*

FPT_TUD_EXT.2.1

The application shall be distributed using the format of the platform-supported package manager.

FPT_TUD_EXT.2.2

The application shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events.

FPT_TUD_EXT.2.3[4]

The application installation package shall be digitally signed such that its platform can cryptographically verify them prior to installation.

6.3.6 Class FTP: Trusted Path/Channel

6.3.6.1 *FTP_DIT_EXT.1 Protection of Data in Transit*

FTP_DIT_EXT.1.1

The application shall [

- invoke platform-provided functionality to encrypt all transmitted data with [HTTPS]

] between itself and another trusted IT product.

6.4 Statement of Security Functional Requirements Consistency

The Security Functional Requirements included in the ST represent all required SFRs specified in the claimed PP as well as a subset of the optional SFRs. All hierarchical relationships, dependencies, and unfulfilled dependency rationales in the ST are considered to be identical to those that are defined in the claimed PP.

[3] TD0561
[4] TD0561

7 Security Assurance Requirements

This section identifies the Security Assurance Requirements (SARs) that are claimed for the TOE. The SARs which are claimed are in exact conformance with the App PP.

Security Target (ASE)	ST introduction (ASE_INT.1)
	Conformance claims (ASE_CCL.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Extended components definition (ASE_ECD.1)
	Stated security requirements (ASE_REQ.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance Documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labelling of the TOE (ALC_CMC.1)
	TOE CM coverage (ALC_CMS.1)
	Timely Security Updates (ALC_TSU_EXT.1)
Tests (ATE)	Independent testing – conformance (ATE_IND.1)
Vulnerability Assessment (AVA)	Vulnerability survey (AVA_VAN.1)

7.1 Class ASE: Security Target evaluation

7.1.1 ST introduction (ASE_INT.1)

7.1.1.1 *Developer action elements:*

ASE_INT.1.1D

The developer shall provide an ST introduction.

7.1.1.2 *Content and presentation elements:*

ASE_INT.1.1C

The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C

The ST reference shall uniquely identify the ST.

ASE_INT.1.3C

The TOE reference shall uniquely identify the TOE.

ASE_INT.1.4C

The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C

The TOE overview shall identify the TOE type.

ASE_INT.1.6C

The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C

The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C

The TOE description shall describe the logical scope of the TOE.

7.1.1.3 *Evaluator action elements:*

ASE_INT.1.1E

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

ASE_INT.1.2E

The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

7.1.2 Conformance claims (ASE_CCL.1)

7.1.2.1 *Developer action elements:*

ASE_CCL.1.1D

The developer shall provide a conformance claim.

ASE_CCL.1.2D

The developer shall provide a conformance claim rationale

7.1.2.2 *Content and presentation elements:*

ASE_CCL.1.1C

The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C

The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C

The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C

The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C

The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C

The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C

The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C

The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C

The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C

The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

7.1.2.3 *Evaluator action elements:*

ASE_CCL.1.1E

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

7.1.3 Security objectives for the operational environment (ASE_OBJ.1)

7.1.3.1 *Developer action elements:*

ASE_OBJ.1.1D

The developer shall provide a statement of security objectives.

7.1.3.2 *Content and presentation elements:*

ASE_OBJ.1.1C

The statement of security objectives shall describe the security objectives for the operational environment.

7.1.3.3 *Evaluator action elements:*

ASE_OBJ.1.1E

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

7.1.4 Extended components definition (ASE_ECD.1)

7.1.4.1 *Developer action elements:*

ASE_ECD.1.1D

The developer shall provide a statement of security requirements.

ASE_ECD.1.2D

The developer shall provide an extended components definition.

7.1.4.2 *Content and presentation elements:*

ASE_ECD.1.1C

The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C

The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C

The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C

The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C

The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

7.1.4.3 *Evaluator action elements:*

ASE_ECD.1.1E

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

ASE_ECD.1.2E

The evaluator shall confirm that no extended component can be clearly expressed using existing components.

7.1.5 Stated security requirements (ASE_REQ.1)

7.1.5.1 *Developer action elements:*

ASE_REQ.1.1D

The developer shall provide a statement of security requirements.

ASE_REQ.1.2D

The developer shall provide a security requirements rationale.

7.1.5.2 *Content and presentation elements:*

ASE_REQ.1.1C

The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.1.2C

All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.1.3C

The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.1.4C

All operations shall be performed correctly.

ASE_REQ.1.5C

Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.1.6C

The statement of security requirements shall be internally consistent.

7.1.5.3 *Evaluator action elements:*

ASE_REQ.1.1E

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

7.1.6 TOE summary specification (ASE_TSS.1)

7.1.6.1 *Developer action elements:*

ASE_TSS.1.1D

The developer shall provide a TOE summary specification.

7.1.6.2 *Content and presentation elements:*

ASE_TSS.1.1C

The TOE summary specification shall describe how the TOE meets each SFR.

7.1.6.3 *Evaluator action elements:*

ASE_TSS.1.1E

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

ASE_TSS.1.2E

The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

7.2 Class ADV: Development

7.2.1 Basic Functional Specification (ADV_FSP.1)

7.2.1.1 *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.

7.2.1.2 *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.

7.2.1.3 *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.

7.3 Class AGD: Guidance Documentation

7.3.1 Operational User Guidance (AGD_OPE.1)

7.3.1.1 *Developer action elements:*

AGD_OPE.1.1D

The developer shall provide operational user guidance.

7.3.1.2 *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.

7.3.1.3 *Evaluator action elements:*

AGD_OPE.1.1E

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

7.3.2 Preparative Procedures (AGD_PRE.1)

7.3.2.1 *Developer action elements:*

AGD_PRE.1.1D

The developer shall provide the TOE including its preparative procedures.

7.3.2.2 *Content and presentation elements:*

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.

7.3.2.3 *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.

7.4 Class ALC: Life Cycle Support

7.4.1 Labeling of the TOE (ALC_CMC.1)

7.4.1.1 *Developer action elements:*

ALC_CMC.1.1D

The developer shall provide the TOE and a reference for the TOE.

7.4.1.2 *Content and presentation elements:*

ALC_CMC.1.1C

The TOE shall be labeled with its unique reference.

7.4.1.3 *Evaluator action elements:*

ALC_CMC.1.1E

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

7.4.2 TOE CM Coverage (ALC_CMS.1)

7.4.2.1 *Developer action elements:*

ALC_CMS.1.1D

The developer shall provide a configuration list for the TOE.

7.4.2.2 *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.

7.4.2.3 *Evaluator action elements:*

ALC_CMS.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and

presentation of evidence.

7.4.3 Timely Security Updates (ALC_TSU_EXT.1)

7.4.3.1 *Developer Actions Element:*

ALC_TSU_EXT.1.1D

The developer shall provide a description in the TSS of how timely security updates are made to the TOE.

ALC_TSU_EXT.1.2D

The developer shall provide a description in the TSS of how users are notified when updates change security properties or the configuration of the product.

7.4.3.2 *Content and presentation elements:*

ALC_TSU_EXT.1.1C

The description shall include the process for creating and deploying security updates for the TOE software.

ALC_TSU_EXT.1.1C

The description shall express the time window as the length of time, in days, between public disclosure of a vulnerability and the public availability of security updates to the TOE.

ALC_TSU_EXT.1.1C

The description shall include the mechanisms publicly available for reporting security issues pertaining to the TOE.

7.4.3.3 *Evaluator action elements:*

ALC_TSU_EXT.1.1E

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

7.5 Class ATE: Tests

7.5.1 Independent Testing - Conformance (ATE_IND.1)

7.5.1.1 *Developer action elements:*

ATE_IND.1.1D

The developer shall provide the TOE for testing.

7.5.1.2 *Content and presentation elements:*

ATE_IND.1.1C

The TOE shall be suitable for testing.

7.5.1.3 *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.

7.6 Class AVA: Vulnerability Assessment

7.6.1 Vulnerability Survey (AVA_VAN.1)

7.6.1.1 *Developer action elements:*

AVA_VAN.1.1D

The developer shall provide the TOE for testing.

7.6.1.2 *Content and presentation elements:*

AVA_VAN.1.1C

The application shall be suitable for testing.

7.6.1.3 *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.

8 TOE Summary Specification

8.1 Cryptographic Support

8.1.1 FCS_CKM_EXT.1 and FCS_CKM.1(1)

The TOE invokes the operating system for establishing HTTPS/TLS communications (FTP_DIT_EXT.1) and relies on the operating system's OpenSSL cryptographic module to generate the Elliptic curve-based keys in support of the Elliptic curve-based key establishment.

The TOE invokes the operating system's OpenSSL cryptographic module to generate RSA scheme certificates for the Sensor Group Certificate. These certificates are used for host sensor group identification during the check-in cycles. All members of a sensor group use the same sensor group certificate for group identification.

The following are the key scheme specifications:

- ECC schemes using "NIST curves" P-256, P-384 and no other curves that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4. ECC schemes are used for supporting HTTPS/TLS communication (FPT_DIT_EXT.1)
- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following FIPS PUB 186-4, "Digital Signature Standard (DSS), Appendix B.3.

8.1.2 FCS_CKM.2

The TOE invokes the platform to use Elliptic curve-based key establishment scheme for the establishment of HTTPS/TLS communications. This key establishment scheme supports the usage of web browser for remote administration and host sensor platform communications. Elliptic curve-based key establishment conforms to NIST SP 800-56A.

NOTE: The RSA scheme certificates generated under FCS_CKM_EXT.1(1) are for host sensor group identification during the check-in cycles. The sensor group certificate is not used for HTTPS/TLS communication establishment and is therefore not claimed for key establishment.

8.1.3 FCS_RBG_EXT.1

The TOE invokes the platform for key generation and trusted communications; therefore, the platform calls on the DRBG services required.

From Python the TOE calls os.urandom which calls OS getrandom() in blocking mode which pulls entropy from /dev/urandom to:

- Setup seed for pbkdf2_hmac encryption algorithm used for hashing user passwords in support of FDP_DAR_EXT.1 Encryption of Sensitive Application Data.

The TOE provides no ability to specify the use of an alternative DRBG.

8.1.4 FCS_STO_EXT.1

The TOE invokes the underlying platform to provide the hash of the user password before storing the credentials in the internal database. At no time is a password stored as plain text. The TOE requires the

underlying platform to provide data-at-rest encryption using LUKS encrypted partitioning to protect the internal database.

The TOE does not invoke or implement the storage of keys. CA and server keys are manually created outside of the TOE and must be manually stored on the platform per RHEL guidance (/etc/pki/ca-trust/ and /etc/pki/tls/certs/) as part of the pre-requisite installation procedures of the TOE.

8.2 User Data Protection

8.2.1 FDP_DAR_EXT.1

The TOE defines the following sensitive data objects:

- Sensor Group Certificates: identification of the host sensor's group used during the check-in cycles with TOE. All members of a sensor group use the same sensor group certificate for group identification. This identifier allows the TOE to know how to classify the source of information received from the host sensor and what host sensor policies the TOE must provide to the host sensor during the check-in cycles. These are not used for HTTPS/TLS communication establishment.
- User credentials: usernames and the hashed representation of the passwords are stored in accordance with FCS_STO_EXT.1.

Sensitive data is stored in the TOE's internal database. As a result of the OS platform being Linux based, the TOE requires the underlying platform to provide data-at-rest encryption using LUKS encrypted partitioning to protect the internal database.

Instructions for preparing the Operational Environment, including the requirement for LUKS, are provided in the supplemental administrative guidance that is included with the TOE.

8.2.2 FDP_DEC_EXT.1

The TOE relies on its underlying platform to provide the actual network connectivity in order to establish communications channels. There are no sensitive information repositories (storage locations for private user or administrator data) that the TOE requires access.

8.2.3 FDP_NET_EXT.1

The TOE requires the OS platform to initiate a HTTPS/TLS (TLS v1.2) server process that only supports the following two remote connections requests:

- response to requests from a remote administrator using a web browser to access the TOE's web UI. Port for remote administration: 443
- response to the host sensor platform's requests for a HTTPS/TLS connection for the periodic check-in. If the HTTPS/TLS connection is interrupted, the host sensor platform will automatically attempt to re-establish the connection. Port for host sensor communication: 443 by default, but can be configured by the administrator

  During a check-in,

- the TOE receives endpoint host OS telemetry data (processes/threads being created, filesystem activity, registry activity, etc.), system logs, and/or memory dumps in accordance with active data collection configuration from the host sensor.
- the host sensor pulls configuration and data collection policy updates, Sensor Group change information including new sensor group certificate (if applicable), notification of host sensor software update being available, and host sensor update package (if applicable).

The TOE does not provide client functionality and does not initiate external network connections to another device.

8.3 Security Management

8.3.1 FMT_CFG_EXT.1

The TOE requires credentials for administration via the web UI. The initial installation of the TOE prompts the enterprise administrator to create a username and password. There are no default credentials. The TOE software is automatically installed with the appropriate file permissions to prevent unauthorized access to the binaries, configuration settings, and data.

8.3.2 FMT_MEC_EXT.1

The TOE relies on the underlying platform's recommended methods for storing and setting configuration options[5]. However, there are no SFR related configuration items stored on the system. The claimed SFR behavior and configuration is hard coded into the TOE binaries and are not configurable.

8.3.3 FMT_SMF.1

The TOE provides a remote web UI to allow enterprise administrators to manage the TSF. The interface provides functionality that is related to the management of the server application itself. Only the functions that are relevant to the TSF and have SFR claims are discussed in this Security Target. The following security-relevant management functions are provided by the TOE:

- Ability to query the software version (FPT_TUD_EXT.1).

There is no management function for checking for TOE updates. The actual downloading and installation of any downloaded updates is manually performed by a person with root access to the platform and therefore is not included in this list.

8.4 Privacy

8.4.1 FPR_ANO_EXT.1

The TOE application does not collect personally identifiable information (PII) for administrators or users. Therefore, the TOE application does not transmit PII data over the network.

[5] RHEL Storage Administration Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html-single/storage_administration_guide/index)

8.5 Protection of the TSF

8.5.1 FPT_AEX_EXT.1

The TOE implements several mechanisms to protect against exploitation. The application does not map memory to an explicit address, allocate any memory region with write and execute permissions, nor does it write user-modifiable files to directories that contain executable files.

The application is compatible with the platform-provided security feature SELinux enabled and enforcing. The TOE provides its own copy of Python compiled with -fPIE flags. The python-based TOE subcomponents are also compiled with -fPIE flags to ensure ASLR enforcement. The Java-based TOE subcomponents rely on the default Java virtual machine that comes with RHEL distribution.

8.5.2 FPT_API_EXT.1

When the TOE is installed on the RHEL OS, it uses only the following supported platform APIs in order to function.

Table 12: TOE APIs

Java Framework API:		
javax.naming.ldap.LdapName	java.net.InetSocketAddress	java.sql
javax.naming.ldap.Rdn	java.net.MalformedURLException	java.text
javax.servlet.	java.net.ServerSocket	java.time
javax.xml.bind.DatatypeConverter	java.net.Socket	java.util
java.io	java.net.UnknownHostException	java.util.zip
java.lang.annotation	java.net.URI	javax.naming.ldap
java.lang.invoke.MethodHandles	java.net.URL	javax.servlet
java.lang.management	java.nio.BufferUnderflowException	javax.xml.bind.DatatypeConverter
java.lang.reflect	java.nio.file	org.slf4j.Logger
java.lang.Thread	java.security.cert.CertificateFactory	org.slf4j.LoggerFactory
java.net.InetAddress	java.security.cert.X509Certificate	
Python Framework API:		
abc	grp	pytz
antlr4	gzip	queue
argparse	hashlib	random
ast	heapq	rapidjson
atexit	http	re
base64	imp	requests
binascii	importlib	saml2
bisect	inspect	select
boto3	io	selinux
bs4	ipaddress	setuptools
builtins	itertools	shlex
cprofile	json	shutil
calendar	kombu	signal
code	locale	simplejson
codecs	logging	smtplib
collections	markdown	socket

configparser	math	stackcollector
contextlib	migrate	stat
copy	mmap	struct
coverage	multiprocessing	subprocess
crypto	netaddr	supervisor
csv	numpy	sys
ctypes	openssl	tabulate
difflib	operator	tarfile
distutils	optparse	tempfile
email	os	termcolor
enum	pathlib	textwrap
errno	pickle	threading
fcntl	pid	traceback
fileinput	pkg_resources	urllib
fnmatch	pkgutil	urllib3
functools	platform	warnings
gc	pprint	weakref
getpass	pstats	werkzeug
gevent	psutil	xml
glob	psycopg2	xmlrpc
greenlet	pwd	zipfile
ghrequests	pyformance	zlib

8.5.3 FPT_IDV_EXT.1

The TOE is versioned using "major.minor.patch.builddate.buildtime" methodology. Major version updates happen when incompatible API changes occur, minor version updates happen when backwards-compatible functionality is added, and patch version updates happen when backwards-compatible bug fixes are implemented. The build date and time reflect the timestamp when compilation occurred.

8.5.4 FPT_LIB_EXT.1

The TOE is packaged with the following third-party open source libraries in order to function.

Table 13: TOE Libraries

Apache Lucene	Google protobuf	OpenResty	RabbitMq
Apache Solr	Jetty	PostgreSql	Redis
Erlang	OpenJDK-11	Python	Tzdata-java

8.5.5 FPT_TUD_EXT.1 and FPT_TUD_EXT.2

The TOE application is packaged in an .rpm file format for the RHEL OS platform. The product's individual RPMs (installations and updates) are signed using VMware CarbonBlack certificate via GPG signing. The YUM repo file is configured to validate GPG signatures automatically. YUM uses the VMware CB public key, available from the same VMware CB EDR REPO that the .rpm was downloaded from, to validate the signatures. With signature verification enabled, YUM will refuse to install any packages not GPG-signed with the correct key for that repository. The binary code is only modified or replaced when a manual software update is performed from the OS. The TOE does not automatically download, modify, replace, or update its own binaries or executable files. The initial installation requires

the enterprise administrator to download a customer specific .rpm file from the VMware CB website onto the host server platform and perform “yum install”.

The TOE and the platform OS provide the enterprise administrator the ability to check the version of the TOE that is currently running on the machine. By using the web UI, the current version is displayed on the bottom right-hand corner of the management console on all webpages. The version of the TOE can also be verified by using the "yum info cb-enterprise" command.

The enterprise administrator can check to see if an update is available by running the “yum check-update cb-enterprise” command. If there is an update available, the enterprise administrator runs the “yum upgrade cb-enterprise” command that allows the TOE to be updated. The TOE does not download or install updates automatically.

Uninstallation of the TOE can be performed by the running the YUM utility to remove all remnants that constitute the TOE or remove packages plus all other third-party packages upon which it depends. When the TOE application is uninstalled, settings are left behind for the purposes of re-installing the TOE software while maintaining previous configuration (recovery).

8.5.5.1 *Timely Security Updates*

As part of providing timely security updates, VMware Carbon Black provides customers with a support section on CarbonBlack.com where they have the ability to submit support issues through the User Exchange link. High severity issues can result in a patch release as soon as remediation is available. Lower severity issues will be incorporated into the next scheduled release. Security fixes will be released as new packages in the same manner as any feature updates (see discussion on FPT_TUD_EXT.1 above). The TOE contains third-party components that VMware Carbon Black does not have control over the implementation of. Any implementation flaws are expected to be addressed within 90 days of reporting. Customers are notified of security-related fixes on the VMware Carbon Black customer portal.

8.6 Trusted Path/Channel

8.6.1 FTP_DIT_EXT.1

The TOE invokes the RHEL distributed OpenSSL library to establish a HTTPS session over TLS v1.2 non-mutual authentication secure channel for all transmitted data. The TOE’s OS platform acts as the HTTPS/TLS server to establish secure communications to the host sensor platform for protecting the data traversing the channel from disclosure and/or modification. In particular:

- A trusted communication between itself and the host sensor platform.
- A trusted communication between itself and the web browser for remote administration.