Assurance Activities Report
for a Target of Evaluation

VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2

Assurance Activities Report (AAR)
Version 1.3

July 21, 2021

Security Target (Version 1.5)

Evaluated by:

Booz | Allen | Hamilton

delivering results that endure

Booz Allen Hamilton Common Criteria Test Laboratory
NIAP Lab # 200423
1100 West St.
Laurel, MD 20707

Prepared for:
National Information Assurance Partnership
Common Criteria Evaluation and Validation Scheme

The Developer of the TOE:
VMware Carbon Black
1100 Winter Street
Waltham, MA 02451

The Author of the Security Target:
Booz Allen Hamilton
1100 West St.
Laurel, MD 20707 USA

The TOE Evaluation was sponsored by:
Booz Allen Hamilton

Evaluation Personnel:
Herbert Markle
Christopher Rakaczky
Courtney Simon

Applicable Common Criteria Version
Common Criteria for Information Technology Security Evaluation, April 2017 Version 3.1 Revision 5

Common Evaluation Methodology Version
Common Criteria for Information Technology Security Evaluation, Evaluation Methodology, April 2017 Version 3.1 Revision 5

Table of Contents

1 Purpose

The purpose of this document is to serve as a non-proprietary attestation that this evaluation has satisfied all of the TSS, AGD, and ATE Assurance Activities required by the Protection Profiles, Extended Packages, and/or PP-Modules to which the TOE claims exact conformance. This will give system integrators valuable information about product configuration and testing, help to align Common Criteria evaluations with DISA Security Requirements Guides and Security Test Implementation Guides (SRGs/STIGs), and thereby streamline the process for U.S. Government procurement of validated products.

2 TOE Summary Specification Assurance Activities

The evaluation team completed the testing of the Security Target (ST) *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Security Target* and confirmed that the TOE Summary Specification (TSS) contains all Assurance Activities as specified by the *Protection Profile for Application Software Version 1.3* [APP_PP]. The evaluators were able to individually examine each SFR's TSS statements and determine that they comprised sufficient information to address each SFR claimed by the TOE as well as meet the expectations of the [APP_PP] Assurance Activities.

Through the evaluation of ASE_TSS.1-1, described in the ETR, the evaluators were able to determine that each SFR was described in enough detail to demonstrate that the TSF addresses the SFR. However, in some cases the Assurance Activities that are specified in the claimed source material instruct the evaluator to examine the TSS for a description of specific behavior to ensure that each SFR is described to an appropriate level of detail. The following is a list of each SFR, the TSS Assurance Activities specified for the SFR, and how the TSS meets the Assurance Activities. Additionally, each SFR is accompanied by the source material [APP_PP] that defines where the most up-to-date TSS Assurance Activity was defined.

FCS_CKM_EXT.1.1 – "*The evaluator shall inspect the application and its developer documentation to determine if the application needs asymmetric key generation services. If not, the evaluator shall verify the generate no asymmetric cryptographic keys selection is present in the ST. Otherwise, the evaluation activities shall be performed as stated in the selection-based requirements."*

Based on the inspection of the application, there is no administrator interface for generating asymmetric keys. Additionally, there is no TOE administration or installation documentation that states the need or ability to generate asymmetric keys on the host platform. Therefore, the "generate no asymmetric cryptographic keys" should be present which it is. The TSS consistently states in section 8.1.1: "The TOE does not perform any asymmetric key generation. Sensor Group certificates, for HTTP/TLS communication, are installed as part of the installation and are generated on the management server, not on the TOE's host endpoint system."

Based on the ST containing the correct selection and the TSS consistently identifies the TOE does not generate the asymmetric cryptographic keys this assurance activity is satisfied.

FCS_RBG_EXT.1.1 – *"If use no DRBG functionality is selected, the evaluator shall inspect the application and its developer documentation and verify that the application needs no random bit generation services.*

If implement DRBG functionality is selected, the evaluator shall ensure that additional FCS_RBG_EXT.2 elements are included in the ST.

If invoke platform-provided DRBG functionality is selected, the evaluator performs the following activities. The evaluator shall examine the TSS to confirm that it identifies all functions (as described by the SFRs included in the ST) that obtain random numbers from the platform RBG. The evaluator shall determine that for each of these functions, the TSS states which platform interface (API) is used to obtain the random

numbers. The evaluator shall confirm that each of these interfaces corresponds to the acceptable interfaces listed for each platform below.

It should be noted that there is no expectation that the evaluators attempt to confirm that the APIs are being used correctly for the functions identified in the TSS; the activity is to list the used APIs and then do an existence check via decompilation."

The ST claims the "use no DRBG functionality" selection. The evaluator, as part of testing, performed a code analysis and documentation search for use of DRBG for TOE defined functionality (see Section 4 FCS_RBG_EXT.1.1 Test 001B of this document) and found no invocation of rand_s, RtlGenRandom, BCryptGenRandom, or CryptGenRandom API. The TSS states in section 8.1.2: "The TOE does not call on DRBG services. The TOE invokes the Windows platform for encrypted storage of credentials and trusted communications. Therefore, the Windows platform calls on the DRBG services required."

Based on the ST containing the selection, "use no DRBG functionality", the TSS consistently identifies the TOE invokes the OS for cryptographic functions such storage of credentials and trusted communications, and the static code review found no invocation of DRBG this assurance activity is satisfied.

FCS_STO_EXT.1.1 – "*The evaluator shall check the TSS to ensure that it lists all persistent credentials (secret keys, PKI private keys, or passwords) needed to meet the requirements in the ST. For each of these items, the evaluator shall confirm that the TSS lists for what purpose it is used, and how it is stored."*

The TSS provides in section 8.1.3 a table for Windows key materials, which is consistent with the SFR requirement. For each item listed in the table, the TSS lists the purpose in the "Purpose" column and how it is stored under the "Storage Location" column. Based on the TSS identifying the credentials (Management Server certificate (public key) and sensor group certificate (TLS Cert Private Key), and how they are stored (in the Microsoft Key Storage Provider) this assurance activity is satisfied.

Based on the TSS description covering the purpose and storage this assurance activity is satisfied.

FDP_DAR_EXT.1.1 – "*The evaluator shall examine the TSS to ensure that it describes the sensitive data processed by the application. The evaluator shall then ensure that the following activities cover all of the sensitive data identified in the TSS.*

If not store any sensitive data is selected, the evaluator shall inspect the TSS to ensure that it describes how sensitive data cannot be written to non-volatile memory. The evaluator shall also ensure that this is consistent with the filesystem test below."

The TSS states in section 8.2.1 that: "The TOE invokes the platform for securely storing certificates and keys based on the Windows key storage.

There is no interactive user interface provided by the TOE and therefore there is no user data generation possible from the TOE. The TOE itself does record the collected event information into log files that is stored on the system hard drive. This information has the potential of containing information gathered from the sensitive data files such as system logs or memory dumps, as defined in FDP_DEC_EXT.1.2. Therefore, the TOE relies on the underlying operating system for encryption of this potentially sensitive data. Windows platforms do not provide data-at-rest encryption. Therefore, additional programs like BitLocker or Encrypting File System (EFS) must be used."

Based on the TSS covering both the sensitive information securely stored under FCS_STO_EXT.1 and the identified system logs and memory dumps requiring the OS to have BitLocker enabled (full disk encryption) this assurance activity is satisfied.

FDP_DEC_EXT.1.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FDP_DEC_EXT.1.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FDP_NET_EXT.1.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FIA_X509_EXT.1.1 – **TD0587** – *"The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place. The evaluator ensures the TSS also provides a description of the certificate path validation algorithm."*

The TSS states in section 8.3.1 that the check of the validity of certificates takes place using OCSP. This section also describes the certificate path algorithm, specifically that all certificate paths terminate with a trusted root CA certificate and that all CA certificates include the basicConstraints extension with the CA flag set to TRUE. The certificate validation service will also ensure that the extendedKeyUsage field is properly set for all certificates depending on their intended usage.

FIA_X509_EXT.1.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FIA_X509_EXT.2.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FIA_X509_EXT.2.2 – *"The evaluator shall check the TSS to ensure that it describes how the TOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.*

The evaluator shall examine the TSS to confirm that it describes the behavior of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The evaluator shall verify that any distinctions between trusted channels are described."

The TSS states in Section 8.3.1 that: "The management server certificate chain is manually installed on the host platform, via OS level commands, as a pre-requisite for installing the TOE. The Sensor Group certificate is installed as part of the installation of the TOE and is installed into the Windows certificate store."

Additionally, "The HTTPS/TLS implementation will automatically reject a certificate if a connection to the OCSP cannot be established and if the certificate is found to be invalid in any way, including if the revocation status is returned as unknown or revoked."

Based on the TSS stating how the TOE is installed and configured to use the correct certificates, that the TOE will reject the certificate when a connection to the OSCP cannot be established, and will reject the certificate if the revocation status is "unknown or revoked", this assurance activities is considered satisfied.

FMT_CFG_EXT.1.1 – "*The evaluator shall check the TSS to determine if the application requires any type of credentials and if the application installs with default credentials."*

The TSS states in section 8.4.1 that: "The TOE does not provide an interactive user interface and therefore, does not require user credentials (default or otherwise) to operate. The TOE operates as a Windows service. In the evaluated configuration, the only credentials used are the certificates used to support HTTPS/TLS communication between the TOE and the management server. The management server certificate is manually installed as a pre-requisite for installing the TOE. The Sensor Group certificate is installed as part of the installation of the TOE and is installed into the Windows certificate store."

Based on the TSS stating that the TOE does not provide an interactive user interface nor require user credentials (default or otherwise) the assurance activity is satisfied.

FMT_CFG_EXT.1.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FMT_MEC_EXT.1.1 – **TD0437** – "*The evaluator shall review the TSS to identify the application's configuration options (e.g. settings) and determine whether these are stored and set using the mechanisms*

supported by the platform or implemented by the application in accordance with the PP-Module for File Encryption. At a minimum the TSS shall list settings related to any SFRs and any settings that are mandated in the operational guidance in response to an SFR.

Conditional: If "implement functionality to encrypt and store configuration options as defined by FDP_PRT_EXT.1 in the PP-Module for File Encryption" is selected, the evaluator shall ensure that the TSS identifies those options, as well as indicates where the encrypted representation of these options is stored."

The TSS states in section 8.4.2 that: "All configuration settings, as established by the management server, are stored on the TOE according to the Windows platform best practices. The credential data is stored according to FCS_STO_EXT.1. Additionally, the TOE application uses the Windows Registry (HKLM/Software/CarbonBlack) to store application configuration settings."

Based on FDP_PRT_EXT.1 not being claimed and the TSS description that the TOE uses the Windows registry and Windows certificate store (the recommended mechanism for a Windows desktop application) this assurance activity is satisfied.

FMT_SMF.1.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPR_ANO_EXT.1.1 – "*The evaluator shall inspect the TSS documentation to identify functionality in the application where PII can be transmitted."*

The TSS states in section 8.5.1 that "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."

Based on the TSS description that the TOE does not transmit PII this assurance activity is satisfied.

FPT_AEX_EXT.1.1 – "*The evaluator shall ensure that the TSS describes the compiler flags used to enable ASLR when the application is compiled."*

The TSS states in section 8.6.1 that: "The TOE is compiled with flags to ensure anti-exploitation capabilities are enabled for address space layout randomization (ASLR), Data Execution Prevention (DEP), and buffer overflow protection." The description provides a table that describes all of the compiler flags the vendors used.

Based on the TSS description and the identified compiler flags listed are associated with ensuring ASLR, DEP, and overflow protection is enabled this assurance activity is satisfied.

FPT_AEX_EXT.1.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_AEX_EXT.1.3 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_AEX_EXT.1.4 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_AEX_EXT.1.5 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_API_EXT.1.1 – "*The evaluator shall verify that the TSS lists the platform APIs used in the application."*

The TSS lists in section 8.6.2 the platform APIs used in the TOE application. For each platform API referenced in the TSS, a search was performed to locate the corresponding platform API documentation and verified that it was supported for the Windows platform OS version.

Based on the TSS description identifying a list of APIs that have been confirmed as supported by the Windows platform this assurance activity is satisfied.

FPT_IDV_EXT.1.1 – "*If "other version information" is selected the evaluator shall verify that the TSS contains an explanation of the versioning methodology."*

The TSS states in section 8.6.3 that: "The TOE is versioned using "major.minor.patch.build" 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. Additional labels for pre-release and build metadata are available as extensions to the major version updates."

Based on the TSS description and explanation of the major.minor.patch.build versioning scheme this assurance activity is satisfied.

FPT_LIB_EXT.1.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.1.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.1.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.1.3 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.1.4 – TD0561 - "*The evaluator shall verify that the TSS identifies how updates to the application are signed by an authorized source. The definition of an authorized source must be contained in the TSS. The evaluator shall also ensure that the TSS (or the operational guidance) describes how candidate updates are obtained."*

The TSS states in section 8.6.5 that: "The TOE application is packaged in the installer.exe and msi.exe formats for the Windows OS platform. The application's software is digitally signed using a VMware Carbon Black commercial CA certificate (authorized source) and then cross-signed by Microsoft's WHQL signing. The software is verified by the management server prior to being made available for installation on an endpoint system."

Based on the TSS description identifying that the application is digitally signed by VMware Carbon Black commercial CA (authorized source) and then cross-signed by Microsoft's WHQL signing this assurance activity is satisfied.

FPT_TUD_EXT.1.5 – "*The evaluator shall verify that the TSS identifies how the application is distributed. If "with the platform" is selected the evaluated shall perform a clean installation or factory reset to confirm that TOE software is included as part of the platform OS. If "as an additional package" is selected the evaluator shall perform the tests in FPT_TUD_EXT.2."*

The TSS states in section 8.6.5 that: "The TOE application is packaged in the installer.exe and msi.exe formats for the Windows OS platform." This is consistent with the SFR selection of "as an additional package to the platform OS." Therefore, as per the assurance activity requirement, FPT_TUD_EXT.2, tests were conducted.

Based on the testing output of FPT_TUD_EXT.2 Tests 24 and 25, this assurance activity is considered satisfied.

FPT_TUD_EXT.2.1 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.2.2 – This SFR does not contain any [APP_PP] TSS Assurance Activities.

FPT_TUD_EXT.2.3 – TD0561- *The evaluator shall verify that the TSS identifies how the application installation package is signed by an authorized source. The definition of an authorized source must be contained in the TSS.*

The TSS states in section 8.6.5 that: "The TOE application is packaged in the installer.exe and msi.exe formats for the Windows OS platform. The application's software is digitally signed using a VMware Carbon Black commercial CA certificate (authorized source) and then cross-signed by Microsoft's WHQL signing. The software is verified by the management server prior to being made available for installation on an endpoint system."

Based on the TSS description identifying that the application is digitally signed by VMware Carbon Black commercial CA (authorized source) and then cross-signed by Microsoft's WHQL signing this assurance activity is satisfied.

FTP_DIT_EXT.1.1 – TD0587 – "*For platform-provided functionality, the evaluator shall verify the TSS contains the calls to the platform that TOE is leveraging to invoke the functionality."*

The TSS states in section 8.7.1 that: "The TOE invokes the platform's WinHTTP to establish the trusted channel (HTTPS session over TLS v1.2) between the TOE and the management server. These protocols are used to protect the data traversing the channel from disclosure and/or modification. The platform only acts as a HTTPS/TLS client on behalf of the TOE."

Based on the TSS description identifying WinHTTP as the means in which the TOE invokes the operating system to initiate the trusted channel this assurance activity is considered satisfied.

3 Operational Guidance Assurance Activities

The evaluation team completed the testing of the Operational Guidance, which includes the review of the *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Supplemental Administrative Guidance for Common Criteria* (AGD) document and confirmed that the Operational Guidance contains all Assurance Activities as specified by the *Protection Profile for Application Software Version 1.3* [APP_PP]. The evaluators reviewed the [APP_PP] to identify the security functionality that must be discussed for the operational guidance. This is prescribed by the Assurance Activities for each SFR and the AGD SARs. The evaluators have listed below each of the SFRs defined in the [APP_PP] that have been claimed by the TOE (some SFRs are conditional or optional) as well as the AGD SAR, along with a discussion of where in the operational guidance the associated Assurance Activities material can be found. The AGD includes references to other guidance documents that must be used to properly install, configure, and operate the TOE in its evaluated configuration. The AGD and its references to other VMware Carbon Black guidance documents were reviewed to assess the Operational Guidance Assurance Activities. The AGD contains references to these documents in Chapter 4 and these references can also be found below:

The following references are used in this section of the document:

[1] VMware Carbon Black EDR User Guide, VMware Carbon Black EDR 7.5

[2] VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Supplemental Administrative Guidance for Common Criteria, v1.1, (AGD)

[3] VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Security Target, v1.5 (ST)

FCS_CKM_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FCS_RBG_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FCS_STO_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FDP_DAR_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FDP_DEC_EXT.1.1 – "*The evaluator shall perform the platform-specific actions below and inspect user documentation to determine the application's access to hardware resources. The evaluator shall ensure that this is consistent with the selections indicated. The evaluator shall review documentation provided by the application developer and for each resource which it accesses, identify the justification as to why access is required."*

Section 5.3 of the AGD explains the network resources are required. "The TOE invokes the OS to establish a trusted channel to the management server for polling cycles which includes:

- sending collected host platform system information
- receiving configuration updates and software updates

Section 6.3 of the AGD explains that: "The TOE requires network access to communicate with the operational environment's management server. During startup, the TOE invokes the platform to establish a persistent HTTPS/TLS connection to the management server for the purpose of periodically transmitting the collected information about the endpoint system and retrieve configuration updates."

Based on the AGD description identifying network resources and provides a consistent description of the purpose, in both 5.3 and 6.3, this assurance activity is considered satisfied.

FDP_DEC_EXT.1.2 – "*The evaluator shall perform the platform-specific actions below and inspect user documentation to determine the application's access to sensitive information repositories. The evaluator shall ensure that this is consistent with the selections indicated. The evaluator shall review documentation provided by the application developer and for each sensitive information repository which it accesses, identify the justification as to why access is required."*

Section 5.3 of the AGD describes the sensitive information repositories requirement as. "The VMware CB EDR Windows Sensor collects system events and information from:

- system log files
- memory dumps

Section 6.3 of the AGD describes the sensitive information required. The TSS describes that the TOE "transmits endpoint host OS telemetry data (processes/threads being created, filesystem activity, registry activity, etc.), system logs, and/or memory dumps in response to the data request received from the management server." System logs and memory dumps are considered sensitive repositories because "System logs and memory dumps can contain data that the operational environment could consider sensitive, credentials from failed login attempts, keys that are in memory that are written out to the system or process crashes. Therefore, system logs and memory dumps are considered sensitive data in the eyes of Common Criteria evaluations."

Based on the AGD description identifying sensitive data and provides a consistent description of the purpose, in both 5.3 and 6.3, this assurance activity is considered satisfied.

FDP_NET_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FIA_X509_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FIA_X509_EXT.1.2 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FIA_X509_EXT.2.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FIA_X509_EXT.2.2 – *"If the requirement that the administrator is able to specify the default action, then the evaluator shall ensure that the operational guidance contains instructions on how this configuration action is performed."*

The TOE claims "not accept the certificate" when unable to determine the validity of a certificate. For the TOE to be in the evaluated operational state the server certificate validation mode must be set to "Strict" validation mode. Section 6.1.2 defines how the administrator of the management server must select Strict certificate validation for the Server certificate validation mode using the management server's administrative interface.

Based on the instructions being present in the AGD this assurance activity is considered satisfied.

FMT_CFG_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FMT_CFG_EXT.1.2 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FMT_MEC_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FMT_SMF.1.1 – "*The evaluator shall verify that every management function mandated by the PP is described in the operational guidance and that the description contains the information required to perform the management duties associated with the management function."*

Section 7.1 of the AGD describes user management activities. "Both the local administrator and endpoint user are considered the owner or user of the endpoint device for which the TOE is installed. A typical endpoint user does not have any management functionality. However, the local administrator (Windows OS administrator) can start and stop the "CarbonBlack" service" (hard stopping all transmissions), verifying version, and uninstall the application." The subsections provide the instructions on how to start/stop, verify version, uninstall, and enable/disable system information from being transmitted.

Based on the AGD description covering the security management functions identified in the ST (version check and enable/disable the transmission of any information describing the system's hardware, software, or configuration), this assurance activity is considered satisfied.

FPR_ANO_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_AEX_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_AEX_EXT.1.2 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_AEX_EXT.1.3 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_AEX_EXT.1.4 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_AEX_EXT.1.5 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_API_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_IDV_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_LIB_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_TUD_EXT.1.1 – *"The evaluator shall check to ensure the guidance includes a description of how updates are performed."*

Section 7.2 outlines how to perform an update. The description covers how the enterprise administrator must obtain the update and install the update onto the management server. During the check in cycle with the management server, the TOE is notified that there is an update and that it must download and install the package. The TOE portion of checking, downloading, digital signature validation, and installation of the update requires no administrator intervention when the TOE has been configured for automatic updates. Steps for manual updates are also provided for when the TOE is configured for manual updates only.

FPT_TUD_EXT.1.2 – "*The evaluator shall verify guidance includes a description of how to query the current version of the application."*

Sections 7.1.3 of the AGD describes how to query the current software version using multiple methods: Windows application wizard, file properties, and running `cb.exe -v` command.

Based on the AGD description providing the steps required to queries/validate the TOE's version number this assurance activity is considered satisfied.

FPT_TUD_EXT.1.3 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_TUD_EXT.1.4 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_TUD_EXT.1.5 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_TUD_EXT.2.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FPT_TUD_EXT.2.2 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

FTP_DIT_EXT.1.1 – This SFR does not contain any [APP_PP] AGD Assurance Activities.

4 Test Assurance Activities (Test Report)

The following sections demonstrate that all ATE Assurance Activities for the TOE have been met. This evidence has been presented in a manner that is consistent with the "Reporting for Evaluations Against NIAP-Approved Protection Profiles" guidance that has been provided by NIAP. Specific test steps and associated detailed results are not included in this report in order for it to remain non-proprietary. The test report is a summarized version of the test activities that were performed as part of creating the Evaluation Technical Report (ETR).

The evaluation team conducted testing activities between November 2020 and June 2021. Testing was conducted at the Booz Allen CCTL in Laurel, MD on an isolated network.

4.1 Platforms Tested and Composition

The evaluation team set up a test environment for the independent functional testing that allowed them to perform all test assurance activities across the VMware CB EDR Windows Sensor over the SFR relevant interfaces.

The evaluation team fully tested the TOE installed on a Windows 10 Enterprise 1903 (May 2019 Update) (x64) on an Intel Core i5-8350U with AES-NI and without SHA Extensions processor. This OS processor combination is in compliance with the Microsoft Windows 10 and Server 2019 version 1903 (May 2019 Update) evaluation and its CAVP certificate C785.

There is one CLI for version verification and the remainder of the security administration requires OS Administrative privileges. The full set of tests were developed to stimulate each applicable TSF relevant interface; which would fully test all combinations of the selected models and their TSF relevant interfaces. The testing is consistent with the use of the interfaces defined within the ST. Thus, the testing of the interfaces was based upon testing SFR functionality related to user actions over each interface.

4.1.1 Test Configuration

The evaluation team configured the TOE for testing according to the *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Supplemental Administrative Guidance for Common Criteria* (AGD) document. The evaluation team set up a test environment for the independent functional

testing that allowed them to perform the assurance activities against the TOE over the SFR relevant interfaces.

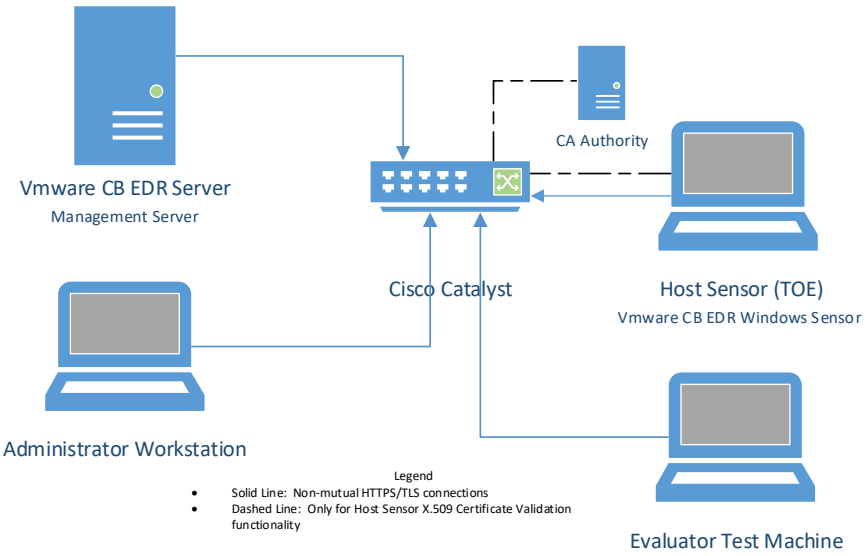

Figure 1 - VMware CB EDR Windows Sensor 7.2 Test Configuration

4.2 Omission Justification

It is expected that the TOE is operating on a Common Criteria certified operating system and platform based on the Microsoft Windows 10 and Server 2019 version 1903 (May 2019 Update) evaluation. The TOE software that is installed on the Windows based OS, is identical (installation, executables, functionality, and features) no matter which variation of the Windows 10 (May 2019 Update) is used. Therefore, equivalency can be claimed for the TOE operating on any of the following 5 MS Windows variants:

- Microsoft Windows 10 Home edition (May 2019 Update) (32-bit and 64-bit versions)
- Microsoft Windows 10 Pro edition (May 2019 Update) (64-bit version)
- Microsoft Windows 10 Enterprise edition (May 2019 Update) (64-bit version)
- Microsoft Windows Server Standard edition, version 1903
- Microsoft Windows Server Datacenter edition, version 1903

Successful completion of all functional tests is sufficient to demonstrate the appropriate behavior of the TSF across the Window's variants.

4.3 Test Cases

The evaluation team completed the functional testing activities within the Booz Allen laboratory environment. The evaluation team conducted a set of testing that includes all ATE Assurance Activities as specified by the *Protection Profile for Application Software Version 1.3* [APP_PP]. The evaluators reviewed the [APP_PP] to identify the security functionality that must be verified through functional testing. This is prescribed by the Assurance Activities for each SFR.
If an SFR is not listed, one of the following conditions applies:

- The Assurance Activity for the SFR specifically indicates that it is simultaneously satisfied by completing a test Assurance Activity for a different SFR.
- The Assurance Activity for the SFR does not specify any actions related to ATE activities or does not meet the conditional statement (e.g., FPT_TUD_EXT.1.4 or FTP_DIT_EXT.1.1).

Note that some SFRs do not have Assurance Activities associated with them at the element level (e.g., FCS_CKM_EXT.1.1). In such cases, testing for the SFR is considered to be satisfied by completion of all Assurance Activities at the component level.

The following lists for each ATE Assurance Activity, the test objective, test instructions, test steps, and test results. Note that unless otherwise specified, the test configuration is to be in the evaluated configuration as defined by the AGD. As part of the cleanup for each test, the TOE is returned to the evaluated configuration.

4.3.1 Windows

The TOE is considered a Windows Classic Desktop application. Only the wording that pertains to Windows Classic Desktop application testing has been reproduced from the PP evaluation activities. Windows Universal Application and .NET references have been removed. A notation has been made to identify when wording has been omitted.

4.3.2 Cryptographic Support

The TOE type for VMWare CB EDR Windows Sensor v7.2 is Application Software. The TOE does not perform any cryptographic functions and completely relies on the underlying OS for all cryptographic support. This includes all trusted channels support and credential storage protection. Therefore, there are no NIST CAVP or CMVP certificates being claimed for this product.

Per Policy 5 Addendum #10 “In cases where the cryptography is platform-provided, how would the evaluator verify the platform satisfies the cryptographic requirements? If the platform has been evaluated and is on the NIAP Product Compliant List (PCL), the evaluator may rely on the Security Target of the evaluated platform to verify the functionality was evaluated.

Therefore, in order to provide assurance of the trusted communications and credentials storage protection, the TOE was installed on the CC certified version Microsoft Windows 10 and Server 2019 version 1903 (May 2019 Update). The Microsoft Windows 10 and Server 2019 evaluation obtained CAVP certificates which are relied on for this evaluation.

FCS_RBG_EXT.1.1

Test Case Number	001A
SFR	FCS_RBG_EXT.1.1 – TD0416
Test Objective	If invoke-platform provided DRBG functionality is selected, the following tests shall be performed: The evaluator shall decompile the application binary using a decompiler suitable for the application (TOE). The evaluator shall search the output of the decompiler to determine that, for each API listed in the TSS, that API appears in the output. If the representation of the API does not correspond directly to the strings in the following list, the evaluator shall provide a mapping from the decompiled text to its corresponding API, with a description of why the API text does not directly correspond to the decompiled text and justification that the decompiled text corresponds to the associated API. The following are the per-platform list of acceptable APIs: For Windows: The evaluator shall verify that rand_s, RtlGenRandom, BCryptGenRandom, or CryptGenRandom API is used for classic desktop applications.

	It is only required that the API is called/invoked, there is no requirement that the API be used directly. In future versions of this document, CryptGenRandom may be removed as an option as it is no longer the preferred API per vendor documentation. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	*NA - The SFR selection in the ST is "use no DRBG functionality". Therefore, this test assurance activity does not apply.*
Test Results	Pass
Execution Method	Manual

Test Case Number	001B
SFR	FCS_RBG_EXT.1.1
Test Objective	TSS AA that is more appropriate in testing while having access to code. *If **use no DRBG functionality** is selected, the evaluator shall inspect the application and its developer documentation and verify that the application needs no random bit generation services.*
Test Instructions	Execute this test per the test steps.
Test Steps	The TOE does not call or have DRBG functionality. ST is declared as using NO DRBG. 1. Perform a static source code analysis. 2. Verify that the TOE does NOT invoke RBG related API. 3. Search through vendor documentation and look for any indication of the need for entropy within the PP scoped functionality.
Test Results	It was confirmed that there were no indication for the need of RBG services - Pass
Execution Method	Manual

4.3.2.1 FCS_STO_EXT.1.1

Test Case Number	002
SFR	FCS_STO_EXT.1.1
Test Objective	For all credentials for which the application invokes platform-provided functionality, the evaluator shall perform the following actions which vary per platform. For Windows: The evaluator shall verify that all certificates are stored in the Windows Certificate Store. The evaluator shall verify that other credentials, like passwords, are stored in the Windows Credential Manager or stored using the Data Protection API (DPAPI). *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.

Test Steps	1. Launch an Administrator Command Prompt. 2. Execute the following command: `mmc` 3. Choose "File -> Add / Remove Snap-in". 4. Double-click on Certificates. 5. Click on Computer account. 6. Click Next. 7. Click on Local computer. 8. Click Finish. 9. Click OK. 10. Examine the certificate store called "Carbon Black". 11. Verify that all TOE certificates are located within this store.
Test Results	It was confirmed that the certificates were stored in the Windows Certificate Store - Pass
Execution Method	Manual

4.3.3 Identification and Authentication

4.3.3.1 FIA_X509_EXT.1

Test Case Number	031
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 1: The evaluator shall demonstrate that validating a certificate without a valid certification path results in the function failing, for each of the following reasons, in turn: • by establishing a certificate path in which one of the issuing certificates is not a CA certificate, • by omitting the basicConstraints field in one of the issuing certificates, • by setting the basicConstraints field in an issuing certificate to have CA=False, • by omitting the CA signing bit of the key usage field in an issuing certificate, and • by setting the path length field of a valid CA field to a value strictly less than the certificate path. The evaluator shall then establish a valid certificate path consisting of valid CA certificates, and demonstrate that the function succeeds. The evaluator shall then remove trust in one of the CA certificates, and show that the function fails.
Test Instructions	Execute this test per the test steps.
Test Steps	**(a) Issuing certificate not a CA certificate:**

	1. Configure the remote entity to present a certificate chain to the TOE such that the issuing certificate of the remote node entity certificate is not a CA certificate. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful. **(b) Omitting the basicConstraints field in one of the issuing certificates:** *NOTE: This is tested in FIA_X509_EXT.1 – Test Case 039.* **(c) Setting the basicConstraints field in an issuing certificate to have CA=False:** *NOTE: This is tested in FIA_X509_EXT.1 – Test Case 040.* **(d) Omitting the CA signing bit of the key usage field in an issuing certificate:** 1. Configure the remote entity to present a certificate chain to the TOE such that the issuing certificate of the remote node entity certificate does not have the CA signing bit in the key usage field. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful. **(e) Setting the path length field of a valid CA field to a value strictly less than the certificate path:** 1. Configure the remote entity to present a certificate chain to the TOE such that the issuing certificate of the remote node entity certificate has a path length value strictly less than the certificate path. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful. **(f1) Valid CA certificates:** 1. Ensure the trusted root CA certificate required to establish a trusted chain of trust against the presented certificates is installed into the TOE's CA trust store. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is successful.

	(f2) Removal of trust in one of the CA certificates: 1. Ensure the trusted root CA certificate required to establish a trusted chain of trust against the presented certificates is removed from the TOE's CA trust store. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity in parts (a), (b), (c), (d), (e), and (f2). The TOE successfully established a TLS connection to the remote entity in part (f1). The observed results corresponded to the expected results. – Pass
Execution Method	Manual

Test Case Number	032
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Configure the remote entity to present a certificate chain to the TOE such that the node entity certificate is expired. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity due to the expired certificate. – Pass
Execution Method	Manual

Test Case Number	033
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created.

	Test 3: The evaluator shall test that the TOE can properly handle revoked certificates-"conditional on whether CRL, OCSP, OCSP Stapling or OCSP Multi-stapling is selected; if multiple methods are selected, then the following tests shall be performed for each method: The evaluator shall test revocation of the node certificate. The evaluator shall also test revocation of an intermediate CA certificate (i.e. the intermediate CA certificate should be revoked by the root CA), if intermediate CA certificates are supported. If OCSP stapling per RFC 6066 is the only supported revocation method, this test is omitted. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the remote server. 2. Cause the TOE to establish a connection to the remote server. 3. Stop capturing packets between the TOE and the remote server. 4. Verify the connection is successful to the remote server. 5. Revoke the remote server certificate. 6. Repeat Steps 1-3. 7. Verify the connection is unsuccessful to the remote server. 8. Unrevoke the remote server certificate. 9. Revoke the intermediate01 CA certificate. 10. Repeat Steps 1-3. 11. Verify the connection is unsuccessful to the remote server.
Test Results	When the node or intermediate CA certificate was revoked, the TOE failed to establish a TLS connection to the remote entity. When none of the certificates were revoked, the TOE successfully established a TLS connection to the remote entity. – Pass
Execution Method	Manual

Test Case Number	034
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 4: If any OCSP option is selected, the evaluator shall configure the OCSP server or use a man-in-the-middle tool to present a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set, and verify that validation of the CRL fails.
Test Instructions	Execute this test per the test steps.

Test Steps	1. Begin capturing packets between the TOE and the remote server. 2. Cause the TOE to establish a connection to the remote server and using a man-in-the-middle-tool, present a OCSP signing certificate to the TOE that does not have the OCSP signing purpose. 3. Stop capturing packets between the TOE and the remote server. 4. Verify the connection is unsuccessful to the remote server.
Test Results	The TOE failed to establish a TLS connection to the remote server due to the OCSP signing purpose missing from the OCSP signing certificate from one of the OCSP responses. – Pass
Execution Method	Manual

Test Case Number	035
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 5: The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the remote entity. 2. Cause the TOE to establish a connection to the remote entity and using a man-in-the-middle-tool, modify any byte in the first eight bytes of the node certificate. 3. Stop capturing packets between the TOE and the remote entity. 4. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity. This was because the presented node certificate was modified such that one of the bytes in the first eight bytes of the certificate caused the certificate to be invalid. – Pass
Execution Method	Manual

Test Case Number	036
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 6: The evaluator shall modify any byte in the last byte of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)

Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the remote entity. 2. Cause the TOE to establish a connection to the remote entity and using a man-in-the-middle-tool, modify any byte in the last byte of the node certificate. 3. Stop capturing packets between the TOE and the remote entity. 4. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity. This was because the presented node certificate was modified such that the last byte in the certificate caused the certificate to be invalid. – Pass
Execution Method	Manual

Test Case Number	037
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the remote entity. 2. Cause the TOE to establish a connection to the remote entity and using a man-in-the-middle-tool, modify any byte in the public key portion of the node certificate. 3. Stop capturing packets between the TOE and the remote entity. 4. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity. This was because the presented node certificate was modified such that the one of the bytes in the certificate's public key caused it to be invalid. – Pass
Execution Method	Manual

Test Case Number	038
SFR	FIA_X509_EXT.1.1 – TD0587
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 8a: (Conditional on support for EC certificates as indicated in FCS_COP.1(3)). The evaluator shall establish a valid, trusted certificate chain

	consisting of an EC leaf certificate, an EC Intermediate CA certificate not designated as a trust anchor, and an EC certificate designated as a trusted anchor, where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain. Test 8b: (Conditional on support for EC certificates as indicated in FCS_COP.1(3)). The evaluator shall replace the intermediate certificate in the certificate chain for Test 8a with a modified certificate, where the modified intermediate CA has a public key information field where the EC parameters uses an explicit format version of the Elliptic Curve parameters in the public key information field of the intermediate CA certificate from Test 8a, and the modified Intermediate CA certificate is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A - Per the test assurance activity, this test is conditional upon support for EC certificates as indicated in FCS_COP.1(3). The Security Target, for FCS_COP.1(3), does not define support for EC certificates. Therefore, this conditional test does not apply.*
Test Results	N/A
Execution Method	Manual

Test Case Number	039
SFR	FIA_X509_EXT.1.2 – TD0495
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 1: The evaluator shall ensure that the certificate of at least one of the CAs in the chain does not contain the basicConstraints extension. The evaluator shall confirm that validation of the certificate path fails (i) as part of the validation of the peer certificate belonging to this chain; and/or (ii) when attempting to add the CA certificate without the basicConstraints extension to the TOE's trust store.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Configure the remote entity to present to the TOE a certificate chain such that the certificate of the CA issuing the remote node entity certificate does not contain the basicConstraints extension. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity because the issuing certificate of the node certificate did not contain the basicConstraints extension. – Pass
Execution Method	Manual

Test Case Number	040
SFR	FIA_X509_EXT.1.2 – TD0495

Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 2: The evaluator shall ensure that the certificate of at least one of the CAs in the chain has the CA flag in the basicConstraints extension not set (or set to FALSE). The evaluator shall confirm that validation of the certificate path fails (i) as part of the validation of the peer certificate belonging to this chain; and/or (ii) when attempting to add the CA certificate with the CA flag not set (or set to FALSE) in the basicConstraints extension to the TOE's trust store.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Configure the remote entity to present to the TOE a certificate chain such that the certificate of the CA issuing the remote node entity certificate has the CA flag not set (i.e. CA=False) in the basicConstraints extension. 2. Begin capturing packets between the TOE and the remote entity. 3. Cause the TOE to establish a connection to the remote entity. 4. Stop capturing packets between the TOE and the remote entity. 5. Verify the connection is unsuccessful.
Test Results	The TOE failed to establish a TLS connection to the remote entity because the issuing certificate of the node certificate had the cA flag in the basicConstraints extension not set. – Pass
Execution Method	Manual

Test Case Number	041
SFR	FIA_X509_EXT.1.2 – TD0495
Test Objective	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. If the application supports chains of length four or greater, the evaluator shall create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. If the application supports a maximum trust depth of two, then a chain with no Intermediate CA should instead be created. Test 3: The evaluator shall construct a certificate path, such that the certificate of the CA issuing the TOE's certificate has the CA flag in the basicConstraints extension set to TRUE. The validation of the certificate path succeeds.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Test Removed per TD0495
Test Results	N/A
Execution Method	Manual

4.3.3.2 FIA_X509_EXT.2

Test Case Number	042
SFR	FIA_X509_EXT.2.2
Test Objective	The evaluator shall perform the following test for each trusted channel:

	Test 1: The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate, and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the operational guidance to determine that all supported administrator-configurable options behave in their documented manner.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE, the remote entity, and the OCSP responder. 2. Cause the TOE to establish a connection to the remote entity. 3. Stop capturing packets between the TOE, the remote entity, and the OCSP responder. 4. Verify the connection to the OCSP responder and remote entity are successful. 5. Disconnect the OCSP responder from the environment. 6. Repeat Steps 1-3. 7. Verify that the connection to the OCSP responder is unsuccessful. 8. Verify that the connection to the remote entity is unsuccessful.
Test Results	The TOE successfully established a TLS connection to the remote entity after receiving responses from the OCSP responder. When the OCSP responder was made unavailable to the TOE, the TOE failed to establish a TLS connection to the remote entity. The observed results corresponded to the expected results. – Pass
Execution Method	Manual

Test Case Number	043
SFR	FIA_X509_EXT.2.2
Test Objective	The evaluator shall perform the following test for each trusted channel: Test 2: The evaluator shall demonstrate that an invalid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity cannot be accepted.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Configure the remote entity to present a certificate chain to the TOE such that the remote server certificate is expired and contains revocation check location information in the AIA field of the certificate. 2. Begin capturing packets between the TOE and the remote server. 3. Cause the TOE to establish a connection to the remote server. 4. Stop capturing packets between the TOE and the remote server. 5. Verify the connection is unsuccessful to the remote server.
Test Results	The TOE failed to establish a TLS connection to the remote server due to the presented node certificate being invalid (i.e. expired). The TOE also performed OCSP checking against this certificate as it contained the OCSP responder URI in the AIA extension, and received a “good” response from the OCSP responder.
Execution Method	Manual

4.3.4 User Data Protection

4.3.4.1 FDP_DEC_EXT.1.1

Test Case Number	003
SFR	FDP_DEC_EXT.1.1 – TD0434
Test Objective	For Windows Desktop Applications the evaluator shall identify in either the application software or its documentation the list of the required hardware resources. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. Examine the guidance documentation. 2. Verify that documentation states that the application restricts its hardware resource access to only the network connectivity.
Test Results	It was confirmed that the hardware resources access declaration for network connectivity restriction was found in Section 5.3 of the AGD - Pass
Execution Method	Manual

4.3.4.2 FDP_DEC_EXT.1.2

Test Case Number	004
SFR	FDP_DEC_EXT.1.2
Test Objective	For Windows Desktop Applications the evaluator shall identify in either the application software or its documentation the list of sensitive information repositories it accesses. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. Examine the software documentation. 2. Verify that the application restricts its sensitive information repository access to only the system logs and memory dumps.
Test Results	It was confirmed that the sensitive information repositories declaration for access to system logs and memory dumps was found in Section 5.3 of the AGD - Pass
Execution Method	Manual

4.3.4.3 FDP_NET_EXT.1.1

Test Case Number	005
SFR	FDP_NET_EXT.1.1
Test Objective	Test 1: The evaluator shall run the application. While the application is running, the evaluator shall sniff network traffic ignoring all non-application associated traffic and verify that any network communications witnessed are documented in the TSS or are user-initiated.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Stop the TOE application services. 2. Begin capturing packets on the machine running the TOE.

	3. Start the TOE application services. 4. The TOE automatically engages in network communication upon launch. a. To continue polling roughly every 30 seconds execute the command from a Windows Command Prompt. 5. Stop capturing packets on the test machine. 6. Verify that any network communications associated with the TOE application are documented in the TSS or are user-initiated.
Test Results	It was confirmed that only the ST declared client interface to the management server was found - Pass
Execution Method	Manual

Test Case Number	006
SFR	FDP_NET_EXT.1.1
Test Objective	Test 2: The evaluator shall run the application. After the application initializes, the evaluator shall run network port scans to verify that any ports opened by the application have been captured in the ST for the third selection and its assignment. This includes connection-based protocols (e.g. TCP, DCCP) as well as connectionless protocols (e.g. UDP).
Test Instructions	Execute this test per the test steps.
Test Steps	1. Stop the TOE application services. 2. Execute the commands to initiate network-based port scan against the TOE's machine. 3. Start the TOE application services. 4. Execute the commands to initiate network-based port scans against the TOE. 5. Inspect the output and compare to determine what ports are opened by the TOE, if any.
Test Results	It was confirmed that only the ST declared client interface to the management server was found. The TOE is a client only and does not open listening ports. - Pass
Execution Method	Manual

4.3.4.4 FDP_DAR_EXT.1.1

Test Case Number	007
SFR	FDP_DAR_EXT.1.1
Test Objective	Evaluation activities (after the identification of the sensitive data) are to be performed on all sensitive data listed that are not covered by FCS_STO_EXT.1. The evaluator shall inventory the filesystem locations where the application may write data. The evaluator shall run the application and attempt to store sensitive data. The evaluator shall then inspect those areas of the filesystem to note where data was stored (if any), and determine whether it has been encrypted. If leverage platform-provided functionality is selected, the evaluation activities will be performed as stated in the following requirements, which vary on a per-platform basis. For Windows: The Windows platform currently does not provide data-at-rest encryption services which depend upon invocation by application developers. The evaluator shall verify that the Operational User Guidance makes the need to activate platform encryption, such as BitLocker or Encrypting File System (EFS), clear to the end user.

Test Instructions	Execute this test per the test steps.
Test Steps	The ST selects "protect sensitive data in accordance with FCS_STO_EXT.1" for this SFR. See FCS_STO_EXT.1 for evidence to support the sensitive data that is covered. *NOTE: The location of where the TOE installs files is determined based on FPT_TUD_EXT.1.3 – Test Case 023 evidence.* *NOTE: The location of where TOE writes file operationally is determined based on FPT_AEX_EXT.1.4 – Test Case 019 evidence.* 1. From an Administrator command prompt, execute the command to confirm that BitLocker Full Disk Encryption is enabled.
Test Results	It was confirmed that BitLocker Full Disk Encryption was enabled - Pass
Execution Method	Manual

4.3.5 Security Management

4.3.5.1 FMT_MEC_EXT.1.1

Test Case Number	008
SFR	FMT_MEC_EXT.1.1 – TD0437 and TD0465
Test Objective	If "invoke the mechanisms recommended by the platform vendor for storing and setting configuration options" is chosen, the method of testing varies per platform as follows: For Windows: For Classic Desktop applications, the evaluator shall run the application while monitoring it with the SysInternals tool ProcMon and make changes to its configuration. The evaluator shall verify that ProcMon logs show corresponding changes to the Windows Registry or C:\ProgramData\ directory. *NOTE: TOE is not a Windows Universal Application package nor is it a .NET application. Therefore, WUA and .NET wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. Using the Windows Registry Editor record the current values of the Carbon Black Registry keys. 2. Execute the tool and begin monitoring the TOE (cb.exe). 3. From the CB EDR server make changes to the TOE configuration by unselecting some of the event collection settings. Then apply by pressing Save Group button. 4. Verify in the tool output that corresponding changes appear with RegSetValue operation set to `0`. 5. Using the Windows Registry Editor verify that changes are made in the registry.
Test Results	It was confirmed that the TOE correctly enforced the new collection policy in the Windows registry which effectively enables/disables what system information is transmitted to the management server. This test is considered satisfied. - Pass
Execution Method	Manual

4.3.5.2 FMT_CFG_EXT.1.1

Test Case Number	009
SFR	FMT_CFG_EXT.1.1
Test Objective	If the application uses any default credentials the evaluator shall run the following tests.

	Test 1: The evaluator shall install and run the application without generating or loading new credentials and verify that only the minimal application functionality required to set new credentials is available.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A – The TOE does not implement any default credentials.*
Test Results	Pass
Execution Method	Manual

Test Case Number	010
SFR	FMT_CFG_EXT.1.1
Test Objective	If the application uses any default credentials the evaluator shall run the following tests. Test 2: The evaluator shall attempt to clear all credentials and verify that only the minimal application functionality required to set new credentials is available.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A – The TOE does not implement any default credentials.*
Test Results	Pass
Execution Method	Manual

Test Case Number	011
SFR	FMT_CFG_EXT.1.1
Test Objective	If the application uses any default credentials the evaluator shall run the following tests. Test 3: The evaluator shall run the application, establish new credentials and verify that the original default credentials no longer provide access to the application.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A – The TOE does not implement any default credentials.*
Test Results	Pass
Execution Method	Manual

4.3.5.3 FMT_CFG_EXT.1.2

Test Case Number	012
SFR	FMT_CFG_EXT.1.2
Test Objective	The evaluator shall install and run the application. The evaluator shall inspect the filesystem of the platform (to the extent possible) for any files created by the application and ensure that their permissions are adequate to protect them. The method of doing so varies per platform. For Windows: The evaluator shall run the SysInternals tools, Process Monitor and Access Check (or tools of equivalent capability, like icacls.exe) for Classic Desktop applications to verify that files written to disk during an application's installation have the correct file permissions, such that a standard user cannot modify the application or its data files. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.

Test Steps	1. Refer to FPT_TUD_EXT.1.3 – Test Case 023, Steps 1 through 5 to determine the set of files created by the TOE application during installation. 2. Inspect filesystem paths created by the installer to determine where the TOE writes its files. 3. Execute the command, specifying the directories based on the inspection performed in Step 3. 4. Verify that each file has the correct file permissions (i.e. standard users cannot modify application or data files).
Test Results	It was confirmed that there were no files with write/modify permissions for standard (non-Security Administrator) users - Pass
Execution Method	Manual

4.3.5.4 FMT_SMF.1.1

Test Case Number	013
SFR	FMT_SMF.1.1
Test Objective	The evaluator shall test the application's ability to provide the management functions by configuring the application and testing each option selected from above. The evaluator is expected to test these functions in all the ways in which the ST and guidance documentation state the configuration can be managed.
Test Instructions	Execute this test per the test steps.
Test Steps	Execute the following two management functions: 1. Enable/disable the transmission of any information describing the system's hardware, software, or configuration *NOTE: Enable/disable the transmission of any information describing the system's hardware, software, or configuration was performed in FMT_MEC_EXT.1.1 – Test Case 008.* 2. Version check *NOTE: Version check was performed in FPT_TUD_EXT.1.1 - Test Case 021, FPT_TUD_EXT.1.2 – Test 022, and FPT_IDV_EXT.1.1 – Test Case 027.*
Test Results	All referenced tests were successfully executed - Pass
Execution Method	Manual

4.3.6 Privacy

4.3.6.1 FPR_ANO_EXT.1.1

Test Case Number	014
SFR	FPR_ANO_EXT.1.1
Test Objective	If require user approval before executing is selected, the evaluator shall run the application and exercise the functionality responsibly for transmitting PII and verify that user approval is required before transmission of the PII.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A – The ST selects "[shall] not transmit PII over a network" for this SFR.*
Test Results	Pass
Execution Method	Manual

4.3.7 Protection of the TSF

4.3.7.1 FPT_API_EXT.1.1

Test Case Number	015
SFR	FPT_API_EXT.1.1
Test Objective	The evaluator shall then compare the list with the supported APIs (available through e.g. developer accounts, platform developer groups) and ensure that all APIs listed in the TSS are supported.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Search Microsoft documentation (MSDN) for each library to determine which category of the Microsoft API to which the library belongs. (this list is considered publicly releasable). 2. Verify that list of Non-proprietary API list matches list in ST.
Test Results	It was confirmed that the discovered API list matches the list in the ST - Pass
Execution Method	Manual

4.3.7.2 FPT_AEX_EXT.1.1

Test Case Number	016
SFR	FPT_AEX_EXT.1.1 – TD0544
Test Objective	The evaluator shall perform either a static or dynamic analysis to determine that no memory mappings are placed at an explicit and consistent address. The method of doing so varies per platform. For those platforms requiring the same application running on two different systems, the evaluator may alternatively use the same device. After collecting the first instance of mappings, the evaluator must uninstall the application, reboot the device, and reinstall the application to collect the second instance of mappings. For Windows: The evaluator shall run the same application on two different Windows systems and run a tool that will list all memory mapped addresses for the application. The evaluator shall then verify the two different instances share no mapping locations. The Microsoft SysInternals tool, VMMap, could be used to view memory addresses of a running application. The evaluator shall use a tool such as Microsoft's BinScope Binary Analyzer to confirm that the application has ASLR enabled.
Test Instructions	Execute this test per the test steps.
Test Steps	**Memory Mapping** 1. Start the TOE on each of the Windows machines. 2. Start VMMAP on each of the Windows machines. 3. In the VMMAP instances, set the process to the TOE's executable and wait for the memory scans to complete. 4. Save the VMMAP results. 5. Compare the results from the two VMMAP scan instances and verify that the two share no mapping locations. **ASLR** 6. Open a Windows PowerShell window on the TOE host platform. 7. Import the module by executing the command. 8. Execute the command for each TOE application file. 9. Verify the tool output contains the statement indicating ASLR is enabled for the TOE application.
Test Results	It was confirmed that there were no TOE controlled common memory mappings between the two machines discovered and the TOE has ASLR enabled - Pass
Execution Method	Manual

4.3.7.3 FPT_AEX_EXT.1.2

Test Case Number	017
SFR	FPT_AEX_EXT.1.2
Test Objective	The evaluator shall verify that no memory mapping requests are made with write and execute permissions. The method of doing so varies per platform. For Windows: The evaluator shall use a tool such as Microsoft's BinScope Binary Analyzer to confirm that the application passes the NXCheck. The evaluator may also ensure that the /NXCOMPAT flag was used during compilation to verify that DEP protections are enabled for the application.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Open a Windows PowerShell window. 2. Import the module by executing the command. 3. Execute the command for each TOE application file. 4. Verify the tool output contains the output indicating that the /NXCOMPAT flag was used during compilation.
Test Results	It was confirmed that the TOE has DEP protections enabled - Pass
Execution Method	Manual

4.3.7.4 FPT_AEX_EXT.1.3

Test Case Number	018
SFR	FPT_AEX_EXT.1.3
Test Objective	The evaluator shall configure the platform in the ascribed manner and carry out one of the prescribed tests: For Windows: If the OS platform supports Windows Defender Exploit Guard (Windows 10 version 1709 or later), then the evaluator shall ensure that the application can run successfully with Windows Defender Exploit Guard Exploit Protection configured with the following minimum mitigations enabled; Control Flow Guard (CFG), Randomize memory allocations (Bottom-Up ASLR), Export address filtering (EAF), Import address filtering (IAF), and Data Execution Prevention (DEP). The following link describes how to enable Exploit Protection, https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-exploit-guard/customize-exploit-protection. If the OS platform supports the Enhanced Mitigation Experience Toolkit (EMET) which can be installed on Windows 10 version 1703 and earlier, then the evaluator shall ensure that the application can run successfully with EMET configured with the following minimum mitigations enabled; Memory Protection Check, Randomize memory allocations (Bottom-Up ASLR), Export address filtering (EAF), and Data Execution Prevention (DEP).
Test Instructions	Execute this test per the test steps.
Test Steps	1. Open a PowerShell window on the server running the TOE application. 2. Execute the following script to set the CFG, Bottom-Up ASLR, EAF, IAF, and DEP protections. 3. Restart the TOE application. 4. Using Process Explorer, obtain the PID of the TOE process as a result of performing Step 3. 5. Execute the command to verify the configuration in Step 2 for each process identifier (PID) of the TOE application: 6. Verify that the TOE application runs successfully with the new Windows Defender Exploit Guard Protection configuration.
Test Results	It was confirmed that TOE ran successfully while the Windows Defender protections were enabled. - Pass
Execution Method	Manual

4.3.7.5 FPT_AEX_EXT.1.4

Test Case Number	019
SFR	FPT_AEX_EXT.1.4 – TD0445
Test Objective	The evaluator shall run the application and determine where it writes its files. For files where the user does not choose the destination, the evaluator shall check whether the destination directory contains executable files. This varies per platform: For Windows: For Windows Desktop Applications the evaluator shall run the program, mimicking normal usage, and note where all user-modifiable files are written. The evaluator shall ensure that there are no executable files stored in the same directories to which the application wrote user-modifiable files. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. From FPT_TUD_EXT.1.3 – Test Case 023 determine where files are installed. 2. Stop "`cb.exe`" using Windows Services manager. 3. Run Windows PowerShell as administrator. 4. Save hash files by executing the commands in the PowerShell Window. 5. Start "`cb.exe`" using Windows Services manager. 6. Allow TOE to operate and initiate the periodic polling several times (approx. 30 sec between cycles). 7. Stop "`cb.exe`" using Windows Services manager. 8. Create new hash files, for comparing to output of Step 4, by executing the commands in the PowerShell Window. 9. Note where, if any, user-modifiable files are written (indicative of a file hash difference or addition of new file). 10. Verify that there are no executable files stored in the same directories to which the application wrote, if any, user-modifiable files (FMT_CFG_EXT.1.2).
Test Results	It was confirmed that no executable files stored in the same directories to which the application wrote user-modifiable files were discovered - Pass
Execution Method	Manual

4.3.7.6 FPT_AEX_EXT.1.5

Test Case Number	020
SFR	FPT_AEX_EXT.1.5
Test Objective	The evaluator will inspect every native executable included in the TOE to ensure that stack-based buffer overflow protection is present. For Windows: For other code, the evaluator shall review the TSS and verify that the /GS flag was used during compilation. The evaluator shall run a tool like, BinScope, that can verify the correct usage of /GS. *NOTE: Code does not run in .NET framework nor is it written in Object Pascal using Delphi IDE. Therefore, the .NET and Object Pascal wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. From FPT_TUD_EXT.1.3 – Test Case 023 determine where executables are installed. 2. Using the tool execute the command. 3. Verify that the code is compiled with the MSVC /GS flag.

	4. Repeat for each TOE executable.
Test Results	It was confirmed that /GS was used when compiling - Pass
Execution Method	Manual

4.3.7.7 FPT_TUD_EXT.1.1

Test Case Number	021
SFR	FPT_TUD_EXT.1.1
Test Objective	The evaluator shall check for an update using procedures described in either the application documentation or the platform documentation and verify that the application does not issue an error. If it is updated or if it reports that no update is available this requirement is considered to be met.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Perform the TOE version check: `cb.exe -v` 2. Publish an update for the TOE on the EDR Server. 3. Wait for the TOE to perform its next sync with the server. 4. Perform the TOE version check: `cb.exe -v` 5. If the version output from Step 4 is a version higher than the output from Step 1, then an update to the TOE was available and it automatically updated. If the version output from Step 4 is the same as the output from Step 1, then the TOE is updated, and no update is available.
Test Results	It was confirmed that the TOE successfully implemented an update and reported the increased version - Pass
Execution Method	Manual

4.3.7.8 FPT_TUD_EXT.1.2

Test Case Number	022
SFR	FPT_TUD_EXT.1.2
Test Objective	The evaluator shall query the application for the current version of the software according to the operational user guidance. The evaluator shall then verify that the current version matches that of the documented and installed version.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Execute the following command from the TOE platform Command Prompt: `C:\Windows\CarbonBlack>cb.exe -v` 2. Examine the version number and verify that it matches the version number defined in the ST and expected installed version.
Test Results	It was confirmed that the version number was correctly displayed - Pass
Execution Method	Manual

4.3.7.9 FPT_TUD_EXT.1.3

Test Case Number	023
SFR	FPT_TUD_EXT.1.3 – TD0548
Test Objective	The evaluator shall verify that the application's executable files are not changed by the application. The evaluator shall complete the following test:

	Test 1: The evaluator shall install the application and then locate all of its executable files. The evaluator shall then, for each file, save off either a hash of the file or a copy of the file itself. The evaluator shall then run the application and exercise all features of the application as described in the ST. The evaluator shall then compare each executable file with the either the saved hash or the saved copy of the files. The evaluator shall verify that these are identical.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Ensure the TOE is not installed on the system. 2. Execute the commands on the system via an Administrator Command Prompt. 3. Install the TOE on the system. 4. Execute the commands on the system via an Administrator Command Prompt. 5. Examine the two filesystem output files for differences between the filesystem before and after the installation of the TOE to determine where the TOE writes its executable files. 6. Stop cb.exe using Windows Services manager. 7. Calculate and save the hashes of the TOE files by executing the command(s) in an Administrator PowerShell window. 8. Restart cb.exe using Windows Services manager. 9. Allow TOE to operate and initiate the periodic polling several times (approx. 30 sec between cycles). 10. Stop cb.exe using Windows Services manager. 11. Calculate and save the hashes of the TOE files by executing the command(s) in an Administrator PowerShell window. 12. Compare the calculated hashes from the output files to confirm that the hash values for each file did not change.
Test Results	It was confirmed that no files changed as a result of exercising the TOE - Pass
Execution Method	Manual

4.3.7.10 FPT_TUD_EXT.2.1

Test Case Number	024
SFR	FPT_TUD_EXT.2.1
Test Objective	The evaluator shall verify that application updates are distributed in the format supported by the platform. This varies per platform: For Windows: The evaluator shall ensure that the application is packaged in the standard Windows Installer (.MSI) format, the Windows Application Software (.EXE) format signed using the Microsoft Authenticode process. See https://msdn.microsoft.com/en-us/library/ms537364(v=vs.85).aspx for details regarding Authenticode signing. *NOTE: TOE is not a Windows Universal Application package. Therefore, WUA wording has been removed to avoid confusion.*
Test Instructions	Execute this test per the test steps.
Test Steps	1. Remotely log into the management server. 2. Select the correct sensor group for the evaluation (Default Group). 3. Click the “Download Sensor Installer” button in upper right corner. 4. Select “Windows Standalone EXE – Latest Version” option. 5. Select “Windows MSI for GPO Installation – Latest Version” option. 6. Unzip the downloads to access individual installation files. 7. Using the tool, open the EXE installer. 8. Verify that the file is in the format for Windows executable files. 9. Execute the command to confirm that the package is digitally signed.

	10. Using the tool, open the MSI installer. 11. Verify that the file is in the format for a Microsoft Installer package file. 12. Execute the command to confirm that the package is digitally signed.
Test Results	It was confirmed that the TOE is packaged in EXE and MSI format - Pass
Execution Method	Manual

4.3.7.11 FPT_TUD_EXT.2.2

Test Case Number	025
SFR	FPT_TUD_EXT.2.2
Test Objective	For All Other Platforms: The evaluator shall record the path of every file on the entire filesystem prior to installation of the application, and then install and run the application. Afterwards, the evaluator shall then uninstall the application, and compare the resulting filesystem to the initial record to verify that no files, other than configuration, output, and audit/log files, have been added to the filesystem.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Ensure the TOE is not installed on the system. 2. Execute the commands on the system via an Administrator Command Prompt. 3. Install the TOE on the system. 4. Launch the TOE application. 5. Allow TOE to operate and initiate the periodic polling several times (approx. 30 sec between cycles). 6. Uninstall the TOE application. 7. Execute the commands on the system via an Administrator Command Prompt. 8. Examine the two filesystem output files for differences between the filesystem before the installation of the TOE and after the uninstallation of the TOE to verify that no files, other than configuration, output and audit/log files have been added to the filesystem.
Test Results	It was confirmed that only permitted audit log files remained on platform after TOE uninstallation - Pass
Execution Method	Manual

4.3.7.12 FPT_LIB_EXT.1.1

Test Case Number	026
SFR	FPT_LIB_EXT.1.1
Test Objective	The evaluator shall install the application and survey its installation directory for dynamic libraries. The evaluator shall verify that libraries found to be packaged with or employed by the application are limited to those in the assignment.
Test Instructions	Execute this test per the test steps.
Test Steps	1. From the evidence of FPT_TUD_EXT.1.3 – Test Case 023 determine where executables are installed. 2. Examine the installation directory and attempt to locate dynamic libraries installed by the TOE. 3. Verify that the installed libraries correspond to those documented in the ST. (The declared libraries in the ST were statically compiled into the TOE meaning there will be no dynamic libraries found)
Test Results	It was confirmed that there were no third-party dynamic libraries discovered installed by the TOE - Pass
Execution Method	Manual

4.3.7.13 FPT_IDV_EXT.1.1

Test Case Number	027

SFR	FPT_IDV_EXT.1.1
Test Objective	The evaluator shall install the application, then check for the / existence of version information. If SWID tags is selected the evaluator shall check for a .swidtag file. The evaluator shall open the file and verify that is contains at least a SoftwareIdentity element and an Entity element.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install the TOE application. 2. Execute the following command from an Administrator Command Prompt: `C:\Windows\CarbonBlack>cb.exe -v`
Test Results	It was confirmed that the version was displayed in the declared Major#.Minor#.Patch#.Build# format - Pass
Execution Method	Manual

4.3.8 Trusted Path/Channel

4.3.8.1 FTP_DIT_EXT.1.1

Test Case Number	028
SFR	FTP_DIT_EXT.1.1 – TD0587
Test Objective	Test 1: The evaluator shall exercise the application (attempting to transmit data; for example by connecting to remote systems or websites) while capturing packets from the application. The evaluator shall verify from the packet capture that the traffic is encrypted with HTTPS, TLS, DTLS, SSH, or IPsec in accordance with the selection in the ST.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets on the TOE host platform. 2. The TOE engages in network communication to the Management Server every 30 seconds without any intervention. 3. Stop capturing packets. 4. Verify that network traffic between the TOE and the Management Server is encrypted using HTTPS.
Test Results	It was confirmed that HTTPS was being used to communicate with the management server - Pass
Execution Method	Manual

Test Case Number	029
SFR	FTP_DIT_EXT.1.1 – TD0587
Test Objective	Test 2: The evaluator shall exercise the application (attempting to transmit data; for example by connecting to remote systems or websites) while capturing packets from the application. The evaluator shall review the packet capture and verify that no sensitive data is transmitted in the clear.
Test Instructions	Execute this test per the test steps.
Test Steps	*NOTE: This testing activity is met by testing performed in FTP_DIT_EXT.1.1 – Test Case 028.*
Test Results	Referenced test was successfully executed - Pass
Execution Method	Manual

Test Case Number	030
SFR	FTP_DIT_EXT.1.1 – TD0587

Test Objective	Test 3: The evaluator shall inspect the TSS to determine if user credentials are transmitted. If credentials are transmitted the evaluator shall set the credential to a known value. The evaluator shall capture packets from the application while causing credentials to be transmitted as described in the TSS. The evaluator shall perform a string search of the captured network packets and verify that the plaintext credential previously set by the evaluator is not found.
Test Instructions	Execute this test per the test steps.
Test Steps	*N/A – The TOE does not transmit user credentials.*
Test Results	Pass
Execution Method	Manual

5 Evaluation Activities for SARs

This section addresses assurance activities that are defined in the *Protection Profile for Application Software Version 1.3* [APP_PP] that correspond with Security Assurance Requirements.

ADV_FSP.1 – *"There are no specific evaluation activities associated with these SARs, except ensuring the information is provided. The functional specification documentation is provided to support the evaluation activities described in 5.1 Security Functional Requirements, and other activities described for AGD, ATE, and AVA SARs. The requirements on the content of the functional specification information is implicitly assessed by virtue of the other evaluation activities being performed; if the evaluator is unable to perform an activity because there is insufficient interface information, then an adequate functional specification has not been provided."*

Section 1.3 of the Security Target describes the following Supporting Environment Interfaces:

- **TOE Application to OS (1) –** The TOE leverages operating system callbacks in order to collect system-relevant data information, store log files, access Windows key store, and invoke network access.
- **TOE Platform to Management Server (2) –** The TOE invokes the operating system to use HTTPS/TLS to establish a trusted communication channel to the management server for transmitting the collected data files, retrieving configuration updates, and obtaining software updates. The TOE platform (OS) is the HTTPS/TLS client for this interface and performs X.509 certificate validation in support of the non-mutually authenticated HTTPS/TLS communications.
- **TOE Platform to CA Authority (3) –** The TOE platform (OS) performs X.509 certificate validation in support of the non-mutually authenticated HTTPS/TLS communications to the Management Server.

The purpose of each TSFI is understood based on the descriptions presented in the ST. In addition to this, the mapping between logical TSFIs and physical interfaces to the application are consistent with the evaluation team's understanding of what each TSFI is used for. Therefore, this assurance activity is considered satisfied.

ADV_FSP.1-3 – *"The evaluator shall check the interface documentation to ensure it identifies and describes the parameters for each TSFI that is identified as being security relevant."*

Section 8.4.3 of the ST and Section 7 of the AGD identifies the security management interfaces. The only interface provided as part of the TOE is the version verification using the cb.exe -v command and must be accomplished as an OS administrator. For all other administrative functions, the OS administrator must use OS services command to start and stop the VMware CB EDR Windows Sensor service, appwiz.cpl to force an uninstallation of the TOE, and identifies additional version verification methods using OS interfaces. The AGD provides instructions for the use of the OS commands for managing the TOE. The TOE provides no additional user interfaces. Therefore, this assurance activity is considered satisfied.

ADV_FSP.1-5 – *"The evaluator shall examine the interface documentation to develop a mapping of the interfaces to SFRs."*

The descriptions provided for each SFR in the TSS Section of the Security Target adequately describe a mapping of any relevant interfaces to that SFR. For example, Sections 8.2.3 (Network Communications) and 8.4.3 (Management Functions), 8.6.5 (Installation and Update), and 8.7.1 (Trusted Path/Channel) in the Security Target describe SFRs that map to interface 2 where the TOE communicates with the management server. Section 8.1.3 (Key Storage), Section 8.6.1 (API list), and Section 8.6.4 (Library list) in the Security Target describe SFRs that map to interface 1 where the TOE interfaces with the OS. . Therefore, this assurance activity is considered satisfied.

AGD_OPE.1 – *"Some of the contents of the operational guidance will be verified by the evaluation activities in 5.1 Security Functional Requirements and evaluation of the TOE according to the [CEM]. The following additional information is also required."*

The TOE comes with its own administrative manual that clearly identifies the version of the TOE. When an end user purchases the TOE, they are given customer portal credentials for the pulling down of documentation and updates to ensure the user has access to the latest information. The *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Supplemental Administrative Guidance for Common Criteria (AGD)* document contains configuration instructions for placing the TOE in its evaluated configuration. Additionally, as part of the CC certification process, the AGD is published on the NIAP web site supplementing the vendor guidance documentation. Therefore, there is a reasonable guarantee that administrators and users are aware of this documentation due to its listing on the Product Complaint List (PCL) in conjunction with the certified product.

"If cryptographic functions are provided by the TOE, the operational guidance shall contain instructions for configuring the cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE."

Section 6.2 of the AGD explains that cryptographic services are invoked from the underlying Windows OS platform. This includes DRBG functionality, HTTPS/TLS trusted communications, and sensitive data encryption storage. It is also notated in this section that the use of other cryptographic engines and cryptographic settings were not evaluated nor tested during the Common Criteria evaluation of the TOE.

"The documentation must describe the process for verifying updates to the TOE by verifying a digital signature – this may be done by the TOE or the underlying platform.
The evaluator shall verify that this process includes the following steps:"

- *Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).*
- *Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes generation of the digital signature. The TOE will likely contain security functionality that does not fall in the scope of evaluation under this PP. The operational guidance shall make it clear to an administrator which security functionality is covered by the evaluation activities."*

Section 7.2 of the AGD describes the process of the TOE receiving updates from the management server during a regular (every 30 second) polling cycle. To maintain security throughout its lifecycle, the TOE provides a mechanism to apply software updates automatically or an update can be installed following the manual installation instructions. In the evaluated configuration, both the manual and automatic updates procedures were successfully followed.

High severity issues can result in a patch release as soon as remediation is available. Lower severity issues will be incorporated into the next monthly release. Security fixes will be released as new packages in the same manner as any feature updates. 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.

There are steps detailed in this section for an administrator on the management server to establish the TOEs update policy (automatic or manual) as well as steps for the endpoints OS administrator to manually obtain and install the update.

Sections 6.1.4 and 7.2 of the AGD explain in detail how the updated TOE version can be verified. Section 7.2 also explains how the update is obtained by the endpoint system (through polling cycle or manual installation by host administrator). The update package when executed invokes the OS APIs to validate the certificate chain (WinVerifyTrust) and install the update correctly.

This clearly demonstrates how the TOE can be updated and verified by the local administrator. Therefore, this assurance activity is considered satisfied.

AGD_PRE.1 – *"As indicated in the introduction above, there are significant expectations with respect to the documentation—especially when configuring the operational environment to support TOE functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for the TOE in the ST."*

Section 5.1 of the AGD describes the TOE components in the evaluated configuration VMware Carbon Black EDR Windows Sensor 7.2 on Common Criteria certified Windows 10 OS (See "*Microsoft Windows Common Criteria Evaluation Microsoft Windows 10 version 1903 (May 2019 Update) Microsoft Windows Server version 1903 (May 2019 Update) Security Target Version 0.04 July 19, 2019*". Section 7 of the AGD contains instructions for the Security Administrator to ensure that the operational environment will fulfill its role in supporting the TOE. These instructions match the assumptions for the TOE's operational environment in Section 4.3 of the ST. Therefore, this assurance activity is considered satisfied.

ALC_CMC.1 – *"The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. Further, the evaluator shall check the AGD guidance and TOE samples received for testing to ensure that the version number is consistent with that in the ST. If the vendor maintains a web site advertising the TOE, the evaluator shall examine the information on the web site to ensure that the information in the ST is sufficient to distinguish the product."*

The evaluation team verified that the Security Target (ST), TOE, and Supplemental Administrative Guidance (AGD) were labeled consistently to correctly identify the operational environment and TOE hardware and software versions in the CC evaluation.

Specifically, Section 1.2 of the ST states in the TOE Reference that the TOE is the VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 (VMware CB EDR Windows Sensor). Section 2 of the AGD states that the AGD is intended for administrators responsible for installing, configuring, and/or operating the VMware CB EDR Windows Sensor 7.2. Finally, the product web site, https://www.carbonblack.com/products/edr/ contains identifying product information including a demo, datasheet, infographics, and descriptions of the product's capabilities. The ST states in the TOE Overview that, "The TOE is an enterprise software application that gathers event data on the endpoints and invokes the OS to securely transmit this information to the operating environment's management server for centralized storage and indexing."

All of this information as stated above provides sufficient context to accurately identify the TOE as such in the ST, AGD, and vendor web site. Therefore, this assurance activity is considered satisfied.

ALC_CMS.1 – *"The "evaluation evidence required by the SARs" in this PP is limited to the information in the ST coupled with the guidance provided to administrators and users under the AGD requirements. By ensuring that the TOE is specifically identified and that this identification is consistent in the ST and in the AGD guidance (as done in the evaluation activity for ALC_CMC.1), the evaluator implicitly confirms the information required by this component. Life-cycle support is targeted aspects of the developer's life-cycle and instructions to providers of applications for the developer's devices, rather than an in-depth examination of the TSF manufacturer's development and configuration management process. This is not meant to diminish the critical role that a developer's practices play in contributing to the overall trustworthiness of a product; rather, it's a reflection on the information to be made available for evaluation.*

The evaluator shall ensure that the developer has identified (in guidance documentation for application developers concerning the targeted platform) one or more development environments appropriate for use in developing applications for the developer's platform. For each of these development environments, the developer shall provide information on how to configure the environment to ensure that buffer overflow protection mechanisms in the environment(s) are invoked (e.g., compiler flags). The evaluator shall ensure that this documentation also includes an indication of whether such protections are on by default, or have to be specifically enabled. The evaluator shall ensure that the TSF is uniquely identified (with respect to other products from the TSF vendor), and that documentation provided by the developer in association with the requirements in the ST is associated with the TSF using this unique identification."

The TOE is compiled with the following flags to ensure anti-exploitation capabilities are enabled for address space layout randomization (ASLR), Data Execution Prevention (DEP), and buffer overflow protection.

Flag	Description	cb.exe
/DYNAMICBASE (Use address space layout randomization)	This linker option enables the building of an executable image that can be loaded at different locations in memory at the beginning of execution. This option also makes the stack location in memory much less predictable.	Yes
/GS (Buffer Security Check)	Instructs the compiler to insert overrun detection code into functions that are at risk of being exploited. When an overrun is detected, execution is stopped. By default, this option is on.	Yes
/guard (Enable Control Flow Guard)	Causes the compiler to analyze control flow for indirect call targets at compile time, and then to insert code to verify the targets at runtime.	Yes
/HIGHENTROPYVA	Specifies whether the executable image supports high-entropy 64-bit address space layout randomization (ASLR).	Yes
/NXCOMPAT, /NXCOMPAT (Compatible with Data Execution Prevention)	These compiler and linker options enable Data Execution Prevention (DEP) compatibility. DEP guards the CPU against the execution of non-code pages.	Yes

The TOE has a unique identifier under ALC_CMC.1, which is the VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2. This included a review of the TSF vendor's website to determine that the identifier was enough to distinguish the TOE from other products from the TSF vendor. The evaluation team also reviewed the following documentation provided by the vendor and confirmed that this identifier was consistently used to reference the TOE:

- VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Security Target, Version 1.5
- VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Supplemental Administrative Guidance for Common Criteria, Version 1.1
- VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor software 7.2 (full build number : 7.2.0.17354)

Therefore, this assurance activity is considered satisfied.

ALC_TSU_EXT.1 – "*The evaluator shall verify that the TSS contains a description of the timely security update process used by the developer to create and deploy security updates. The evaluator shall verify that this description addresses the entire application. The evaluator shall also verify that, in addition to the*

TOE developer's process, any third-party processes are also addressed in the description. The evaluator shall also verify that each mechanism for deployment of security updates is described.

The evaluator shall verify that, for each deployment mechanism described for the update process, the TSS lists a time between public disclosure of a vulnerability and public availability of the security update to the TOE patching this vulnerability, to include any third-party or carrier delays in deployment. The evaluator shall verify that this time is expressed in a number or range of days.

The evaluator shall verify that this description includes the publicly available mechanisms (including either an email address or website) for reporting security issues related to the TOE. The evaluator shall verify that the description of this mechanism includes a method for protecting the report either using a public key for encrypting email or a trusted channel for a website."

The evaluation team verified that the Security Target (ST) contains in the TSS section for FPT_TUD_EXT.1 (Section 8.6.5.1) a description of the timely security update process used by the developer to create and deploy 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 installation package contains all third-party components that are required. The end customer should never attempt to update the third-party packages. 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."

This adequately addresses the entire application, including that there are no third-party update processes to consider when updating the TOE. It also describes the process by which security updates are retrieved, a specific timeframe (in days) for which reported flaws are expected to be addressed, and that the Customers are notified using the Carbon Black website reporting mechanism when fixes are available. Therefore, this assurance activity is considered satisfied.

ATE_IND.1 – "*The evaluator shall prepare a test plan and report documenting the testing aspects of the system, including any application crashes during testing. The evaluator shall determine the root cause of any application crashes and include that information in the report. The test plan covers all of the testing actions contained in the [CEM] and the body of this PP's evaluation activities.*

While it is not necessary to have one test case per test listed in an evaluation activity, the evaluator must document in the test plan that each applicable testing requirement in the ST is covered. The test plan identifies the platforms to be tested, and for those platforms not included in the test plan but included in the ST, the test plan provides a justification for not testing the platforms. This justification must address the differences between the tested platforms and the untested platforms, and make an argument that the differences do not affect the testing to be performed. It is not sufficient to merely assert that the differences have no effect; rationale must be provided. If all platforms claimed in the ST are tested, then no rationale is necessary. The test plan describes the composition of each platform to be tested, and any setup that is necessary beyond what is contained in the AGD documentation. It should be noted that the evaluator is expected to follow the AGD documentation for installation and setup of each platform either as part of a test or as a standard pre-test condition. This may include special test drivers or tools. For each driver or tool, an argument (not just an assertion) should be provided that the driver or tool will not adversely affect the performance of the functionality by the TOE and its platform.

This also includes the configuration of the cryptographic engine to be used. The cryptographic algorithms implemented by this engine are those specified by this PP and used by the cryptographic protocols being evaluated (e.g SSH). The test plan identifies high-level test objectives as well as the test procedures to be followed to achieve those objectives. These procedures include Expected Results.

The test report (which could just be an annotated version of the test plan) details the activities that took place when the test procedures were executed, and includes the actual results of the tests. This shall be a cumulative account, so if there was a test run that resulted in a failure; a fix installed; and then a successful re-run of the test, the report would show a "fail" and "pass" result (and the supporting details), and not just the "pass" result."

The evaluation team created a Detailed Test Report (DTR) to address all aspects of this requirement. The DTR is made up of the proprietary *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Test Plan* and the *VMware_CB_EDR_Win_Sensor_AppPPv1.3_Test_Matrix.xls.* The DTR discusses the test facility, environment, configuration, test tools, equivalency argument, test cases, test procedures, expected results, identification of evidence collected, and analysis of test results. The evaluator's test environment diagram is located in section entitled Test Environment of the *VMware Carbon Black Endpoint Detection and Response (EDR) Windows Sensor 7.2 Test Plan* document. Section 4 of this document presents a public releasable summary of the testing activity per SFR accomplished during testing. Therefore, this assurance activity is considered satisfied.

AVA_VAN.1 – TD0554 *"The evaluator shall generate a report to document their findings with respect to this requirement. This report could physically be part of the overall test report mentioned in ATE_IND, or a separate document. The evaluator performs a search of public information to find vulnerabilities that have been found in similar applications with a particular focus on network protocols the application uses and document formats it parses.*

The evaluator documents the sources consulted and the vulnerabilities found in the report.

For each vulnerability found, the evaluator either provides a rationale with respect to its non-applicability, or the evaluator formulates a test (using the guidelines provided in ATE_IND) to confirm the vulnerability, if suitable. Suitability is determined by assessing the attack vector needed to take advantage of the vulnerability. If exploiting the vulnerability requires expert skills and an electron microscope, for instance, then a test would not be suitable and an appropriate justification would be formulated.

***For Windows, Linux, macOS and Solaris**: The evaluator shall also run a virus scanner with the most current virus definitions against the application files and verify that no files are flagged as malicious."*

The evaluation team reviewed vendor documentation, formulated hypotheses, performed vulnerability analysis, and documented the hypotheses and analysis in accordance with the [APP_PP] requirements. Keywords were identified based upon review of the Security Target and AGD. The following keywords were identified:

Keyword	Description
CarbonBlack	This is a generic term for searching for known vulnerabilities produced by the company as a whole. Company has been referred to as both CarbonBlack (one word) and Carbon Black (two words). This key word will also obtain any information on the full product name. The terms: EDR, EDR Sensor, CB Sensor, Response Sensor, Windows, and version number will be used as a further delimiter on any found responses.
Carbon Black	This is a generic term for searching for known vulnerabilities produced by the company as a whole. This key word will also obtain any information on the full product name. The terms: EDR, EDR Sensor, CB Sensor, Response Sensor, Windows, and version number will be used as a further delimiter on any found responses.
Endpoint Detection	TOE application software technology type. The use of Response will be used as a further delimiter on any found responses.
Google protobuf 2.6.1	Third-party library that is included in the installed TOE software in order to function. Version number used as further delimiter on any found responses.

Keyword	Description
zlib 1.2.11	Third-party library that is included in the installed TOE software in order to function. Version number used as further delimiter on any found responses.

These keywords were used individually and as part of various permutations and combinations to search for vulnerabilities on public vulnerability sources (updated July 21, 2021). The following public vulnerability sources were searched:

- Common Vulnerabilities and Exposures: https://www.cvedetails.com/vulnerability-search.php
- NIST National Vulnerabilities Database (can be used to access CVE and US-CERT databases identified below): https://web.nvd.nist.gov/view/vuln/search
- US-CERT: http://www.kb.cert.org/vuls/html/search
- SecurITeam Exploit Search: www.securiteam.com
- Tipping Point Zero Day Initiative http://www.zerodayinitiative.com/advisories
- Offensive Security Exploit Database: https://www.exploit-db.com
- Rapid7 Vulnerability Database: https://www.rapid7.com/db/vulnerabilities
- Security Focus: http://www.securityfocus.com/vulnerabilities/
- Vendor site https://community.carbonblack.com

In summary, there were no open vulnerabilities found to be applicable to this evaluation.

Upon the completion of the vulnerability analysis research, the team had identified generic vulnerabilities upon which to build a test suite. These tests were created specifically with the intent of exploiting these vulnerabilities within the TOE or its configuration. Testing that was conducted under the functional testing that would have been duplication of a vulnerability tests were not re-run. This left one remaining exploit to further explore: malicious binary.

The team tested the following areas: Fill in from actual VAN

- Virus Scan
  This test scans the TOE binary with a virus scanner using the most current virus definitions against the application files and then the evaluator verifies that no files are flagged as malicious.

The evaluation team determined that no residual vulnerabilities exist that are exploitable by attackers with Basic Attack Potential. Therefore, this assurance activity is considered satisfied.

5.1 Conclusions

The TOE was evaluated against the ST and has been found by this evaluation team to be conformant with the ST. The overall verdict for this evaluation is: Pass.

6 Glossary of Terms

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 application's TSS by the ST author.
Trusted Channel	An encrypted connection between the TOE's host platform 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.

Table 7-1: Terminology

Term	Definition
Local Administrator	The Windows OS administrator who has system permissions to access sensitive data and perform management functionality on the endpoint system.
Management Server	Operational environment server that is used for the remote management of VMware CB EDR Windows Sensors (TOE) by an enterprise administrator. This is a separate product called VMware Carbon Black EDR Server (VMware CB EDR Server) and is not part of the TOE boundary.
Endpoint System	A device or set of devices, such as a laptop or desktop, with the Windows operating system that hosts the TOE.
Endpoint User	An individual who has access to the TOE but is not able to manage its behavior.
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 7-2: Vendor Terminology

Acronym	Definition
API	Application Programming Interface
ASLR	Address Space Layout Randomization
CA	Certificate Authority
CB	Carbon Black
CC	Common Criteria
CLI	Command Line Interface
DEP	Data Execution Prevention
HTTPS	Hyper Text Transfer Protocol Secure
IT	Information Technology
NIAP	National Information Assurance Partnership
OS	Operating System
PII	Personally Identifiable Information
PP	Protection Profile
DRBG	Deterministic Random Bit Generator
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function

Table 7-3: Acronyms