Symantec Endpoint Protection

Security Target

Acumen Security, LLC.

Table Of Contents

Revision History

Version	Date	Description
1.0	September 2018	Initial Release
1.1	November 2018	Updated per review
1.2	November 2018	Updated per ECR review

1 Security Target Introduction

1.1 Security Target and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Category	Identifier
ST Title	Symantec Endpoint Protection Security Target
ST Version	1.2
ST Date	November 2018
ST Author	Acumen Security, LLC.
TOE Identifier	Symantec Endpoint Protection (SEP)
TOE Software Version	Version 14.2
TOE Developer	Symantec Corporation
Key Words	Endpoint Security, Client, Application, Cyber Defense

Table 1 TOE/ST Identification

1.2 TOE Overview

The Symantec Endpoint Protection client (hereafter referred to as the TOE or SEP) is a multifaceted endpoint threat control agent blending features of traditional antivirus, HIDS, host-based firewalls, etc., into a single software package.

The SEP comprises a set of applications (.exe) and libraries (.dll), written in C++, running as native code on the operating system. It is composed of components which run in user space (the traditional "application"), as well as service providers which run in privileged mode in kernel space, essentially as drivers, to allow the software to control security-relevant functionality on the host operating system, such as blocking network traffic to malicious hosts, and shutting down host access to removable media.

The platform for this evaluation will be the Windows Operating System.

1.3 TOE Architecture

1.3.1 Physical Boundaries

1.3.1.1 Hardware

The TOE is a software-only evaluation running on a Windows OS platform. The following minimum requirements are needed for the underlying platform to ensure the TOE functions as required:

- Operating System
  - Windows 10
- Processor
  - 32-bit processor: 1 GHz Intel Pentium III or equivalent minimum (Intel Pentium 4 or equivalent recommended)
  - 64-bit processor: 2 GHz Pentium 4 with x86-64 support or equivalent minimum
- Physical RAM
  - 512MB (1GB recommended)
- Hard Drive
  - 395 MB (Additional 135MB required during installation)

1.3.1.2 Software

The software boundary of the TOE incudes the Symantec Endpoint Protection Client application as well as the Graphical User Interface (GUI). For cryptographic operations, the TOE uses the Windows built-in TLS v1.2 implementation in support of HTTPS/TLS communications.

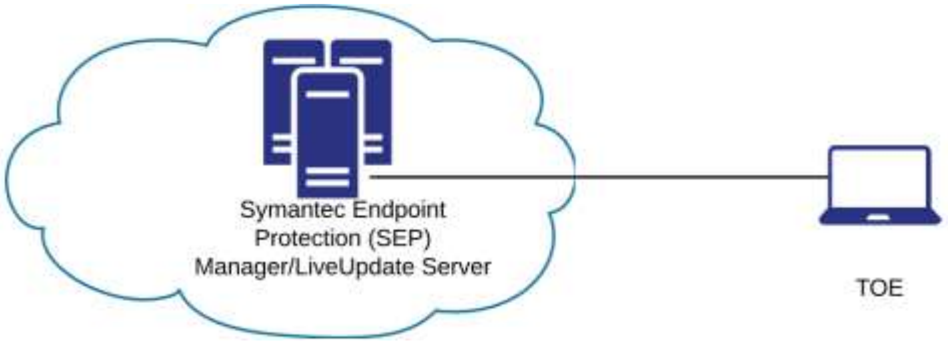
1.3.1.3 Operational Environment

In support of the TOE, the following components are present within the Operational Environment:

Component	Usage/Purpose
Symantec Endpoint Protection (SEP) Manager	SEP Manager maintains the authenticated user accounts and information regarding how it itself authenticates to databases. The SEPM provides management over the Endpoint client configuration.
LiveUpdate Server	LiveUpdate provides administrators a method in which to download definitions, signatures, and other content and distributes the updates to client computers. The connection is secured via TLS.

Table 2 Operational Environment Components

The following diagram shows a typical TOE deployment. The TOE is deployed throughout the network and communicates with the Symantec Endpoint Protection (SEP) Manager/LiveUpdate Server on a corporate network.


Symantec Endpoint Protection (SEP) Manager/LiveUpdate Server
TOE

Notes regarding this above diagram:

- The TOE is running on a Windows 10 PC
- The diagram shows a connection to a Symantec Endpoint Protection (SEP) Manager and a LiveUpdate Server. These services may or may not be residence on the same device.

1.3.2 Logical Boundaries

The TOE provides the security functionality required by [ASPP].

1.3.2.1 Cryptographic Support

The TOE leverages the Windows built-in TLS v1.2 implementation. When establishing a session over TLS, the Windows built-in TLS v1.2 ensures the identifier presented in the exchange matches the correct reference identifier before proceeding with the connection. The Windows built-in TLS v1.2 also performs validation of TLS server certificates. If for any reason during session establishment the validity of a certificate cannot be performed successfully, the Windows built-in TLS v1.2 will not accept the certificate or establish the session. The TOE does not use any DRBG functionality for its cryptographic operations.

1.3.2.2 User Data Protection

In the evaluated configuration, the TOE does not store sensitive data on the drive. In addition, the TOE is restricted to use of only the underlying platforms network connectivity for client/server communications and content updates. These are triggered either by user action or via response to a SEP Manager request. While the TOE writes to the Windows event logs, it does not provide functionality to read the generated events.

1.3.2.3 Identification and Authentication

The TOE supports use of X.509 certificates for TLS communication between the TOE and SEP Manager. This is performed via the X509TrustManager.

1.3.2.4 Security Management

The TOE does not install with any default credentials and does not store any credentials on the system. The authentication mechanisms of the underlying platform are used to ensure only authorized users of that platform can gain access to the application and underlying platform functionality.

Configuration options are stored via native mechanisms (Windows Registry) and proprietary secure storage. Protection of these configuration options is provided using Access Control Lists (ACLs) and SymProtect (Symantec Tamper Protection). By default, the application is configured with file permissions which protect it and its data from unauthorized access

1.3.2.5 Privacy

In the evaluated configuration, the TOE does not transmit any Personally Identifiable Information (PII) over the network.

1.3.2.6 Protection of the TSF

In the evaluated configuration, the TOE does not request memory mapping to any explicit address. However, the TOE does request allocation of memory regions for write and execute permissions. This allocation is performed using PAGE_EXECUTE_READWRITE. It is important to note that the application does not provide the user with the ability to write modifiable files to directories containing executable files.

The TOE is compiled with use of the GS flag to provide protection against stack-based buffer overflow. This provides buffer security checks during compilation of code by checking for risks such as buffer overruns on return addresses and potentially vulnerable parameters.

For updates to the TOE, SEP client implements its own functionality (LiveUpdate) to check for updates which are distributed as MSI files on the Windows platform. TOE updates are digitally signed for image validation. Checking of the software version can be performed through the TOE's GUI as well as using the SWID tags provided with the application. Additional updates to the MSI include content updates and security updates which can be used to update the binary code to ensure up-to-date protection. If the application is uninstalled from the platform, all traces of the application will be purged from the platform.

For the TOE to function as defined within the protection profile, Windows Defender should be disabled on the underlying platform.

1.3.2.7 Trusted Path/Channels

During operation of the TOE, transmitted data is encrypted via HTTPS and TLSv1.2. TLS communication is provided via the Windows built-in TLS v1.2. LiveUpdate, the service used for transmission of security definitions, are sent via HTTPS.

1.3.3 TOE Documentation

- [ST] Symantec Endpoint Protection, Version 14.2 Security Target
- [AGD] Symantec Endpoint Protection Installation and Administration Guide

1.3.4 Other References

Protection Profile for Application Software, version 1.2, dated, 25 April 2016 [ASPP]

2 Conformance Claims

2.1 CC Conformance

This TOE is conformant to:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 2 extended
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 3 extended

2.2 Protection Profile Conformance

This TOE is conformant to:

- Protection Profile for Application Software, version 1.2, dated, 25 April 2016 [ASPP].

2.3 Conformance Rationale

This Security Target provides exact conformance to Version 1.2 of the Protection Profile for Application Software. The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile performing only operations defined there.

2.3.1 Technical Decisions

The following Technical Decisions have been considered for this evaluation:

TD	Applicable	Notes
0359: Buffer Protection	Yes	
0327 – Default file permissions for FMT_CFG_EXT.1.2	Yes	
0326 – RSA-based key establishment schemes	No	The TOE does not support key generation, key establishment, or TLS server functionality.
0305 – Handling of TLS connections with and without mutual authentication	No	The TOE does not support mutual authentication.
0304 – Update to FCS_TLSC_EXT.1.2	Yes	
0300 – Sensitive Data in FDP_DAR_EXT.1	Yes	
0296 – Update to FCS_HTTPS_EXT.1.3	Yes	
0295 – Update to FPT_AEX_EXT.1.3 Assurance Activities	Yes	
0293 – Update to FCS_CKM.1(1)	No	Superseded by TD0326.
0283 – Cipher Suites for TLS in SWApp v1.2	Yes	
0269 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	Superseded by TD0295.
0268 – FMT_MEC_EXT.1 Clarification	Yes	
0267 – TLSS testing - Empty Certificate Authorities list	No	The TOE does not support TLS server functionality.
0244 – FCS_TLSC_EXT - TLS Client Curves Allowed	Yes	
0241 – Removal of Test 4.1 in FCS_TLSS_EXT.1.1	No	The TOE does not support TLS server functionality.
0238 – User-modifiable files FPT_AEX_EXT.1.4	Yes	

TD	Applicable	Notes
0221 – FMT_SMF.1.1 - Assignments moved to Selections	Yes	
0218 – Update to FPT_AEX_EXT.1.3 Assurance Activity	No	Superseded by TD0326.
0217 – Compliance to RFC5759 and RFC5280 for using CRLs	Yes	
0215 – Update to FCS_HTTPS_EXT.1.2	Yes	
0192 – Update to FCS_STO_EXT.1 Application Note	Yes	
0178 – Integrity for installation tests in AppSW PP	No	The TOE does not run on Apple iOS.
0177 – FCS_TLSS_EXT.1 Application Note Update	No	The TOE does not support TLS server functionality.
0174 – Optional Ciphersuites for TLS	Yes	
0172 – Additional APIs added to FCS_RBG_EXT.1.1	Yes	
0163 – Update to FCS_TLSC_EXT.1.1 Test 5.4 and FCS_TLSS_EXT.1.1 Test	Yes	
0131 – Update to FCS_TLSS_EXT.1.1 Test 4.5	No	The TOE does not support TLS server functionality.
0122 – FMT_SMF.1.1 Assignments moved to Selections	No	Superseded by TD0221.
0121 – FMT_MEC_EXT.1.1 Configuration Options	Yes	
0119 – FCS_STO_EXT.1.1 in PP_APP_v1.2	Yes	
0107 – FCS_CKM - ANSI X9.31-1998, Section 4.1.for Cryptographic Key Generation	No	Superseded by TD0326.

Table 3 TDs

3 Security Problem Definition

The security problem definition has been taken from [ASPP] and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any organizational security policies that the TOE is expected to enforce.

3.1 Threats

The following threats are drawn directly from the ASPP.

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

Table 4 Threats

3.2 Assumptions

The following assumptions are drawn directly from the ASPP.

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

Table 5 OSPs

3.3 Organizational Security Policies

There are no OSPs for the application

4 Security Objectives

The security objectives have been taken from [ASPP] and are reproduced here for the convenience of the reader.

4.1 Security Objectives for the TOE

The following security objectives for the TOE were drawn directly from the ASPP.

ID	TOE Objective
O.INTEGRITY	Conformant TOEs ensure the integrity of their installation and update packages, and also leverage execution environment-based mitigations. Software is seldom if ever shipped without errors, and the ability to deploy patches and updates to fielded software with integrity is critical to enterprise network security. Processor manufacturers, compiler developers, execution environment vendors, and operating system vendors have developed execution environment-based mitigations that increase the cost to attackers by adding complexity to the task of compromising systems. Application software can often take advantage of these mechanisms by using APIs provided by the runtime environment or by enabling the mechanism through compiler or linker options. Addressed by: FDP_DEC_EXT.1, FMT_CFG_EXT.1, FPT_AEX_EXT.1, FPT_TUD_EXT.1
O.QUALITY	To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs. Addressed by: FMT_MEC_EXT.1, FPT_API_EXT.1, FPT_LIB_EXT.1
O.MANAGEMENT	To facilitate management by users and the enterprise, conformant TOEs provide consistent and supported interfaces for their security-relevant configuration and maintenance. This includes the deployment of applications and application updates through the use of platform-supported deployment mechanisms and formats, as well as providing mechanisms for configuration. This also includes providing control to the user regarding disclosure of any PII. Addressed by: FMT_SMF.1, FPT_IDV_EXT.1, FPT_TUD_EXT.1.5, FPR_ANO_EXT.1
O.PROTECTED_STORAGE	To address the issue of loss of confidentiality of user data in the event of loss of physical control of the storage medium, conformant TOEs will use data-at-rest protection. This involves encrypting data and keys stored by the TOE in order to prevent unauthorized access to this data. This also includes unnecessary network communications whose consequence may be the loss of data. Addressed by: FDP_DAR_EXT.1, FCS_STO_EXT.1, FCS_RBG_EXT.1
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application. Addressed by: FTP_DIT_EXT.1, FCS_TLSC_EXT.1, FCS_DTLS_EXT.1, FCS_RBG_EXT.1

Table 6 Objectives for the TOE

4.2 Security Objectives for the Operational Environment

The following security objectives for the operational environment assist the TOE in correctly providing its security functionality. These track with the assumptions about the environment.

ID	Objective for the Operation Environment

OE.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.
OE.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.
OE.PROPER_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

Table 7 Objectives for the environment

5 Security Requirements

This section identifies the Security Functional Requirements for the TOE and/or Platform. The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012 and all international interpretations.

Requirement	Auditable Event
Mandatory SFRs	
FCS_RBG_EXT.1	Cryptographic Operation - Keyed-Hash Message Authentication
FCS_STO_EXT.1	Storage of Secrets
FDP_DEC_EXT.1	Access to Platform Resources
FDP_NET_EXT.1	Network Communications
FDP_DAR_EXT.1	Encryption Of Sensitive Application Data
FMT_MEC_EXT.1	Supported Configuration Mechanism
FMT_CFG_EXT.1	Secure by Default Configuration
FMT_SMF.1	Specification of Management Functions
FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Info
FPT_API_EXT.1	Use of Supported Services and APIs
FPT_AEX_EXT.1	Anti-Exploitation Capabilities
FPT_TUD_EXT.1	Integrity for Installation and Update
FPT_LIB_EXT.1	Use of Third Party Libraries
FTP_DIT_EXT.1	Protection of Data in Transit
Optional, Selection-Based and Objective SFRs	
FCS_CKM_EXT.1	Cryptographic Key Generation Services
FCS_TLSC_EXT.1	TLS Client Protocol
FCS_TLSC_EXT.4	TLS Client Protocol
FCS_HTTPS_EXT.1	HTTPS Protocol
FIA_X509_EXT.1	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FPT_IDV_EXT.1	Software Identification and Versions

Table 8 SFRs

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with **bold** text and are surrounded by brackets;
- Refinement: Indicated with **bold** text;
- Selection: Indicated with ***bold italic*** text and are surrounded by brackets;
- Assignment within a selection: Indicated with ***underlined bold italic*** text and are surrounded by brackets;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the PP itself, the formatting used in the PP has been retained.

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations matches the formatting specified within the PP.

5.2 Security Functional requirements

5.2.1 Cryptographic Support (FCS)

FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1

The application shall [***generate no asymmetric cryptographic keys***].

FCS_HTTPS_EXT.1 HTTPS Protocol

FCS_HTTPS_EXT.1.1

The application shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2

The application shall implement HTTPS using TLS in accordance with [***FCS_TLSC_EXT.1***].

FCS_HTTPS_EXT.1.3

The application shall [***not establish the connection***] if the peer certificate is deemed invalid.

FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1

The application shall [***use no DRBG functionality***] for its cryptographic operations

FCS_STO_EXT.1 Storage of Credential

FCS_STO_EXT.1.1

The application shall [***not store any credentials***] to non-volatile memory.

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1

The application shall [***invoke platform-provided TLS 1.2***] supporting the following cipher suites:

[

- ***TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246***
- ***TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246***
- ***TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246***
- ***TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288***
- ***TLS_DHE_RSA_WITH_AES_256_GCM_ SHA384 as defined in RFC 5288***
- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289***
- ***TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289***
- ***TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289***
- ***TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289***

- ***TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289]***

and no other cipher suite.

FCS_TLSC_EXT.1.2

The application shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3

The application shall establish a trusted channel only if the peer certificate is valid.

FCS_TLSC_EXT.4 TLS Client Protocol

FCS_TLSC_EXT.4.1

The application shall present the supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [***secp256r1, secp384r1, secp521r1***] ~~and no other curves~~.

5.2.2 User Data Protection (FDP)

FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1

The application shall restrict its access to [***network connectivity, [mouse and keyboard]***].

FDP_DEC_EXT.1.2

The application shall restrict its access to [***system logs***].

FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1

The application shall restrict network communication to [**:**

- ***respond to*** *[**SEP Manager policy updates and scan commands as well as LiveUpdate definition updates**],*
- *[**uploading status information and detection events to SEP Manager**]*

]

FDP_DAR_EXT.1 Encryption of Sensitive Application Data

FDP_DAR_EXT.1.1

The application shall [

not store any sensitive data

] in non-volatile memory.

5.2.3 Identification and Authentication (FIA)

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1

The application shall [***invoked platform-provided functionality***] to validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The application shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The application shall validate the revocation status of the certificate using [***a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3***].
- The application shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
    - S/MIME certificates presented for email encryption and signature shall have the Email Protection purpose (id-kp 4 with OID 1.3.6.1.5.5.7.3.4) in the extendedKeyUsage field.
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.
    - Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the extendedKeyUsage field.

FIA_X509_EXT.1.2

The application shall treat a certificate as a CA certificate only if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

The application shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [***HTTPS, TLS***].

FIA_X509_EXT.2.2

When the application cannot establish a connection to determine the validity of a certificate, the application shall [***not accept the certificate***].

5.2.4 Security Management (FMT)

FMT_MEC_EXT.1 Supported Configuration Mechanism

FMT_MEC_EXT.1.1

The application shall invoke the mechanisms recommended by the platform vendor for storing and setting configuration options.

FMT_CFG_EXT.1 Secure by Default Configuration

FMT_CFG_EXT.1.1

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

FMT_CFG_EXT.1.2

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

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1

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

[

- ***enable/disable the transmission of any information describing the system's hardware, software, or configuration,***
- ***enable/disable transmission of any application state (e.g. crashdump) information,***
- ***[Configuration of push/pull mode for content updates,***
- ***Configuration of LiveUpdate policy server,***
- ***Configure policy update delay (when computer is idle)]***

]

].

5.2.5 Privacy (FPR)

FPR_ANO_EXT.1 User Consent for Transmission of Personally Identifiable Information

FPR_ANO_EXT.1

The application shall [***not transmit PII over a network***].

5.2.6 Protection of TSF (FPT)

FPT_API_EXT.1 Use of Supported Services and APIs

FPT_API_EXT.1.1

The application shall use only documented platform APIs.

FPT_AEX_EXT.1 Anti-Exploitation Capabilities

FPT_AEX_EXT.1.1

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

FPT_AEX_EXT.1.2

The application shall [***not allocate any memory region with both write and execute permissions***].

FPT_AEX_EXT.1.3

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

FPT_AEX_EXT.1.4

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

FPT_AEX_EXT.1.5

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

FPT_IDV_EXT.1 Software Identification and Versions

FPT_IDV_EXT.1.1

The application shall include SWID tags that comply with the minimum requirements for SWID tag from ISO/IEC 19770-2:2009 standard.

FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1

The application shall [***provide the ability***] to check for updates and patches to the application software.

FPT_TUD_EXT.1.2

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

FPT_TUD_EXT.1.3

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

FPT_TUD_EXT.1.4

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

FPT_TUD_EXT.1.5

The application shall [***provide the ability***] to query the current version of the application software.

FPT_TUD_EXT.1.6

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

FPT_LIB_EXT.1 Use of Third Party Libraries

FPT_LIB_EXT.1.1

The application shall be packaged with only [**Apache, AspectJ, Commons IO, Java JRE, Microsoft JDBC Driver for SQL Server, OAuth, PCRE, PHP, PNG, SQLite, Spring, c-ares, commons-jelly, curl, ezcomponents-lite, Jackson, jquery, libxml2, libxslt**].

5.2.7 Trusted Path/Channel (FTP)

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1

The application shall [***encrypt all transmitted data with [HTTPS, TLS]***] between itself and another trusted IT product.

5.3 TOE SFR Dependencies Rationale for SFRs

The Protection Profile for Application Software contains all the requirements claimed in this Security Target. As such, the dependencies are not applicable since the PP has been approved.

5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the Protection Profile for Application Software which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Assurance Class	Components	Components Description
Development	ADV_FSP.1	Basic Functional Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
Life Cycle Support	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM Coverage
	ALC_TSU_EXT.1	Timely Security Updates
Tests	ATE_IND.1	Independent Testing – Conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability Analysis

Table 9 Security Assurance Requirements

5.5 Rationale for Security Assurance Requirements

The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Symantec to satisfy the assurance requirements. The table below lists the details.

SAR Component	How the SAR will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes).
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the

SAR Component	How the SAR will be met
	users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and how potential changes are incorporated.
ALC_CMS.1	
ALC_TSU_EXT.1	As a founding member of the Organization for Internet Safety (OIS), Symantec is committed to following the Responsible Disclosure guidelines developed by OIS and described in ISO 29417 for externally reported vulnerabilities in Symantec products. Users of Symantec Endpoint Protection should report any security related issues via the secure@symantec.com email address. Symantec encourages finders to use encrypted communication channels to protect the confidentiality of vulnerability reports. Symantec's PGP public key is available from the company website. If the submitted finding is confirmed as valid, Symantec will move forward with providing remediation or mitigation of the issue depending on type, severity, and number of impacted products or services. The Symantec PSIRT team will keep the reporter of the vulnerability up-to-date on progress until the issue has been fully addressed. If the submitted finding is confirmed as valid, Symantec will move forward with providing remediation or mitigation of the issue depending on type, severity, and number of impacted products or services. The Symantec PSIRT team will keep the reporter of the vulnerability up-to-date on progress until the issue has been fully addressed. Software updates/fixes are provided via the Symantec user portal as signed .msi files. Public availability of an update for a publicly disclosed vulnerability is typically 90 days or less.
ATE_IND.1	Symantec will provide the TOE for testing.
AVA_VAN.1	Symantec will provide the TOE for testing.

Table 10 TOE Security Assurance Measures

6 TOE Summary Specification

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

TOE SFR	Rationale
FCS_CKM_EXT.1	The TOE does not perform generation of asymmetric cryptographic keys.
FCS_RBG_EXT.1	The TOE does not directly use any DRBG functionality for any SFR related functionality.
FCS_STO_EXT.1	All secure credentials used for authorization by the TOE are stored on the Symantec Endpoint Protection (SEP) Manager server (external to the TOE). The TOE establishes a secure TLS connection with SEP Manager before receiving these credentials to ensure they are not transmitted in plaintext. These credentials include the administrator names and their associated password hashes as well as database credentials for access to the applicable database indicated for the application. As identified in FCS_CKM_EXT.1, the TOE is not responsible for generation of any asymmetric cryptographic keys.
FCS_TLSC_EXT.1, FCS_TLSC_EXT.4, FCS_HTTPS_EXT.1	The TOE implements TLS 1.2 for use in establishing secure connections to external IT entities. By default, TLS 1.0, TLS 1.1, SSL 2.0 and SSL 3.0 connections are denied. The TOE supports the following encryption algorithms for use with TLS connections: • TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246 • TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • TLS_DHE_RSA_WITH_AES_256_GCM_ SHA384 as defined in RFC 5288 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 During establishment of the TLS 1.2 session, the TOE will leverage the platform to perform verification of the presented identifier in the peer certificate to ensure that it is a valid reference identifier. This ensures that the reference identifier is conformant with RFC 6125. The TOE supports Common Name (CN) and Subject Alternative Names (SAN) reference identifiers. IP addresses and DNS are supported for both CN and SAN. The TOE supports wildcards and IP addresses. Additionally, the TOE does not perform certificate pinning. The TOE presents the Elliptic Curves Extension (also known as the Supported Groups Extension) containing secp256r1, secp384r1, and secp521r1 as part of the Client Hello. These curve are available by default without additional configuration.
FDP_DAR_EXT.1	During operation of the TOE, no sensitive data is stored in non-volatile memory. User credentials that may be leveraged for authentication purposes are stored on the SEP Manager application external to the TOE. These values are written to volatile memory and overwritten after use.

TOE SFR	Rationale
FDP_DEC_EXT.1	During operation of the TOE, access to the underlying platform is limited to use of network connectivity hardware for communication with SEP Manager and LiveUpdate servers as well as the mouse and keyboard for interacting with the application. Additionally, the TOE maintains restricted access to the windows event log for storing of relevant audit events. These are stored in the Event log but are not viewable by the TOE.
FDP_NET_EXT.1	During regular operation of the TOE, secure TLS sessions may be established to communicate with the SEP Manager. These interactions are performed based on the following events: • Response to SEP Manager Policy Updates • Response to SEP Manager Definition Updates • Response to SEP Manager Scan Commands • Uploading status information to SEP Manager • Uploading Detection events to SEP Manager
FIA_X509_EXT.1, FIA_X509_EXT.2	Certificate validation and certificate path validation performed by the TOE platform (Windows 10) is conformant with RFC 5280. The Certificate Revocation List (CRL) is conformant to RFC 5759. Validity checks are performed using X509TrustManager, a Java API packaged with the TOE. For certificates to successfully validate, the certificate cannot be revoked. In addition to the revocation check, the certificate must have a valid basicConstraints extension and extendedKeyUsage field. If for any reason the TOE is unable to determine the validity of a certificate, the certificate will not be accepted.
FMT_CFG_EXT.1	No credentials are required for use with the TOE. Authentication through the underlying Windows platform is used to ensure only authorized access to the application is possible.
FMT_MEC_EXT.1 FMT_SMF.1	The TOE maintains a limited set of configuration settings to run in the evaluated configuration. Configuration options include: • Enable/disable the transmission of any information describing the system's hardware, software, or configuration, • Enable/disable transmission of any application state (e.g. crashdump) information, • Configuration of push/pull mode for content updates • Configuration of LiveUpdate policy server • Configure policy update delay (when computer is idle)
FPR_ANO_EXT.1	The TOE does not transmit any PII over the network.
FPT_AEX_EXT.1	When the TOE is compiled, the ASLR compiler flags are enabled (/DYNAMICBASE flag). The developer also ensures during compilation that all necessary flags are set to protect against stack-based buffer overflow (/GS flag).
FPT_API_EXT.1	The TOE supports the following API: WinHTTP, BCrypt, Windows Driver Kit, CStdStubBuffer, CryptoAPI, National Language Support, WinAPI, WinINet, Native Windows API, WINGDIPAPI, Windows Shell API, Winsock API, WNet API, Remote Desktop Services API, C Library API, basic_iostream, basic_ios, basic_ostream, Clatan2, libm, wstat32i64, wstat64i32, wutime32, wutime64, localtime32, localtime64, malloc, lseeki64, mbstring, stdio, stat32, stat64, strtoi64, wcstoui64, wstat32i64, wstat64i32, wutime32, wutime64, string, wchar, rand
FPT_IDV_EXT.1	The TOE installation includes the use of SWID tags conformant to the ISO/IEC 19770-2:2009 standard. All SWID tags include the required SoftwareIdentity element as well as the Entity element. SWID tags are stored with the .swidtag extension.
FPT_LIB_EXT.1	The TOE, during installation, comes packaged with the following libraries: Apache, AspectJ, Commons IO, Java JRE, Microsoft JDBC Driver for SQL Server, OAuth, PCRE, PHP, PNG, SQLite,

TOE SFR	Rationale
	Spring, c-ares, commons-jelly, curl, ezcomponents-lite, Jackson, jquery, libxml2, and libxslt.
FPT_TUD_EXT.1	During operation of the TOE, if an update becomes available, LiveUpdate can be used to automatically receive new policy and definition updates. TOE software can be manually by installing an MSI obtained from the SEPM server.. All updates are digitally signed by the SEP Manager or LiveUpdate Server to ensure they are provided from an authorized source. Users of the Symantec Endpoint Protection Client should report any security related issues via the Symantec Support web page, which provides a secure channel. Software updates/fixes are provided via the Symantec secure download page. Public availability of an update for a publicly disclosed vulnerability is typically 90 days or less and a maximum of 180 days.
FTP_DIT_EXT.1	All communication sent between the TOE and any external IT entity is encrypted to protect all transmitted data. This communication is performed over HTTPS and TLS. User credentials sent to the Symantec Endpoint Protection are secured through these secure channels.

Table 11 TOE Summary Specification SFR Description

--End of Document--