Security Target

McAfee Change Control and Application Control 8.3.0

with

ePolicy Orchestrator 5.10.0

Document Version: 1.2

October 15, 2020

McAfee,LLC.

2821 Mission College Blvd.

Santa Clara, CA 95054

Abstract

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE): McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0. This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and a specification for the IT security functions provided by the TOE that meet the set of requirements.

Table of Contents

Table of Figures

List of Tables

1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the ST organization. The Target of Evaluation is the McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0, and will hereafter be referred to as the TOE throughout this document. The TOE is a change control and application control software solution with robust management functionality.

1.1 Purpose

This ST is divided into nine sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functions and describes the physical and logical scope for the TOE, as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any Common Criteria (CC), ST Protection Profile, and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, organizational security policies, and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Requirements (Section 6) – Presents the SFRs and SARs met by the TOE.
- TOE Summary Specification (Section 7) – Describes the security functions provided by the TOE that satisfy the security functional requirements and objectives.
- Rationale (Section 8) - Presents the rationale for the security objectives, requirements, and SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms (Section 9) – Defines the acronyms and terminology used within this ST.

1.2 Security Target and TOE References

Table 1 – ST and TOE References

ST Title	Security Target McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0
ST Version	Version 1.2
ST Author	McAfee, LLC
ST Publication Date	October 15, 2020
TOE Reference	McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0
Keywords	Change Control, Application Control, McAfee, ePolicy Orchestrator, ePO, McAfee Agent, Change Prevention

1.3 Product Overview

The Product Overview provides a high level description of the McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0, which is the subject of the evaluation. The following section, TOE Overview, provides the introduction to the parts of the overall product offering that are being evaluated.

McAfee Change Control and Application Control 8.3.0 with ePolicy Orchestrator 5.10.0 provides change control and monitoring on servers and desktops. It also ensures that only authorized code can run on those managed systems. This functionality is managed through the ePolicy Orchestrator (ePO) management software.

The product consists of four logical components:

- Change Control
- Application Control
- ePO (for management of Change Control and Application Control)
- McAfee Agent

These four logical components are implemented via four physical software components:

- Solidcore Service – Manages the policy for the Filesystem Driver and interfaces with the CLI and McAfee Agent.
- Filesystem Driver (swin.sys) – The portion of the product implemented in the Operating System's (OS) kernel space; the filesystem driver intercepts and analyzes all file system, registry, memory, and other critical reads and writes occurring in the OS and implements the core application control and change control and monitoring actions.
- ePO – Used for remote management of the Solidcore Service.
- McAfee Agent – A generic agent used by ePO for communication with a managed endpoint. The agent distributes policies from and reports back to ePO.

In addition, the product interacts with a third-party database in the TOE environment.

The database and the four physical software components of the product are shown in Figure 1 below as they are configured in a typical implementation of the product.

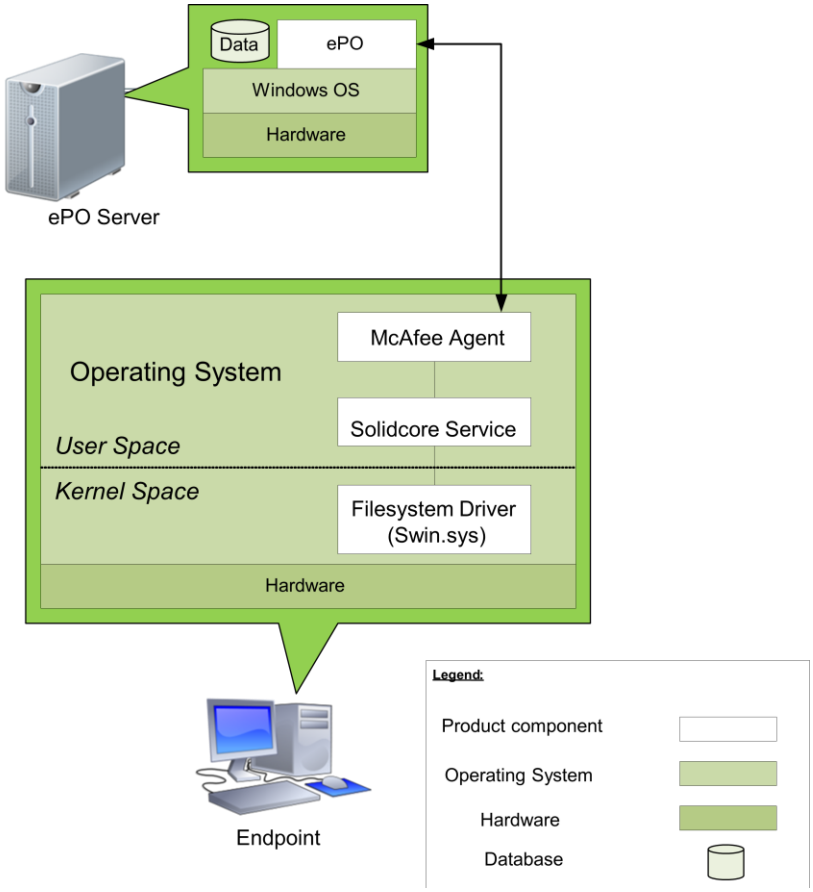

Figure 1 – Software Components of the Product

The following sections describe each of the logical components of the product.

1.3.1 Change Control Monitoring

The Solidcore Service contains Change Control functionality, which monitors change actions happening on the managed system. Change Control can monitor changes to the following:

- Files and directories
- Windows Registry entries
- Process execution/termination
- User activity (Logon/Logoff)

Change Control tracks all changes to the files and directories on the managed system. Types of changes monitored on files and directories include:

- Creation
- Modification of contents
- Deletion
- Renaming
- File attribute modification
- Access Control List (ACL) modification
- Owner modification

Change Control also monitors changes to network file shares, such as Network File Server (NFS) and Client for NFS Services (NFS Client), as well as Common Internet File System (CIFS)/Server Message Block (SMB) for Windows systems. Change Control also monitors changes to file attributes on Windows systems, such as 'FILE_ATTRIBUTE_ENCRYPTED', and 'FILE_ATTRIBUTE_HIDDEN', etc. Change Control monitors the start and stop events for process execution, as well. In addition, it monitors the success or failure of user logon and logoff attempts, and other account changes.

For each change made to an object, Change Control generates a change event. It uses event filters to tailor which change events appear in the event viewer. These filters can be customized by the administrator. Filters can be set on files, directories, registries, process names, file extensions, and user names. Filters match criteria based on file extension, path name, process name, user name, or registry name for change events. Filters can be configured in two different ways:

- Include filters cause events matching the filtering criteria to be reported to the user
- Exclude filters cause events matching the condition to be suppressed and not reported to the user.

The filtering of change events for the purpose of reporting them ensures that only change events the administrator is interested in are recorded. Many change events are program-generated, and may not be of interest to the administrator. Filtering helps reduce the volume of change events being recorded, and thereby reduces the 'noise' on the system. Filter rules are implemented in a predefined order of precedence. For example, filters based on user name will have the highest precedence over all other filter rules.

1.3.2 Change Control

The Solidcore Service also contains Change Policy Enforcement functionality, which prevents specified reads or writes to files and directories on the managed systems. Any addition, removal, or modification of software on the managed system is allowed only when the product is in Update Mode, which also tracks every change action made.

1.3.2.1 Write Protection

Critical files, directories, and volumes can be write-protected using the 'deny-write' feature of Solidcore Services. This renders the specified files as read only. The following operations are controlled by this feature:

- Deletion
- Renaming
- Creation of hard links
- Modifying contents

- Appending
- Truncating
- Changing owner
- Creation of Alternate Data Stream[1] (ADS)

When a directory or volume is specified for write-protection, all files in that directory or volume are added to the write-protected list. These specifications are inherited by sub-directories, as well. In addition to the operations listed above, creation of new files is also denied for directories or volumes listed as write-protected. If any file or directory within a parent directory is write-protected, renaming of the parent directory is also denied. All operations listed above on a write-protected file, directory, or volume are considered unauthorized, and are therefore stopped and an event is generated in the Event Log.

Critical registry keys can also be protected against change using the 'deny-write' feature. All enforcement rules to control modifications to registry keys can be applied using this feature. Any unauthorized attempts to modify a write-protected registry key will be stopped, and a change event will be generated.

1.3.2.2 Read Protection

Critical files, directories, and volumes can also be read-protected using the 'deny-read' feature of Solidcore Services. This enforces read-protection on specified files, directories, and volumes, and also denies the execution of script files. When a directory or volume is specified for read-protection, all files in that directory or volume are added to the read-protected list. The rules are inherited by sub-directories, as well. All unauthorized attempts to read a read-protected file, directory, or volume are stopped, and an event is generated in the Event Log.

1.3.3 Application Control

The Solidcore Service also contains Application Control functionality, which prevents the execution of unauthorized program code on a managed system. Upon initial configuration, Application Control takes an initial snapshot of the software implemented on a managed system, and creates a whitelist inventory of the program code that exists at that time on the system. The listed program code includes binary executables such as '.exe' and '.dll' files, as well as scripts, such as '.bat', '.cmd', and '.vbs' files. This becomes the list of code that will be allowed to run on the managed system.

The following types of control are enforced on the program code that is resident on the managed system's disk, or executed on the managed system:

- Execution control
- Memory control
- Tamper-proofing

1.3.3.1 Execution Control

Execution control prevents all programs not in the inventory from executing on the managed system. All programs not in the inventory are considered unauthorized, their execution is prevented, and their

[1] Alternate Data Streams are metadata associated with a file system object, and are also known as "forks".

failure to execute is logged. This enforcement prevents unauthorized programs such as worms, viruses, and spyware, which install themselves, from executing; and also provides protection against fileless malware and script-based attacks.

1.3.3.2 Memory Control

Memory control protects running processes from malicious attempts to hijack them. Unauthorized code injected into a running process is trapped, halted, and logged. In this fashion, attempts to gain control of a system through buffer overflow and similar exploits are rendered ineffective, and logged.

1.3.3.3 Tamper-proofing

Tamper-proofing prevents intentional and unintentional changes to files that are in the inventory by users or programs.

The Solidcore Service can be put into "Update Mode" in order for software maintenance to be performed. This allows all update actions to be bracketed within an update window. Update actions include addition, removal, or modification of software on the system. It will track every update action and automatically updates the whitelist inventory. This enables new or modified software to run when the managed system returns to normal operation ("Enable Mode").

In addition to real-time prevention of execution of unauthorized code, Application Control also performs reviews of the Event Log and other internal logs of changes to the managed system to identify applications that are attempting to perform updates, or fail to run when they execute. At times these applications should be allowed to update or run, and this information is brought to the attention of the administrator. The administrator can then take the recommended action.

1.3.4 ePolicy Orchestrator

The ePolicy Orchestrator, or ePO, provides a platform for centralized policy management and enforcement of the Application Control and Change Control product on the managed systems. It uses the System Tree to organize managed systems into units for monitoring, assigning policies, scheduling tasks, and taking actions. The System Tree is a hierarchical structure that allows administrators to combine managed systems into groups. Policies can then be applied to groups of managed systems, rather than individually.

ePO allows administrators to manage the targeted systems from a single location through the combination of product policies and client tasks. Policies ensure that the application control and change control features are configured correctly. Client tasks are the scheduled actions that run on the managed systems hosting the Solidcore Services. Client tasks are commonly used for product deployment, product functionality, upgrades, and updates.

The ePO software is comprised of several components:

- ePO server
- Database
- Master repository
- McAfee Agent

Each of these is described in the following sections.

1.3.4.1 ePO Server

This is the center of the managed environment. The ePO server delivers application control and change control policies, controls updates, and processes the events for all the managed systems. It includes the following subcomponents:

- Application server – includes the Automatic Response[2] functionality, Registered Servers (see below), and the user interface
- Agent handler – distributes network traffic generated by agent-to-server communications; responsible for communicating policies, tasks, and properties
- Event parser – parses events received from Solidcore Services

1.3.4.2 Database

The database is the central storage component for all data created and used by ePO. The database can be housed on the ePO server, or on a separate server, depending on the specific needs of the organization.

1.3.4.3 Master Repository

The Master Repository is the central location for all McAfee updates and signatures, and it resides on the ePO server. The Master Repository retrieves user-specified updates and signatures from McAfee or from user-defined source sites.

1.3.4.4 McAfee Agent

The McAfee Agent is a vehicle of information and enforcement between the ePO server and each managed system. The McAfee Agent retrieves updates, ensures task implementation, enforces policies, and forwards events for each managed system. It uses a separate secure channel to transfer data to the ePO server. The McAfee Agent can also be configured as a SuperAgent with the addition of a repository.

1.4 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

The TOE is an application and change control software-only TOE. The TOE includes all the functionality described above in Section 1.3, except where indicated. Those features and functionality excluded from the scope of the TOE are listed below in Section 1.5.3. The TOE runs on the platforms described below in Section 1.4.2.

Figure 2 shows the details of the deployment configuration of the TOE.

[2] Automatic Response functionality allows administrators to create rules for responding to events that are specific to the managed business environment, such as sending email notifications or SNMP traps, or creating issues for use with integrated third-party ticketing systems.

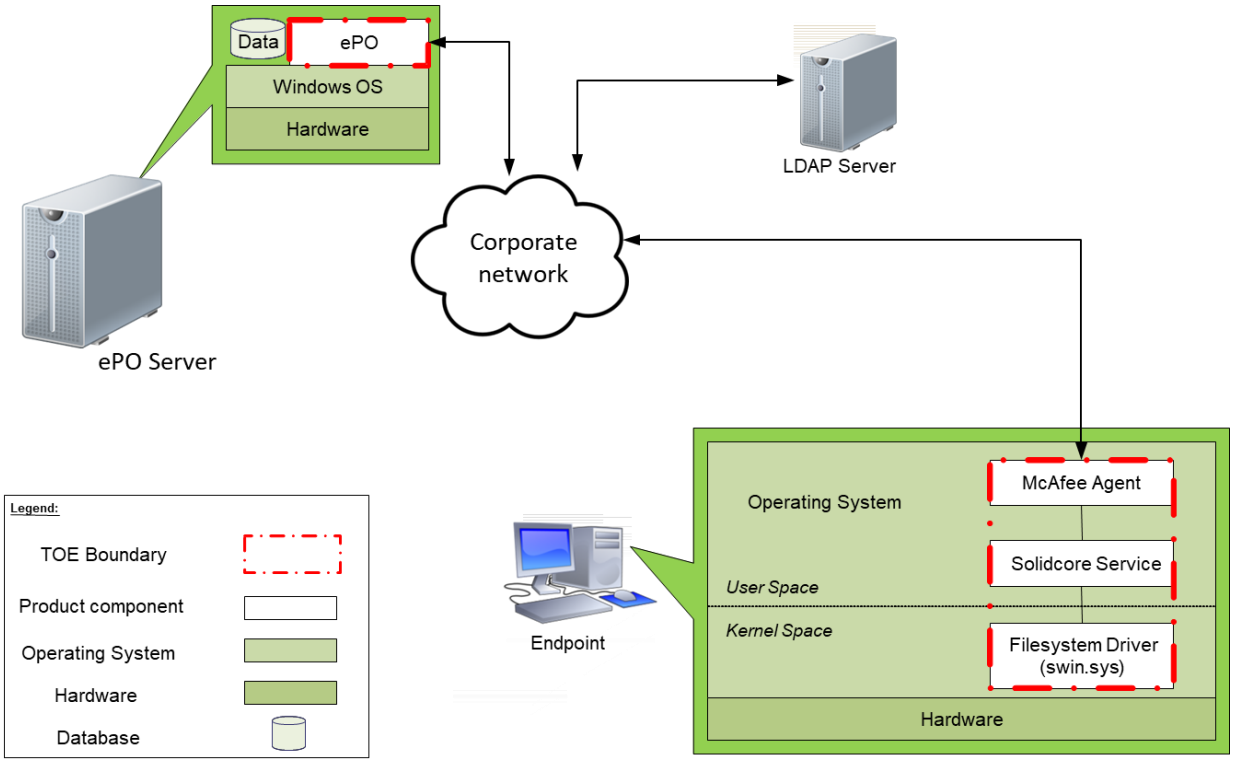

Figure 2 – Deployment Configuration of the TOE

1.4.1 Brief Description of the Components of the TOE

The TOE consists of the following software components:

- Solidcore Service – manages the policy for the Filesystem Driver and interfaces with the CLI and McAfee Agent;
- Filesystem Driver (swin.sys) – the portion of the product implemented in the Operating System's (OS) kernel space; the file system driver intercepts and analyzes all file system, registry, memory, and other critical reads and writes occurring in the OS and implements the core application control and change control actions;
- ePO – for remote management of the Solidcore Service;
- McAfee Agent – a plug-in to the Solidcore Service used by ePO.

The software packages that comprise the TOE are as follows:

- McAfee Solidcore ePO Server Extension 8.3.0-225,
- Solidcore client 8.3.0-303[3],
- ePO Server 5.10.0 (download package EPO_510_2428_18_LR4.zip),
- ePO Server 5.10.0 Update 6 (download package ePO_5.10.0_Update_6.zip),
- McAfee Agent 5.6.4.151,
- McAfee Agent Extension 5.6.4.179.

[3] "Solidcore" represents the Change Control and Application Control software. The Solidcore Service and Filesystem Driver are provided by the Solidcore Client.

The CLI Utility is excluded from the evaluation, and must be disabled.

1.4.2 TOE Environment

In the evaluated configuration Change Control and Application Control run on the following endpoint platforms:

- Windows 10 version 1909
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019

ePO runs on Windows Server 2019

The following third-party products are used by the TOE in the CC-evaluated configuration:

- Active Directory (LDAP) Server
- MS SQL Server 2017 database

1.5 TOE Description

This section primarily addresses the physical and logical components of the TOE included in the evaluation.

1.5.1 Physical Scope

Figure 3 illustrates the physical scope and the physical boundary of the overall solution and ties together all of the components of the TOE and the constituents of the TOE Environment.

The TOE is an application control and change control product that runs on a Windows platform compliant to the minimum software and hardware requirements as listed in Table 2. The physical TOE boundary is depicted in Figure 3 below. The essential logical components for the proper operation of the TOE in the evaluated configuration are the TOE software, one of the designated Windows OSs, and an LDAP Server. The general-purpose hardware platforms for the TOE, physical network cables and devices, and servers running required network services (such as Domain Name System – DNS) are the only required physical components for the proper operation of the TOE.

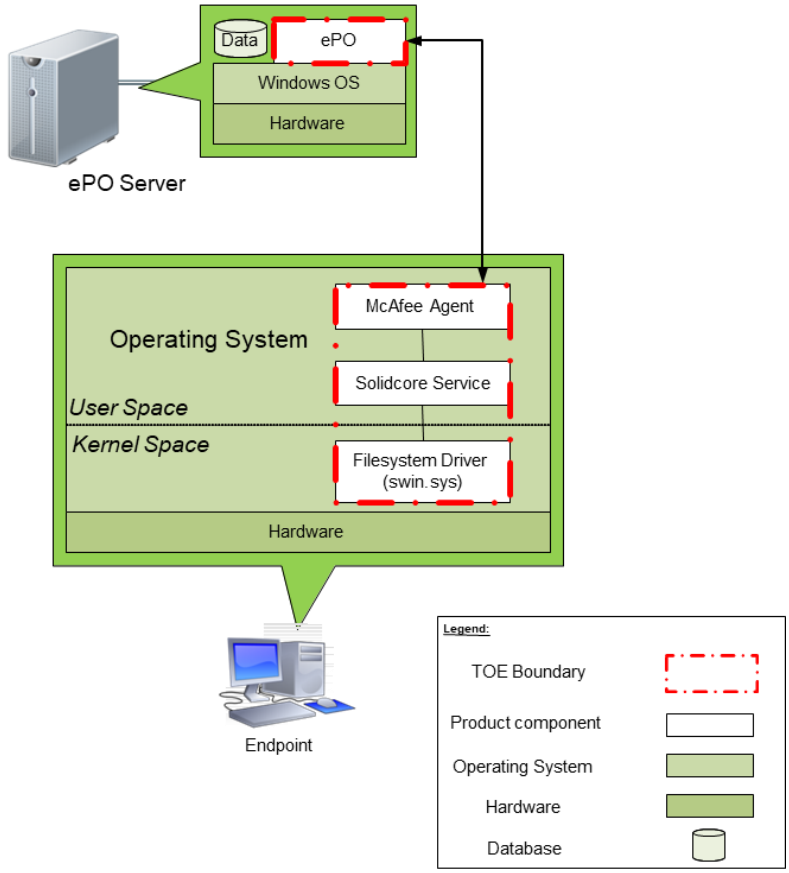

Figure 3 – Physical TOE Boundary

1.5.1.1 TOE Platform Minimum Requirements

Table 2 specifies the minimum system requirements for operation of the TOE in the evaluated configuration.

Table 2 – TOE Platform Minimum Requirements

Component	Minimum System Requirements
Endpoint Workstation	• Single Intel Pentium CPU or higher • 512 MB RAM • 100 MB free disk space • TCP/IP protocol installed • Operating system: choice of: - Windows 10 version 1909 - Windows Server 2012 R2 - Windows Server 2016 - Windows Server 2019
ePO Server	• 64-bit Intel compatible (4 cores minimum recommended) • 8 GB Physical RAM

Component	Minimum System Requirements
	• 20 GB free disk space • 1024 x 768, 256-color, VGA monitor • 100 MB or higher Network Interface Card • Internet Explorer 8-12 or Firefox 10-44 or Chrome 17-48 or Safari 6-9 browser • MS SQL Server 2017 database • Windows Server 2019 operating system

1.5.1.2 Guidance Documentation

The following guides are required reading and part of the TOE[4]:

McAfee ePolicy Orchestrator

- McAfee ePolicy Orchestrator 5.10.0 Product Guide (Revision B, 2-12-2019)
- McAfee ePolicy Orchestrator 5.10.0 Installation Guide (8-6-2018)
- Release Notes McAfee ePolicy Orchestrator 5.10.0 (Revision B, 8-28-2018)
- Release Notes McAfee ePolicy Orchestrator 5.10.0 Update 6 (1-14-2020)

McAfee Agent

- McAfee Agent 5.6.x Product Guide (Revision C, 3-10-2020)
- McAfee Agent 5.6.x Installation Guide (3-10-2020)
- McAfee Agent 5.6.x Release Notes (3-10-2020)

McAfee Change Control and Application Control

- McAfee Application Control and McAfee Change Control 8.3.x – Windows Product Guide (3-27-2020)
- McAfee Application Control and McAfee Change Control 8.3.x – Windows Installation Guide (3-27-2020)
- McAfee Change Control and Application Control 8.3.0 - CC Evaluation and Configuration Guide (5-27-2020)
- McAfee Application Control and McAfee Change Control 8.3.x - Windows Release Notes (3-27-2020)

[4] The current versions of the guidance documents listed here are available from the McAfee documentation site (docs.mcafee.com), where they can be viewed using a web browser. Each page is uniquely identified there with the date of last update. If desired the entire document can be downloaded as a PDF file. This will reflect the document content on the date of download. The download filename will be appended with the date, giving it a unique identification.
In the case of ePO and McAfee Agent, versions of the guidance documents are also available for download from the product download site, although as new product versions are released this practice will cease. These document filenames will not be appended with a date, but the date of release is shown on the download page. Any subsequent revised versions of these documents carry a letter (Rev B, C, D…) indicating the level of revision. This revision letter is also included in the filename.

1.5.2 Logical Scope

The security functional requirements implemented by the TOE are usefully grouped under the following Security Function Classes:

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TOE Security Functionality
- McAfee Application and Change Control

1.5.2.1 Security Audit

The TOE generates audit records for all ePO and Solidcore administrator actions. Authorized administrators can view, sort, and filter the audit records. The ePO-generated audit records can be filtered to present only failed actions, or only entries that are within a certain age. Solidcore-generated audit records can be filtered and sorted on the following fields:

- User who performed the action,
- target object of the action,
- computer on which the action was performed,
- action timestamp, and
- action type.

1.5.2.2 Cryptographic Support

The TOE protects transmissions between ePO and McAfee Agent from disclosure and undetected modification by encrypting the transmissions.

1.5.2.3 Identification and Authentication

User identification and authentication are enforced by the TOE. Users must log in to ePO with a valid user name and password via a GUI before any access is granted by the TOE to TOE functions or data. When the credentials are presented by the user, ePO determines if the user name is defined and enabled, and the password is correct. If not, the login process is terminated and the login GUI is redisplayed.

Upon successful login, the permission sets are bound to the session. Those attributes remain fixed for the duration of the session. If the attributes for a logged-in user are changed, those changes will not be bound to a session until the next login by that user.

1.5.2.4 Security Management

The TOE provides administrator support functionality that enables a user to configure and manage TOE components. Management of the TOE is performed via the ePO GUI. Management permissions are defined per-user. The TOE maintains two types of roles:

- Where Users are assigned to the “administrator” permission set, which is a superset of all other permission sets. This includes the default “admin” user account created when ePO is installed. Users assigned to this permission set are known as “Administrator”

- Where Users are assigned to selected permission sets. Users assigned to permission sets (excluding the administrator permission) set are known as “Users with Selected Permissions”.

1.5.2.5 Protection of the TSF

The TOE protects transmissions between the ePO and the McAfee Agent from disclosure by securing the transmissions using RSA BSAFE.

1.5.2.6 McAfee Change Control and Application Control

The TOE provides Application Control and Change Control functionality for managed systems. It does this by collecting information about the program code, files, directories, and volumes that are to be protected. Each time a program attempts to execute, or a process or user attempts to modify a protected resource, the TOE analyzes the attempted action, and determines whether it should be allowed or not. It then takes the appropriate action.

1.5.3 Product Physical/Logical Features and Functionality not included in the TOE

1.5.3.1 Features/Functionality that are not part of the evaluated configuration of the TOE

- CLI Utility
- Reputation based execution using McAfee TIE and GTI
- Product Integrity
- Package Control
- Observation throttling
- AntiDos
- Heartbeat Timeout
- Message Exchange Interval
- Secure Signed Update Utility
- Distributed Repositories
- SNMP
- SuperAgents
- Windows and certificate authentication
- Remote Agent Handlers
- Ticketing functionality
- Rogue System Detection
- Open API to Third-party products

1.5.3.2 Operating system platforms not covered by the evaluation

Change Control and Application Control can be installed on a wide range of endpoint platforms, but only those listed in Section 1.4.2 above are covered by this evaluation. A full list of supported platforms can be found in KB87944.

ePolicy Orchestrator can also be installed on the following 64-bit operating system platforms, but these are not covered by this evaluation[5]:

[5] If using Windows Server 2012 or later, also install Microsoft update 2919355.

- Windows Server 2008 R2 Service Pack 1
- Windows Server 2012
- Windows Server 2012 Service Pack 1
- Windows Server 2016

2 Conformance Claims

This section provides the identification for any CC, Protection Profile (PP), and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in Section 8.1.

Table 3 – CC and PP Conformance

Common Criteria (CC) Identification and Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017; CC Part 2 extended; CC Part 3 conformant; PP claim (none).
PP Identification	None
Evaluation Assurance Level	EAL2+ (Augmented with Flaw Reporting Procedures (ALC_FLR.2))

3 Security Problem Definition

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It provides the statement of the TOE security environment, which identifies and explains all:

- Known and presumed threats countered by either the TOE or by the security environment
- Organizational security policies with which the TOE must comply
- Assumptions about the secure usage of the TOE, including physical, personnel and connectivity aspects

3.1 Threats to Security

This section identifies the threats to the Information Technology (IT) assets against which protection is required by the TOE or by the security environment. The threat agents are divided into two categories:

- Attackers who are not TOE users: They have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters and no physical access to the TOE.
- TOE users: They have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters and physical access to the TOE. (TOE users are, however, assumed not to be willfully hostile to the TOE.)

Both are assumed to have a low level of motivation. The IT assets requiring protection are the user data saved on or transitioning through the TOE and the hosts on the protected network. Removal, diminution and mitigation of the threats are through the objectives identified in Section 4 Security Objectives. The following threats are applicable:

Table 4 – Threats

Name	Description
T.AUTHENTICATE	An authorized user may be unaware of an inadvertent change to TOE data or functions they are authorized to modify.
T.COMPROMISE	An unauthorized user may attempt to disclose, remove, destroy, or compromise the integrity of the data collected and produced by the TOE by bypassing a security mechanism.
T.PROTECT	An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data, or inappropriately change the configuration of the TOE.
T.APP_CHG_CONTROL	An attacker may be able to inappropriately change targeted objects or execute inappropriate software on the managed system without being detected.

3.2 Organizational Security Policies

An Organizational Security Policy (OSP) is a set of security rules, procedures, or guidelines imposed by an organization on the operational environment of the TOE. There are no OSPs defined for this Security Target.

3.3 Assumptions

This section describes the security aspects of the intended environment for the evaluated TOE. The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user guidance. The following specific conditions are required to ensure the security of the TOE and are assumed to exist in an environment where this TOE is employed.

Table 5 – Assumptions

Name	Description
A.ACCESS	The TOE has access to all the IT System data it needs to perform its functions.
A.TIME	The IT Environment will provide reliable timestamps for the TOE to use.
A.LOCATE	The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.
A.PROTECT	The TOE software critical to security policy enforcement, and the hardware on which it runs, will be protected from unauthorized physical modification.
A.MANAGE	There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.
A.NOEVIL	The authorized administrators are not careless, wilfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.
A.DYNAMIC	The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.

4 Security Objectives

Security objectives are concise, abstract statements of the intended solution to the problem defined by the security problem definition (see Section 3). The set of security objectives for a TOE form a high-level solution to the security problem. This high-level solution is divided into two part-wise solutions: the security objectives for the TOE, and the security objectives for the TOE's operational environment. This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security Objectives for the TOE

The specific security objectives for the TOE are as follows:

Table 6 – Security Objectives for the TOE

Name	Description
O.AUDIT	The TOE must record audit records for data accesses and use of the TOE functions on the management system.
O.ACCESS	The TOE must allow authorized users to access only authorized TOE functions and data.
O.AUDIT_REVIEW	The TOE must provide authorized administrators with the ability to review, order, and filter the audit trail.
O.IDENTIFY	The TOE must be able to identify and authenticate users prior to allowing access to TOE administrative functions and data.
O.EADMIN	The TOE must include a set of functions that allow efficient management of its functions and data.
O.PROTECT	The TOE must ensure the integrity of audit and system data by protecting it from unauthorized modifications and access during transfer.
O.COLLECT	The TOE shall collect a list of objects that are to be protected and an inventory of allowable program code for the managed systems.
O.ANALYZE	The TOE must apply analytical processes and information to derive conclusions about allowed and disallowed accesses to objects.
O.REACT	The TOE shall take appropriate action on all allowed and disallowed accesses to objects.

4.2 Security Objectives for the Operational Environment

4.2.1 IT Security Objectives

The following IT security objectives are to be satisfied by the environment:

Table 7 – IT Security Objectives

Name	Description
OE.TIME	The TOE environment must provide reliable timestamps to the TOE.
OE.INTEROP	The TOE is interoperable with the managed systems it monitors.

4.2.2 Non-IT Security Objectives

The following non-IT environment security objectives are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 8 – Non-IT Security Objectives

Name	Description
NOE.INSTALL	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner that is consistent with IT security.
NOE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy, and the hardware on which the TOE runs, are protected from any physical attack.
NOE.PERSON	Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.

5 Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

5.1 Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE. The extended SFRs are organized by class. Table 9 identifies all extended SFRs implemented by the TOE

Table 9 – Extended TOE Security Functional Requirements

Name	Description
EXT_MAC_SDC.1	Application and Change Control Data Collection
EXT_MAC_RCT.1	Application and Change Control React

5.1.1 Class EXT_MAC: McAfee Application and Change Control

Application and Change Control functions involve enforcement of restrictions on execution of applications on the targeted system, and on modification of files on the targeted system. The EXT_MAC: McAfee Application and Change Control class was modeled after the CC FAU: Security Audit class.

The extended family EXT_MAC_SDC: Application and Change Control Data Collection was modeled after the CC family FAU_GEN: Security Audit Data Generation. The extended family EXT_MAC_RCT: Application and Change Control React was modeled after the families FAU_SAA: Potential Violation Analysis and FAU_ARP: Security Alarms.

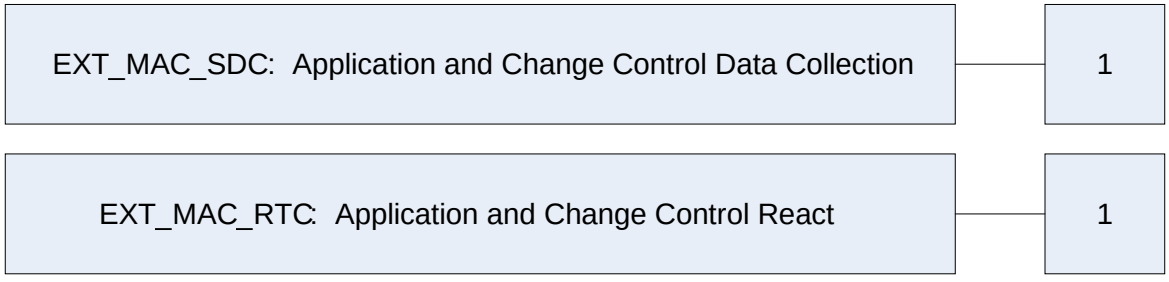

Figure 4 – EXT_MAC: McAfee Application and Change Control Class Decomposition

5.1.1.1 Application and Change Control Data Collection (EXT_MAC_SDC)

Family Behaviour

This family defines the requirements for creating a baseline snapshot of the targeted system for use in determining which applications will be allowed to execute on the system, as well as identifying changes to files, directories, network shares, registry keys, and user accounts. This family enumerates the types of program code that shall be collected by the TOE Security Function (TSF), and identifies what type of control will be enforced on the executable code. This family also determines which change events will be prevented, and which change events will be monitored and reported.

Component Levelling

Figure 5 – Application and Change Control Data Collection family decomposition

EXT_MAC_SDC.1 Application and change control data collection, specifies the list of executable code that shall be allowed to run on the targeted system, as well as identifies changes to files, directories, network shares, registry keys, and user accounts.

Management: EXT_MAC_SDC.1

- There are no management activities foreseen.

Audit: EXT_MAC_SDC.1

- There are no auditable events foreseen.

EXT_MAC_SDC.1 Application and change control data collection

Hierarchical to: No other components

Dependencies: No dependencies

EXT_MAC_SDC.1.1 The System shall be able to collect the following information from the targeted IT System resource(s): [assignment: lists of program code allowed to execute and events indicating allowed, prevented, and monitored actions].

EXT_MAC_SDC.1.2 At a minimum, the System shall collect and record the following information:

- ***[assignment: list of data collected].***

5.1.1.2 Application and Change Control React (EXT_MAC_RCT)

Family Behaviour

This family defines the analysis the TOE performs on the collected application and change control data and the actions to be taken by the TOE in response to the findings of the analysis. This family enumerates the types of program code that shall be collected by the TSF, and identifies what type of

control will be enforced on the executable code. This family also determines which changes are to be prevented, and which are to be monitored and reported.

Component Levelling

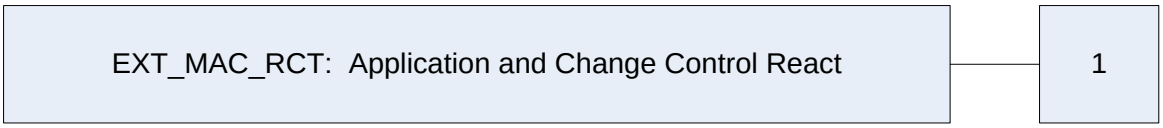

Figure 6 – Application and Change Control React family decomposition

EXT_MAC_RCT.1 Application and change control react, specifies the list of actions that shall be taken for each analytical result obtained against the collected application and change control data.

Management: EXT_MAC_RCT.1

- The management (addition, removal, or modification) of actions.

Audit: EXT_MAC_RCT.1

- Minimal: Actions taken due to application analysis requirements.

EXT_MAC_RCT.1 Application and change control react

Hierarchical to: No other components

Dependencies: EXT_MAC_SDC.1

EXT_MAC_RCT.1.1 The System shall perform the following analysis function(s) on all application data collected and take the associated action(s) in response [assignment: analytical function(s) and associated action(s)].

5.2 Extended TOE Security Assurance Components

This section specifies the extended SARs for the TOE. There are no extended SARs defined for this ST.

6 Security Requirements

6.1 Introduction

This section defines the SFRs and SARs to be met by the TOE. These requirements are presented following the conventions identified below.

Several font styles are used within this security target. These presentation choices are discussed here to aid the security target reader.

The CC allows for assignment, refinement, selection and iteration operations to be performed on security functional requirements. All of these operations are used within this security target. These operations are performed as described in Part 2 of the CC, and are shown as follows:

- Completed assignment statements are identified using [*italicized text within brackets*].
- Completed selection statements are identified using [underlined text within brackets].
- Refinements are identified using **bold text**. Any text removed is stricken (Example: ~~TSF Data~~) and should be considered as a refinement.
- Extended Functional and Assurance Requirements are identified using "EXT_" at the beginning of the short name.
- Iterations are identified by appending a number in parentheses following the component title. For example, FAU_GEN.1(1) Audit Data Generation would be the first iteration and FAU_GEN.1(2) Audit Data Generation would be the second iteration.

6.2 Security Functional Requirements

This section specifies the SFRs for the TOE, organised by CC class. Table 10 identifies all SFRs implemented by the TOE, and indicates the types of operations performed on each requirement.

Table 10 – TOE Security Functional Requirements

Name	Description	S	A	R	I
FAU_GEN.1	Audit data generation	✓	✓		
FAU_SAR.1	Audit review		✓		
FAU_SAR.2	Restricted audit review				
FAU_SAR.3	Selectable audit review		✓		
FCS_CKM.1(1)	Cryptographic key generation (MA)		✓		✓
FCS_CKM.1(2)	Cryptographic key generation (ePO)		✓		✓
FCS_CKM.4	Cryptographic key destruction		✓		
FCS_COP.1	Cryptographic operation		✓	✓	

Name	Description	S	A	R	I
FIA_ATD.1	User attribute definition		✓		
FIA_UID.2	User identification before any action				
FIA_UAU.2	User authentication before any action				
FMT_MTD.1	Management of TSF data	✓	✓	✓	
FMT_SMF.1	Specification of management functions		✓		
FMT_SMR.1	Security roles		✓		
FPT_ITT.1	Basic internal TSF data transfer protection	✓			
EXT_MAC_SDC.1	Application and change control data collection		✓		
EXT_MAC_RCT.1	Application and change control react		✓		

Note: S=Selection; A=Assignment; R=Refinement; I=Iteration

6.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit Data Generation

Hierarchical to: No other components.
Dependencies: FPT_STM.1 Reliable time stamps

FAU_GEN.1.1

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

a) Start-up and shutdown of the audit functions;
b) All auditable events, for the [not specified] level of audit; and
c) [*all Solidcore and ePO administrator actions*].

FAU_GEN.1.2

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*no other audit-relevant information*].

FAU_SAR.1 Audit review

Hierarchical to: No other components.
Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.1.1

The TSF shall provide [*authorised users assigned to the Administrator permission set or assigned to both Global Reviewer and Solidcore Reviewer permission sets*] with the capability to read [*all information*] from the audit records.

FAU_SAR.1.2

The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

FAU_SAR.2 Restricted audit review

Hierarchical to: No other components.

Dependencies: **FAU_SAR.1 Audit review**

FAU_SAR.2.1

The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

FAU_SAR.3 Selectable audit review

Hierarchical to: **No other components.**
Dependencies: **FAU_SAR.1 Audit review**

FAU_SAR.3.1

The TSF shall provide the ability to apply [*sorting and filtering*] of audit data based on [*the fields listed in Table 11 below*].

Table 11 – Selectable audit review fields

TOE Component	Field	Filter/Sort
ePO	Action	Sort
	Completion time	Filter, Sort
	Details	Sort
	Priority	Sort
	Start Time	Filter, Sort
	Success	Filter, Sort
	User Name	Sort
Solidcore	Who performed the action	Filter
	Target object of the action	Filter
	Computer on which action was performed	Filter
	Action timestamp	Filter
	Action type	Filter

Application Note: All ePO Administrator actions, plus the start-up/shutdown functions are recorded in the ePO Audit Log.
All Solidcore actions from endpoints are stored in the ePO database.

6.2.2 Class FCS: Cryptographic Support

FCS_CKM.1(1) Cryptographic key generation (MA)

Hierarchical to: **No other components.**
Dependencies: ***[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]***
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1(1)

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*HMAC_DRBG for random number generation*] and specified cryptographic key sizes [*256 and 2048 bits*] that meet the following [*NIST SP 800-90A*].

FCS_CKM.1(2) Cryptographic key generation (ePO)

Hierarchical to: **No other components.**
Dependencies: ***[FCS_CKM.2 Cryptographic key distribution, or***
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1(2)

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*CTR_DRBG for random number generation*] and specified cryptographic key sizes [*256 and 2048 bits*] that meet the following [*NIST SP 800-90A*].

FCS_CKM.4 Cryptographic key destruction

Hierarchical to: **No other components.**
Dependencies: ***[FDP_ITC.1 Import of user data without security attributes, or***
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [*zeroization*] that meets the following: [*FIPS 140-2 level 1*].

FCS_COP.1 Cryptographic operation

Hierarchical to: **No other components.**
Dependencies: ***[FDP_ITC.1 Import of user data without security attributes, or***
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1

The TSF shall perform [*list of cryptographic operations – see Table 12 below*] in accordance with a specified cryptographic algorithm [*cryptographic algorithm – see Table 12 below*] and cryptographic key sizes [*cryptographic key sizes – see Table 12 below*] that meet the following: [*list of standards – see Table 12 below*].

Table 12 - Cryptographic Operations

Cryptographic Operations	Cryptographic Algorithm	Key Sizes (bits)	Standards
Key Exchange/Authentication	RSA	2048	NIST 800-56B
Symmetric encryption and decryption	Advanced Encryption Standard (AES) (CBC[6], mode)	256	FIPS 197
Secure Hashing	SHA-256	Not Applicable	FIPS 180-3

[6] CBC – Cipher Block Chaining

6.2.3 Class FIA: Identification and Authentication

FIA_ATD.1 User attribute definition

Hierarchical to: **No other components.**
Dependencies: **No dependencies**
FIA_ATD.1.1

The TSF shall maintain the following list of security attributes belonging to individual users: [

a) *ePO User name;*
b) *Enabled or disabled;*
c) *Authentication configuration;*
d) *Obfuscated password (when Local ePO authentication is configured);*
e) *Permission sets*].

FIA_UID.2 User identification before any action

Hierarchical to: **FIA_UID.1 Timing of identification**
Dependencies: **No dependencies**
FIA_UID.2.1

The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.2 User authentication before any action

Hierarchical to: **FIA_UAU.1 Timing of authentication**
Dependencies: **FIA_UID.1 Timing of identification**
FIA_UAU.2.1

The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

6.2.4 Class FMT: Security Management

FMT_MTD.1 Management of TSF data

Hierarchical to: **No other components.**
Dependencies: **FMT_SMF.1 Specification of management functions**
FMT_SMR.1 Security roles
FMT_MTD.1.1

The TSF shall restrict the ability to [query, modify, delete, [*create, enable, disable, and use as specified in Table 13 below*]] the [*TSF data listed in Table 13 below*] to [*an administrator or a user with the permissions identified in Table 13 below*].

Table 13 – TSF Data Access Permissions

TSF Data	Associated Permission	Operations Permitted
Dashboards	Use public dashboards	Use public dashboards
	Use public dashboards; create and edit private dashboards	Use public dashboards; create and modify **private** dashboards

TSF Data	Associated Permission	Operations Permitted
	Use public dashboards; create and edit private dashboards	Use public dashboards; create, delete, and modify private dashboards; make private dashboards public
Audit Log	View audit log	View
	View and purge audit log	View and delete
Permission Set	n/a (only allowed by an Administrator)	Query, new, delete, duplicate, edit, and assign (to a user) permissions
Queries and Reports	Use public groups	Query and use public groups
	Use public groups; create and edit private queries/reports	Query and use public groups; create and modify private queries
	Edit public groups; create and edit private queries/reports; make private queries/reports public	Edit public groups; create, delete, and modify private queries/reports; Make private queries/reports public
Systems	View “System Tree” tab	Query
	Actions	Wake up Agents; view Agent Activity Log Edit System Tree groups and systems Deploy agents
System Tree Access	Access nodes and portions of the System Tree	Access nodes and portions of the System Tree
Solidcore General	Queries, Dashboards	Run queries, view dashboard
	Events	View events, View events, manual reconciliation
	Responses	Create Solidcore event responses
	Alerts	View alerts; View and dismiss alerts
	Client Task Log	View Client Task Log, View and delete Client Task Log
	Inventory	Access to view Inventory, Access to view, modify, import Inventory
	Observations	Manage observation features
	Content Change Tracking	View Content changes, View content changes, Set Base version, Create content Change response
	Policy Discovery	View policy discovery, View policy discovery, Allow/Ban policy discovery requests, Delete policy discovery requests

TSF Data	Associated Permission	Operations Permitted
	Certificates	Access to view certificates, Access to view, modify, import, upload certificates;
	Installers	Access to view installers, Access to view, modify, import, upload installers
	Rule Groups	View permission/Edit permission for the following: Updater processes, Executable files Users Certificates Installers Exclusions Directories Filters Execution Control Rules
Solidcore Policy Permissions	Application Control	View and change policy and task settings
	Change Control	View and change policy and task settings
	Integrity Monitor	View and change policy and task settings
	General policies	View and change policy and task settings

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.
Dependencies: No Dependencies
FMT_SMF.1.1
The TSF shall be capable of performing the following management functions: [*management of TSF data*].

FMT_SMR.1 Security roles

Hierarchical to: No other components.
Dependencies: FIA_UID.1 Timing of identification
FMT_SMR.1.1
The TSF shall maintain the roles [*Administrator and Users with Selected Permissions*].
FMT_SMR.1.2
The TSF shall be able to associate users with roles.

6.2.5 Class FPT: Protection of the TSF

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to: No other components.
Dependencies: No dependencies
FPT_ITT.1.1

The TSF shall protect TSF data from [disclosure] when it is transmitted between separate parts of the TOE.

6.2.6 Class EXT_MAC: McAfee Application and Change Control

EXT_MAC_SDC.1 Application and change control data collection

Hierarchical to: No other components
Dependencies: No dependencies
EXT_MAC_SDC.1.1
The System shall be able to collect the following information from the targeted IT System resource(s):
[
For application control:

a. *A whitelist inventory of program code, including binary executables and scripts;*
b. *Events indicating prevented unauthorized executions of program code;*
c. *Events indicating prevented attempts to modify files;*

For change control:

d. *Events indicating access to critical files, directories, and volumes that are designated as write-protected;*
e. *Events indicating access to all critical files, directories and volumes that are designated as read-protected;*
f. *Events indicating access to all critical registry keys that are designated as write-protected;*

For change control monitoring:

g. *Events indicating the following actions on files, content of directories, content of network shares: creation, modification of contents, deletion, renaming, file attribute modification, ACL modification, owner modification;*
h. *Events indicating the start and stop events for process execution;*
i. *Events indicating the success or failure of user logon or logoff attempts and user account management activities such as user account creation, user account deletion, user account modification (account enabled, account disabled, and password changed).*

]

Application Note: *Critical registry keys (Change Control item #f) are considered to be those under the HKEY_LOCAL_MACHINE branch. Protecting other keys may affect user operation.*

EXT_MAC_SDC.1.2
At a minimum, the System shall collect and record the following information:
[

a) *For application control:*
The Program Name (the application that is performing the action) and the Object Name (the object that is being acted upon);
b) *For change control:*
The name of the protected file (which may include the directory or volume in the path), or key;
c) *For change control monitoring:*
Event generated time, event id, Event Display Name, Object name, and as appropriate the File name, User Name or Program name.

]

EXT_MAC_RCT.1 Application and change control react

Hierarchical to: **No other components**
Dependencies: **EXT_MAC_SDC.1**

EXT_MAC_RCT.1.1
The System shall perform the following analysis function(s) on all application data captured and take the associated action(s) in response:
[

Analytical Function	*Associated Action*
a) For application control:	
i. Compare the attributes of any program attempting to make changes to an application on the endpoint with the Application Control rules to determine whether it has Updater permission.	*Allow only authorized applications (those with Updater permission) to make changes to applications on the endpoint.* *(If an application does not have the Updater permission it will be prevented from making any updates to applications on an endpoint)*
ii. Compare the attributes of any program attempting to execute (that is not contained in the whitelist inventory) with the Application Control rules.	*Allow execution of any program on the basis of checksum, certificate/publisher name or trusted directory, or deny execution of any program on the basis of checksum or name in accordance with the Application Control rules.*
iii. Compare the identifier of any program attempting to execute with the whitelist inventory.	*Allow execution of any program listed on the whitelist inventory* *If the program is not included in the whitelist inventory (and has not matched any of the Application Control rules) it will be denied.*
iv. When attempts are made to execute files (e.g. interpreters) check the execution control attribute-based rules.	*If a file is allowed to execute after the Application Control checks, then attribute based rules can be defined to block or monitor execution, for example, programs with atypical arguments, execution by certain users, or with specified parent process.*
b) For change control:	
i. Compare the name of any file that a process is attempting to delete, rename, create hard links for, modify contents of, append data to, truncate, change owner of, and create Alternate Data Stream for with those listed as write-protected	*Prevent deletion of, renaming of, creation of hard links for, modification of contents of, appending data to, truncation of, change of owner for, and creation of Alternate Data Stream for any file listed as write-protected*
ii. Compare the name of any file that a process is attempting to read, or execute script files against, with those listed as read-protected	*Deny reading of data in, and execution of script files against any file listed as read-protected*
iii. Compare the identifier for any registry key that a process is attempting to modify with those listed as write-protected	*Prevent modification of registry keys listed as write-protected*
c) For change control monitoring:	

i. Compare the change events to the include filters and exclude filters defined for change control monitoring	*Write the filtered change events to the change logs[7].*

].

[7] Where change control monitoring captures more change data than necessary for organisational needs, filters can be applied to specify which events should be included or excluded certain events, so that only those events to be included are actually written to the change log. Rules can be created to define the required include/exclude filters.

6.3 Security Assurance Requirements

This section defines the assurance requirements for the TOE. Assurance requirements are taken from the CC Part 3 and are EAL2 augmented with ALC_FLR.2. Table 14 – Assurance Requirements summarizes the requirements.

Table 14 – Assurance Requirements

Assurance Requirements	
Class ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
Class ALC : Life Cycle Support	ALC_CMC.2 Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_FLR.2 Flaw reporting procedures
Class ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
Class AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
Class ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing – sample
Class AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

7 TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

7.1 TOE Security Functions

The security functions are provided by the TOE to meet the security functional requirements. Each function is described in this section, and the related security functional requirements are given. This serves both to describe the security functions, and to provide a rationale that the security functions satisfy the necessary requirements.

Table 15 – Mapping of TOE Security Functions to Security Functional Requirements

TOE Security Function	SFR ID	Description
Security Audit	FAU_GEN.1	Audit data generation
	FAU_SAR.1	Audit review
	FAU_SAR.2	Restricted audit review
	FAU_SAR.3	Selectable audit review
Cryptographic Support	FCS_CKM.1(1)	Cryptographic key generation (MA)
	FCS_CKM.1(2)	Cryptographic key generation (ePO)
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1	Cryptographic operation
Identification and Authentication	FIA_ATD.1	User attribute definition
	FIA_UID.2	User identification before any action
	FIA_UAU.2	User authentication before any action
Security Management	FMT_MTD.1	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
Protection of TOE Security Functions	FPT_ITT.1	Basic internal TSF data transfer protection
McAfee Application and Change Control	EXT_MAC_SDC.1	Application and change control data collection
	EXT_MAC_RCT.1	Application and change control react

7.1.1 Security Audit

The TOE generates audit records for start-up/shutdown functions, Solidcore actions and all ePO administrator actions. The details of the SolidCore actions, sent from the endpoints, are recorded in the database. The records of SolidCore actions may also be viewed at the endpoint. The events associated with ePO administrator actions and start-up/shutdown functions are recorded in the ePO audit log. Authorized administrators can view, sort, and filter the audit records. The ePO-generated audit records can be filtered to present only failed actions, or only entries that are within a certain age. Solidcore-generated audit records can be filtered on the following fields:

- User who performed the action,
- Target object of the action,
- Computer on which the action was performed,
- Action timestamp, and
- Action type.

TOE Security Functional Requirements Satisfied: FAU_GEN.1, FAU_SAR.1, FAU_SAR.2, FAU_SAR.3.

7.1.2 Cryptographic Support

The TOE protects transmissions between the ePO and the McAfee Agent from disclosure by encrypting the transmissions under TLS. In FIPS mode, ePO uses OpenSSL 1.0.2t with FIPS Object Module 2.0.16 (FIPS 140-2 certificate #2398) for TLS 1.2. This is implemented using the Apache Server. McAfee Agent uses OpenSSL 1.0.2r with FIPS Object Module 2.0.16 (FIPS 140-2 certificate #2398) to provide cryptographic services for this link.

McAfee affirms that the cryptographic modules have been implemented in accordance with their FIPS 140 security policies, and when ePO and MA are configured in FIPS mode for TLS communication the cryptographic functions operate as intended. ePO and MA are tested by McAfee on all supported platforms.

TOE Security Functional Requirements Satisfied: FCS_CKM.1(1&2), FCS_CKM.4, FCS_COP.1

7.1.3 Identification and Authentication

User identification is enforced by the TOE. Users must log in to the ePO with a valid user name and password via a GUI before any access is granted by the TOE to TOE functions or data. When the credentials are presented by the user, ePO determines if the user name is defined and enabled. If not, the login process is terminated and the login GUI is redisplayed.

The password entered by the user is verified against the hashed version of the password stored in the database. If it is validated, the TOE grants access to authorized TOE functionality. If the password is not validated, the login GUI is redisplayed to the user.

For each defined user account, the following information is configured:

- User name

- Enabled or disabled
- Whether authentication for this user is to be performed by ePO or Windows (the evaluated configuration requires local ePO authentication for all users)
- Hashed copy of the password (in the evaluated configuration where local ePO authentication is configured),
- Permission sets granted to the user

Upon successful login and each consecutive action taken that causes a GUI refresh, the permissions are bound. Those attributes remain fixed until an action causes the GUI to refresh. If the attributes for a logged-in user are changed, those changes will not be bound to a subject until the next GUI action by that user.

TOE Security Functional Requirements Satisfied: FIA_ATD.1, FIA_UID.2, FIA_UAU.2.

7.1.4 Security Management

The TOE provides administrator support functionality that enables a user to configure and manage TOE components. Management of the TOE is performed via the ePO GUI. Management permissions are defined per-user.

The TOE provides functionality to manage the following TSF data:

- Dashboards
- Audit Log
- Permission Sets
- Queries and Reports
- Systems
- System Tree Access
- SolidCore General
- SolidCore Policy Permissions

The TOE maintains two types of roles: “Administrator” and users with selected permissions. A permission set is a group of permissions that can be granted to any users by assigning it to those users’ accounts. One or more permission sets can be assigned to any users who are not Administrators (Users assigned to the “administrator” permission set). Administrators are granted all permissions. Each user authorized for login to ePO must be defined within ePO. Only Administrators may perform ePO user account management functions (create, view, modify, and delete).

One or more permission sets may be associated with an account. Administrators are granted all permissions. Permissions exclusive to Administrators (that are not granted via permission sets) include:

- Create and delete user accounts
- Create, delete, and assign permission sets.

TOE Security Functional Requirements Satisfied: FMT_MTD.1, FMT_SMF.1, FMT_SMR.1.

7.1.5 Protection of the TSF

Communications between McAfee Agents and ePO take the form of XML messages. Communications can include policies to implement, properties collected from the Endpoint machine, event data gathered by the Solidcore application, or tasks to be run on the Endpoint. The messages are transferred via HTTPS. The TOE protects these transmissions between the ePO and the McAfee Agent from disclosure using TLS 1.2.

TOE Security Functional Requirements Satisfied: FPT_ITT.1.

7.1.6 McAfee Application and Change Control

The TOE provides Application Control and Change Control functionality for managed systems. It does this by collecting information about the program code, files, directories, and volumes that are to be protected. Each time a program attempts to execute, or a process or user attempts to modify a protected resource, the TOE analyzes the attempted action, and determines whether it should be allowed or not. It then takes the appropriate action.

7.1.6.1 Application Control

Application Control has to be deployed in Enabled Mode when operating in accordance with the evaluated configuration operational environment.

Application Control functionality prevents the execution of unauthorized program code and prevents unauthorized updates to applications on a managed system. Upon initial configuration, Application Control takes an initial snapshot of the software implemented on a managed system, and creates a whitelist inventory of the program code that exists at that time on the system. The listed program code includes binary executables such as '.exe' and '.dll' files, as well as scripts, such as '.bat', '.cmd', and '.vbs' files. This becomes the list of code that will be allowed to run on the managed system.

In addition, the administrator can configure Application Control rules to explicitly allow/deny the execution of programs on the managed system, and also to control what programs are permitted to make updates to application files on the managed system.

If a program attempts to execute and make updates to application files on the managed system, the program is compared to the set of programs with Updater Permission. If the program is an authorized Updater (has Updater permission) it is allowed to make changes to applications on the endpoint. Without Updater Permission the program attempting to make the updates is unable to make changes to the managed system.

The Application Control rules provide various mechanisms (Binaries, Publisher, Installer, Trusted Directory) by which to explicitly permit execution of a program on the basis of the program attributes. The methods are applied such that the file attributes are operated in the following order of precedence (with ban entries taking precedence over allow entries):

- Reputation (from TIE/GTI) – Not covered by the TOE configuration
- Checksum
- Certificate/Publisher

- Name
- Trusted Directory

The Application Control rules can also be used to explicitly deny execution of a program on the basis of the program name or checksum.
If a program does not match any of the Application Control rules, the TOE compares the program identifier with the list of identifiers collected in the whitelist inventory at initial configuration. If the program is listed on the whitelist, the TOE allows the program to execute.
If the program has not matched either one of the Application Control rules or an entry in the whitelist, the TOE stops the program from executing.

For protection from fileless malware and script-based attacks additional execution control attribute-based rules can be defined. If a file's execution is allowed after the Application Control checks, then attribute-based or granular rules, if any are defined, come into play. Specific rules can be defined using one or more attributes of the file (such as path, parent process, command line argument and user) to allow, block, or monitor the file. When multiple rules are matched for a particular scenario, allow rules have the highest precedence, followed by block and monitor rules, respectively.
Attribute-based rules can be defined to allow or block files based on context. For example, rules can be created to prevent execution of a file with atypical input arguments, or by a particular user, or limit execution to a particular parent process.

7.1.6.2 Change Control

Change Control functionality prevents specified reads or writes to files and directories on the managed systems. Critical files, directories, and volumes can be write-protected using the 'deny-write' feature of Solidcore Services. This renders the specified files as read only. Critical files, directories, and volumes can also be read-protected using the 'deny-read' feature of Solidcore Services. This enforces read-protection on specified files, directories, and volumes, and also denies the execution of script files that access read-protected files.

The TOE maintains a list of critical files, directories, volumes, and registry keys that are to be write-protected. If a process attempts to delete, rename, create hard links for, modify the contents of, append data to, truncate, change the owner of, or create Alternate Data Streams for a file that is listed as write-protected, the TOE will prevent the action from taking place.

The TOE also maintains a list of all critical files, directories, and volumes that are to be read-protected. If a process attempts to read files or execute script files against a file that is listed as read-protected, the TOE will prevent the action from taking place.

7.1.6.3 Change Control Monitoring

Change Control Monitoring functionality tracks change actions happening on the managed system. The TOE collects events indicating change actions on files, directories, network shares, and file attributes. It also collects events that indicate the starting and stopping of processes, and the success or failure of user logon or logoff attempts and user account management activities. The TOE then compares these events with the 'include' and 'exclude' filters defined by the administrator. If there is a match, then the TOE writes the specified events to the change logs for viewing by administrators.

TOE Security Functional Requirements Satisfied: EXT_MAC_SDC.1, EXT_MAC_RCT.1.

8 Rationale

8.1 Conformance Claims Rationale

This Security Target extends Part 2 and conforms to Part 3 of the Common Criteria Standard for Information Technology Security Evaluations, version 3.1 revision 4.

8.2 Security Objectives Rationale

This section provides a rationale for the existence of each threat, policy statement, and assumption included in the Security Target. Sections 8.2.1, 8.2.2 and 8.2.3 show that the mapping from the threats, polices, and assumptions to the security objectives is complete. The following discussion provides detailed evidence of coverage for each threat, policy, and assumption.

8.2.1 Security Objectives Rationale Relating to Threats

Table 16 shows the mapping of threats to security objectives.

Table 16 – Threats: Security Objectives Mapping

Threats	Objectives	Rationale
T.AUTHENTICATE An authorized user may be unaware of an inadvertent change to TOE data or functions they are authorized to modify.	O.AUDIT The TOE must record audit records for data accesses and use of the TOE functions on the management system.	O.AUDIT counters this threat by ensuring that the TOE tracks all management actions taken against the TOE.
	O.AUDIT_REVIEW The TOE must provide authorized administrators with the ability to review, order, and filter the audit trail.	O.AUDIT_REVIEW counters this threat by ensuring that administrators can review the audited changes to the TOE configuration.
	O.IDENTIFY The TOE must be able to identify and authenticate users prior to allowing access to TOE administrative functions and data.	O.IDENTIFY counters this threat by ensuring that only identified and authenticated users can access the TOE administrative functions and data.
T.COMPROMISE An unauthorized user may attempt to disclose, remove, destroy, or compromise the integrity of the data collected and produced by the TOE by bypassing a security mechanism.	O.ACCESS The TOE must allow authorized users to access only authorized TOE functions and data.	O.ACCESS counters this threat by ensuring that the TOE allows only authorized users access to the TOE functions and data.
	O.PROTECT The TOE must ensure the integrity of audit and system data by protecting it from unauthorized modifications and access during transfer.	O.PROTECT counters this threat by ensuring that the TOE protects the TOE data from unauthorized access during transfer.

Threats	Objectives	Rationale
T.PROTECT An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data, or inappropriately change the configuration of the TOE.	O.ACCESS The TOE must allow authorized users to access only authorized TOE functions and data.	O.ACCESS counters this threat by ensuring that the TOE protects the TOE functions and data from unauthorized access.
	O.EADMIN The TOE must include a set of functions that allow efficient management of its functions and data.	O.EADMIN counters this threat by ensuring that the TOE provides a means to effectively manage the TOE.
	O.PROTECT The TOE must ensure the integrity of audit and system data by protecting it from unauthorized modifications and access during transfer.	O.PROTECT counters this threat by ensuring that the TOE protects the TOE data from access by unauthorized users.
T.APP_CHG_CONTROL An attacker may be able to inappropriately change targeted objects or execute inappropriate software on the managed system without being detected.	O.COLLECT The TOE shall collect a list of objects that are to be protected and an inventory of allowable program code for the managed systems.	O.COLLECT counters this threat by ensuring that the TOE collects information about the managed systems to be used to determine whether given processes or changes should be allowed or disallowed.
	O.ANALYZE The TOE must apply analytical processes and information to derive conclusions about allowed and disallowed accesses to objects.	O.ANALYZE counters this threat by ensuring that the TOE applies analytical processes and information to derive conclusions about allowed and disallowed actions on the managed systems.
	O.REACT The TOE shall take appropriate action on all allowed and disallowed accesses to objects.	O.REACT counters this threat by ensuring that the TOE takes actions to prevent or allow changes or program executions on the managed systems.

Every Threat is mapped to one or more Objectives in the table above. This complete mapping demonstrates that the defined security objectives counter all defined threats.

8.2.2 Security Objectives Rationale Relating to Policies

There are no Policies defined for this Security Target. Therefore, there are no Security Objectives relating to policies.

8.2.3 Security Objectives Rationale Relating to Assumptions

Table 17 – Assumptions: Objectives Mapping

Assumptions	Objectives	Rationale
A.ACCESS The TOE has access to all the IT System data it needs to perform its functions.	OE.INTEROP The TOE is interoperable with the managed systems it monitors.	OE.INTEROP upholds this assumption by ensuring that the TOE can interoperate with the managed systems, thereby having access to all the system data it needs to perform its functions.
A.TIME The IT Environment will provide reliable timestamps for the TOE to use.	OE.TIME The TOE environment must provide reliable timestamps to the TOE.	OE.TIME upholds the assumption that the environment provides reliable timestamps to the TOE.
A.LOCATE The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	NOE.PHYSICAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy, and the hardware on which the TOE runs, are protected from any physical attack.	NOE.PHYSICAL upholds this assumption by ensuring that physical security is provided within the TOE environment to provide appropriate protection to the network resources.
A.PROTECT The TOE software critical to security policy enforcement, and the hardware on which it runs, will be protected from unauthorized physical modification.	NOE.PHYSICAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy, and the hardware on which the TOE runs, are protected from any physical attack.	NOE.PHYSICAL upholds this assumption by ensuring that the TOE environment provides protection from external interference or tampering.
A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.	NOE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	OE.MANAGE satisfies the assumption that competent individuals are assigned to manage the TOE and the TSF.
A.NOEVIL The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.	NOE.INSTALL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner that is consistent with IT security.	NOE.INSTALL upholds this assumption by ensuring that personnel installing, managing, and operating the TOE do so efficiently and correctly.
	NOE.PHYSICAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy, and the hardware on which the TOE runs, are	NOE.PHYSICAL upholds this assumption by ensuring that the users who install, manage, and operate the TOE do so in a manner that protects it from physical access by unauthorized personnel.

Assumptions	Objectives	Rationale
	protected from any physical attack.	
	NOE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	OE.MANAGE satisfies the assumption that the users who manage the TOE are non-hostile, appropriately trained and follow all guidance.
A.DYNAMIC The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.	OE.INTEROP The TOE is interoperable with the managed systems it monitors.	OE.INTEROP upholds this assumption by ensuring that the TOE interoperates with the managed systems, thereby allowing them to be managed by the TOE.
	NOE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	NOE.PERSON upholds this assumption by ensuring that only properly trained personnel are allowed to operate the TOE.

Every Assumption is mapped to one or more Objectives in the table above. This complete mapping demonstrates that the defined security objectives uphold all defined assumptions.

8.3 Rationale for Extended Security Functional Requirements

A class of EXT_MAC requirements was created to specifically address the Application Control and Change Control functionality of the TOE. The FAU: Security Audit class was used as a model for creating these requirements. The purpose of this class of requirements is to define the security functionality provided by the Solidcore Service of the TOE. There are no existing CC SFRs that can be used to appropriately describe this Solidcore functionality, so the extended components were created with wording that adequately captures the Solidcore functionality being claimed. These requirements have no dependencies outside their own class since the stated requirements embody all the necessary security functions. These requirements exhibit functionality that can be easily documented in the ADV assurance evidence and thus do not require any additional Assurance Documentation.

8.4 Rationale for Extended TOE Security Assurance Requirements

No extended TOE Security Assurance Requirements were defined for this Security Target.

8.5 Security Requirements Rationale

The following discussion provides detailed evidence of coverage for each security objective.

8.5.1 Rationale for Security Functional Requirements of the TOE Objectives

Table 18 – Objectives: SFRs Mapping

Objective	Requirements Addressing the Objective	Rationale
O.AUDIT The TOE must record audit records for data accesses and use of the TOE functions on the management system.	FAU_GEN.1 Audit data generation	The requirement meets this objective by ensuring that the TOE maintains a record of defined security-related events, including relevant details about the event.
O.ACCESS The TOE must allow authorized users to access only authorized TOE functions and data.	FAU_GEN.1 Audit data generation	The requirement meets this objective by providing audits of all management actions taken on the TOE for review by administrators.
	FAU_SAR.1 Audit review	The requirement meets this objective by providing the capability to review the audit trail of all management actions taken on the TOE.
	FAU_SAR.2 Restricted audit review	The requirement meets the objective by ensuring that the TOE allows only authorized administrators the ability to review the audit records.
	FAU_SAR.3 Selectable audit review	The requirement meets the objective by ensuring that the TOE provides only authorized administrators the ability to review, order, and filter the audit trail.
	FIA_ATD.1 User attribute definition	The requirement meets the objective by ensuring that the TOE maintains a list of security attributes belonging to individual users.
	FIA_UID.2 User identification before any action	The requirement meets the objective by ensuring that the TOE identifies all users prior to allowing them access to any TOE functions or data.
	FIA_UAU.2 User identification before any action	The requirement meets the objective by ensuring that the TOE authenticates all users prior to allowing them access to any TOE functions or data.
	FMT_MTD.1 Management of TSF data	The requirement meets the objective by ensuring that only authorized users are allowed access to TSF data.

Objective	Requirements Addressing the Objective	Rationale
	FMT_SMF.1 Specification of management functions	The requirement meets the objective by ensuring that only authorized administrators are allowed access to TSF functions and data.
	FMT_SMR.1 Security roles	The requirement meets the objective by ensuring that only users with authorized administrative roles are allowed access to TSF functions and data.
O.AUDIT_REVIEW The TOE must provide authorized administrators with the ability to review, order, and filter the audit trail.	FAU_SAR.1 Audit review	The requirement meets the objective by ensuring that the TOE provides the ability to review the audit trail.
	FAU_SAR.2 Restricted audit review	The requirement meets the objective by ensuring that the TOE allows authorized administrators the ability to review the audit records.
	FAU_SAR.3 Selectable audit review	The requirement meets the objective by ensuring that the TOE provides authorized administrators the ability to review, order, and filter the audit trail.
O.IDENTIFY The TOE must be able to identify and authenticate users prior to allowing access to TOE administrative functions and data.	FIA_UID.2 User identification before any action	The requirement meets the objective by ensuring that the TOE identifies all users prior to allowing them access to any TOE functions or data.
	FIA_UAU.2 User authentication before any action	The requirement meets the objective by ensuring that the TOE authenticates all users prior to allowing them access to any TOE functions or data.
O.EADMIN The TOE must include a set of functions that allow efficient management of its functions and data.	FMT_MTD.1 Management of TSF data	The requirement meets the objective by ensuring that the TOE provides a means to effectively manage the TOE data.
	FMT_SMF.1 Specification of management functions	The requirement meets the objective by ensuring that the TOE includes administrative functions to facilitate the management of the TSF.
	FMT_SMR.1 Security roles	The requirement meets the objective by ensuring that the TOE provides administrative roles to facilitate the management of the TSF.

Objective	Requirements Addressing the Objective	Rationale
O.PROTECT The TOE must ensure the integrity of audit and system data by protecting it from unauthorized modifications and access during transfer.	FCS_CKM.1 Cryptographic key generation	This requirement supports the objective by generating keys used to protect TSF data when it is transmitted between separate parts of the TOE.
	FCS_CKM.4 Cryptographic key destruction	This requirement supports the objective by destroying keys after use.
	FCS_COP.1 Cryptographic operation	The requirement meets the objective by ensuring that the TOE protects TSF data from disclosure when it is transmitted between separate parts of the TOE.
	FPT_ITT.1 Basic internal TSF data transfer protection	The requirement meets the objective by ensuring that the TOE protects TSF data from disclosure when it is transmitted between separate parts of the TOE.
O.COLLECT The TOE shall collect a list of objects that are to be protected and an inventory of allowable program code for the managed systems.	EXT_MAC_SDC.1 Application and change control data collection	The requirement meets this objective by ensuring that the TOE collects information about allowed and disallowed changes to objects and execution of programs on the managed systems.
O.ANALYZE The TOE must apply analytical processes and information to derive conclusions about allowed and disallowed accesses to objects.	EXT_MAC_RCT.1 Application and change control react	The requirement meets this objective by ensuring that the TOE analyzes the collected change control and application control events and actions.
O.REACT The TOE shall take appropriate action on all allowed and disallowed accesses to objects.	EXT_MAC_RCT.1 Application and change control react	The requirement meets this objective by ensuring that the TOE takes appropriate actions, as defined by policy, on all allowed and disallowed accesses to objects.

8.5.2 Security Assurance Requirements Rationale

EAL2 was chosen to provide a moderate level of assurance that is consistent with good commercial practices. As such, minimal additional tasks are placed upon the vendor assuming the vendor follows reasonable software engineering practices and can provide support to the evaluation for design and testing efforts. The chosen assurance level is appropriate with the threats defined for the TOE environment. While the System may monitor a hostile environment, the servers on which it is located are assumed to provide protection by employing measures appropriate to that environment. At EAL2,

the System will have incurred a search for obvious flaws to support its introduction into the protected environment.

The augmentation of ALC_FLR.2 was chosen to give greater assurance of the developer's on-going flaw remediation processes.

8.5.3 Dependency Rationale

This ST does satisfy all the requirement dependencies of the Common Criteria. Table 19 lists each requirement to which the TOE claims conformance with a dependency and indicates whether the dependent requirement was included. As the table indicates, all dependencies have been met.

Table 19 – Functional Requirements Dependencies

SFR ID	Dependencies	Dependency Met	Rationale
FAU_GEN.1	FPT_STM.1	Met by the environment	Although FPT_STM.1 is not included, the TOE Environment provides reliable timestamps to the TOE. An environmental objective states that the TOE will receive reliable timestamps, thereby satisfying this dependency.
FAU_SAR.1	FAU_GEN.1	✓	
FAU_SAR.2	FAU_SAR.1	✓	
FAU_SAR.3	FAU_SAR.1	✓	
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1]	✓	Met using FCS_COP.1
	FCS_CKM.4	✓	
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Met using FCS_CKM.1
FCS_COP.1	FCS_CKM.1	✓	
	FCS_CKM.4	✓	
FIA_ATD.1	No dependencies		
FIA_UID.2	No dependencies		
FIA_UAU.2	FIA_UID.1	✓	Although FIA_UID.1 is not included, FIA_UID.2, which is hierarchical to FIA_UID.1 is included. This satisfies this dependency.
FMT_MTD.1	FMT_SMF.1	✓	
	FMT_SMR.1	✓	

SFR ID	Dependencies	Dependency Met	Rationale
FMT_SMF.1	No dependencies		
FMT_SMR.1	FIA_UID.1	✓	Although FIA_UID.1 is not included, FIA_UID.2, which is hierarchical to FIA_UID.1 is included. This satisfies this dependency.
FPT_ITT.1	No dependencies		
EXT_MAC_SDC.1	No dependencies		
EXT_MAC_RCT.1	EXT_MAC_SDC.1	✓	

9 Acronyms

This section describes the acronyms.

Table 20 – Acronyms

Acronym	Definition
ACL	Access Control List
ADS	Alternate Data Stream
CC	Common Criteria
CEM	Common Evaluation Methodology
CLI	Command Line Interface
CM	Configuration Management
CPU	Central Processing Unit
DHE	Diffie Hellman Exchange
DNS	Domain Name System
EAL	Evaluation Assurance Level
ePO	ePolicy Orchestrator
FTP	File Transfer Protocol
GB	Gigabyte
GCM	Galois/Counter Mode
GHz	Gigahertz
GTI	Global Threat Intelligence
GUI	Graphical User Interface
HTTP	HyperText Transfer Protocol
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MB	Megabyte
MS	Microsoft
NFS	Network File Server
OS	Operating System
OSP	Organizational Security Policy
PP	Protection Profile
RAM	Random Access Memory

Acronym	Definition
RSD	Rogue System Detection
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
ST	Security Target
TIE	Threat Information eXchange
TCP/IP	Transmission Control Protocol/Internet Protocol
TOE	Target of Evaluation
TSF	TOE Security Functionality
UNC	Universal Naming Convention
VGA	Video Graphic Array