Check Point Endpoint Security Media Encryption

Security Target

Version 1.0
June 23, 2010

Prepared for:

5 Ha'Solelim St.

Tel Aviv, Israel 67897

Prepared By:

Science Applications International Corporation

Common Criteria Testing Laboratory

7125 Columbia Gateway Drive, Suite 300
Columbia, MD 21046

And By:

Metatron Security Services Ltd.

66 Yosef St.,
Modiin, Israel 71724

List of Tables

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization.

The Security Target contains the following additional sections:

- Section 2 – Target of Evaluation (TOE) Description
  This section gives an overview of the TOE, describes the TOE in terms of its physical and logical boundaries, and states the scope of the TOE.
- Section 3 – TOE Security Environment
  This section details the expectations of the environment and the threats that are countered by the TOE and IT environment.
- Section 4 – TOE Security Objectives
  This section details the security objectives of the TOE and IT environment.
- Section 5 – IT Security Requirements
  The section presents the security functional requirements (SFR) for the TOE and IT Environment that supports the TOE, and details the assurance requirements.
- Section 6 – TOE Summary Specification
  The section describes the security functions, represented in the TOE, that satisfy the security requirements.
- Section 7 – Protection Profile Claims
  This section presents any protection profile claims.
- Section 8 – Rationale

This section closes the ST with the justifications of the security objectives, requirements and TOE summary specifications as to their consistency, completeness, and suitability.

1.1. Security Target, TOE and CC Identification

ST Title – Check Point Endpoint Security Media Encryption Security Target

ST Version – Version 1.0

ST Date – June 23, 2010

TOE Identification – Endpoint Security Media Encryption 4.95 HFA 01 build 238

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 2.3, August 2005

1.2. ST Overview

Check Point Endpoint Security Media Encryption is a workstation security software product that provides centrally-managed control of workstation device interfaces. The product can be configured to prevent use of unauthorized devices, and to block introduction of executable code via workstation device ports. A removable media device encryption capability complements device access control, ensuring that only authorized users can access media contents.

The Target of Evaluation includes Endpoint Security Media Encryption Server software used to manage Endpoint Security Media Encryption endpoints. Administrators can publish flexible device access policies and review detailed audit logs of endpoint workstation device access events.

1.3. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, Version 2.3, August 2005.
    - Part 2 Conformant
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 2.3, August 2005.
    - Part 3 Conformant
    - Assurance Level: EAL 4 augmented with ALC_FLR.3 (systematic flaw remediation)
    - Strength of Function Claim: SOF-basic

1.4. Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

2. TOE Description

The Target of Evaluation (TOE) is the Check Point Endpoint Security Media Encryption[1] software product, version 4.95. ME can be used to control access to removable I/O devices, such as removable hard drives and USB memory devices, as well as removable media[2] (i.e., floppy disks, CDs, and DVDs).

ME is distributed as a component of the Check Point Endpoint Security Client R71 suite. The Endpoint Security Client suite consists of the following features:

Full Disk Encryption (FDE)	
Full Disk Encryption	Provides centrally managed, full disk encryption that protects data, operating systems and temporary files without relying on user interaction. Encryption is transparent to the user, who never needs to bother about what to encrypt or when.

Media Encryption (ME)	
Port Protection	Controls access to devices through all PC ports, such as IrDA, Com, USB, Firewire and LPT ports. This feature prevents users from connecting unauthorized devices to client machine ports, including hardware such as a modem and provides On/Off/read only protection.
Media Encryption	Encrypts and protects information stored on removable media, such as USB disks, external disk drives, CDs, DVDs, floppy disks. etc. Access is limited to authorized personnel with the appropriate password.

Secure Access (SA)	
Firewall	Protects endpoint systems by restricting both inbound and outbound traffic, ensuring that they are in a secure state before allowing access to the network and automatically enforcing policies that specify which programs are allowed to run on client machines.
Anti-Malware	Detects and removes viruses, spyware, and other malware based on a combination of signatures, behavior blockers, and heuristic analysis, featuring the highest detection rates and hourly signature updates through the SmartDefense™ update service.
VPN Client	Enables secure remote access to end users by encrypting and authenticating data transmitted during remote access sessions between endpoints and corporate network.

Together, the three components of the Endpoint Security Client suite: ME, FDE and SA provide a comprehensive, centrally-managed unified security policy enforcement solution that helps enterprises protect endpoints and data from a wide variety of threats, while minimizing end-user impact.

[1] Also referred to in this ST as “Media Encryption” or “ME”, for short. Where the ST refers to the “ME client”, it is referring to the client software installation on the protected workstation. Otherwise “ME” refers to the complete product installation, including both ME client and the Endpoint Security Media Encryption Server.

[2] Note that ‘removable media device’ is a subset of ‘removable I/O device’, both of which are distinct from ‘removable media’. Removable I/O devices are any devices that can be attached and detached, in its entirety, from a host workstation. Removable media devices are those removable I/O devices capable of storing data (e.g., the contents can potentially be written). Removable media includes floppy disks, CDs, and DVDs where the media itself is removable from a device attached to a host workstation.

Each of the three components is independent of the others, and requires a separate license to be installed. The TOE boundary includes only ME; however, the other components may be part of the IT environment of the TOE.

2.1. TOE Overview

The ME client runs in the context of Microsoft Windows 2000 and XP Professional, Windows Vista, and Windows 2000 Server Edition operating systems. When installed according to the available guidance, ME offers a number of security features designed to protect the host operating system as well as data stored on removable media devices and removable media. The protection features, most specifically, revolve around controlling access to removable I/O devices and removable media and encrypting data on removable media devices and removable media to ensure it remains protected even when taken out of the scope of ME control. The following list provides a more detailed summary of ME client features:

- Device Management
  - Access to removable I/O devices can be controlled so that full, read-only, or no access can be allowed to users of the workstation where the device is present.
  - Access can be controlled based on the model/brand of the device.
- Removable Media Management
  - Removable media devices can be digitally signed[3] in order to ensure integrity of the removable media device contents while out of the scope of control of ME.
  - Access to removable media device can be restricted so that only properly signed media can be accessed.
  - Access can be allowed to removable media devices that are not signed or where the signature is invalid after a configured action has been taken. ME can be configured to allow the user to delete content or alternately to invoke a configured scanning program (e.g., virus scanner) prior to (re)signing the removable media device and allowing access.
- Removable Media Encryption
  - The client can be configured to transparently encrypt the contents of removable media devices and removable media (not supported for floppy disks).
  - The client can be configured to place an application on the removable media device or removable media that can be run outside the scope of control of ME, providing access to the encrypted content given the applicable user password (used to derive the media encryption key).
- Detailed Auditing
  - The client can be configured to audit attempted security breaches (e.g., write access to a read-only device or media, access an unapproved device or media), removable media device or removable media file operations, etc.
  - For each event type, the client can be configured to ignore the event, register the event, or take immediate action (i.e., raise an alert).
  - The client stores audit records in a local database protected with a ME-generated hash, NTFS file permissions (configured by the client and enforced by the host operating system), and anti-tamper features of the client (discussed later). Stored audit records will be forwarded to an associated Endpoint Security Media Encryption Server at a configured interval, though immediate events are forwarded to the server as quickly as possible. Note that audit records cannot be reviewed by the client – they are available for review only on the server.

[3] Note that while the developer documentation refers to signatures and authentication, the applicable media integrity is verified using a combinatoric hash that represents the media contents. This ST refers to the hash alternately as a signature or digital checksum.

- Program Security Guard (PSG)
    - The PSG function of the client restricts access to configured file types. File types are configured based on their extensions and any attempts to create or modify files of the configured type will be prevented in general (i.e., not just when dealing with removable media).
    - PSG allows specific programs to be exempt from the file type restrictions otherwise imposed. This enables certain programs, such as a virus scanner or trusted installer programs, to create or modify otherwise restricted file types.

The Endpoint Security Media Encryption Server application runs in the context of one or more server hosts running the Microsoft Windows 2000 and 2003 Server Edition and Windows 2000 and XP Professional operating systems in order to facilitate the central management of numerous ME clients. The server provides administrators full control of the management/configuration interfaces where the settings of clients are determined. The server forwards updated configuration settings to ME clients and receives audit data generated on the associated clients so that it can be reviewed.

2.2. TOE Architecture

A given instance of the TOE consists of one or more Endpoint Security Media Encryption Server instances using a shared database installation (in the IT environment) and one or more ME client installations associated with that server installation.

The Endpoint Security Media Encryption Server application is installed as a service in the context of the Microsoft Windows 2000 and 2003 Server Edition operating system. TOE Administrators manage the application using the Endpoint Security Media Encryption Server Administration Console, a Microsoft Management Console (MMC) snap-in, installed either locally or remotely to the server. They can configure the policies to be enforced within the associated clients and manage audit data coming back from associated clients. A single SQL or MSDE database server, outside the TOE, is used by all server application instances in order to store client configuration data and audit records. Multiple Administration Consoles instances may be installed and used; however, only a single concurrent instance may be used in the evaluated configuration for updating the database.

When the ME clients are installed, they are configured with the IP addresses of the associated servers. They will use these addresses to contact a server, where a unique identifier will be created for identification of that client, in order to obtain the policy configuration it will enforce. Subsequently, the client will receive policy updates from the server at configured intervals or in response to a request for a policy update. In addition, the client will forward any audit records it has generated either at a configured interval or immediately depending on the nature of the event.

The ME clients consist of drivers installed as filters within the operating system kernel, a service to facilitate communication with the server, and some other applications to instantiate a limited set of pop-up and tray icon functions (if so configured). Once installed, the client is mostly transparent to the user. There is a tray icon where they can review the status of the product and perceive that it is operating and, depending on the specific configuration, the user may see pop-ups when specific security-relevant events occur (e.g., disallowed attempt to use a removable media device). Depending on policy settings received from the server, the user may be able to perform limited management activities, such as authorizing or importing (encrypting) removable media devices, and setting removable device off-line access passwords.

The ME client drivers serve to:

- allow the client to become aware of new removable I/O devices or removable media introduced to the operating system and to control (i.e., by blocking disallowed access attempts within the kernel) the ability to use those devices in accordance with its configured policies;
- allow the client to sign removable media devices each time its contents are changed and to verify the signature when removable media is introduced, and to take configured actions when the signature is not correct (e.g., invoke a third-party scanning program, allow offending content to be deleted, disallow access altogether);

- allow the client to intercept and encrypt and decrypt data between user mode applications and removable media devices or removable media;
- allow the client to intercept attempts to create or modify specific types of files in order to enforce (i.e., by blocking disallowed file operations) its configured PSG policies; and
- allow the client to monitor file and registry access attempts so that it can protect its own binary and data files and keys to ensure its own functions cannot be disabled or otherwise tampered with.

When the ME client observes security-relevant events, such as an attempted violation of its configured polices, it can create an audit record of that event. The audit records are stored in a database implemented within the client and forwarded to the associated Endpoint Security Media Encryption Server at a configured interval. The exception is that specific events can be configured to be reported immediately and when those occur the client will forward that audit record to the server as soon as possible regardless of the configured reporting interval.

As indicated above, the ME client enforces the policy configured on the associated server and delivered to the client. The client offers no interfaces to manage its own policies and controls access to its configuration data so that even an administrator on the client operating system has no ready means (e.g., to modify the configuration data directly) to change the configuration.

Communication between the Endpoint Security Media Encryption Server and the ME clients as well as communication with the Administration Console is based on the hosting Microsoft Windows operating system's Security Service Provider Interface (SSPI using the NTLM SSP) for providing user identification and authentication and for encryption of data in transit. The Endpoint Security Media Encryption Server uses operating system user account and group assignments for assigning users to management roles, for selecting the policy profile to be applied by the client, and for enforcing access control to encrypted media. User group assignments may also be synchronized with a Novell NDS directory.

2.2.1. Physical Boundaries

The TOE subsystems include:

- ME server: Endpoint Security Media Encryption Server application (service) and Administration Console (MMC snap-in)
- ME client: drivers, service, and applications

ME is licensed separately from the other components of the Check Point Endpoint Security Client R71 suite: FDE and SA. The ME license includes the Endpoint Security Media Encryption Server license. FDE and SA are delivered on the same installation media, and may be installed on the ME client workstation; however, they are not considered part of the TOE.

The IT Environment components include:

- Microsoft Windows operating systems as indicated above in section 2.1
- Microsoft MSSQL Server 2000/2005, MSSQL Express or MSDE[4] database server accessible to the Endpoint Security Media Encryption Server
- Optional Novell Client for user database synchronization with Novell NDS networks
- Optional data scanning software– see Appendix A – Supported Antivirus Solutions (pre-configured)
- Optional SMTP mail server for sending email alerts

[4] A copy of Microsoft SQL Server 2000 Desktop Engine (MSDE 2000) is bundled with the product; however, it is not considered part of the TOE, and the customer may choose to skip its installation and use an existing compatible database server if so desired.

- Optional Check Point WebRH Server R71 with ME WebRH Extension for Web-based key recovery

2.2.2. Logical Boundaries

The security functions that must be provided by the IT environment include:

- Security audit
  - Protection of audit data so that only authorized administrators can access (e.g., to review or delete) the recorded data.
- Identification and authentication
  - Authentication of end users and administrators.
- Protection of the TSF
  - Protection of TOE applications at rest (i.e., files) and during execution, as well as TOE configuration and audit data.
  - Protection of the communication between distributed TOE components.
  - Protection of the TOE-protected resources and invocation of TOE security functions when applicable so that the TOE security functions cannot be bypassed.
  - Provision of reliable timestamps to the TOE for use in audit records.

The security functions provided by the TOE, as further summarized below, include:

- Security audit
- Cryptographic support
- User Data Protection
- Identification and authentication
- Security management
- Protection of the TSF

2.2.2.1. Security audit

The TOE generates audit events for security relevant events that occur relative to the ME client. The client can be configured to ignore auditable events, log auditable events, or immediately report auditable events. Events, other than ignored events, are recorded within an audit log stored in a database implemented within the TOE and delivered to the associated Endpoint Security Media Encryption Server at a configured interval. Immediate events are reported to the associated server as soon as possible. Email alerts can be generated by the server for administrator-defined event types.

The audit records identify the event, event details, the user associated with the applicable operating system process (as queried from the underlying operating system), and the time the event occurred. When the event is delivered to the server the server also ensures that the client identity, which was created when the client was installed and is determined each time the client interacts with the server, is associated with the event. The server stores received events within an associated SQL or MSDE database and offers functions to review and delete audit events. Note that

the audit data can be reviewed outside the scope of the TOE as well using tools, of the user's choice, capable of interacting with the configured database. The TOE relies on the database to protect the audit data appropriately.

The auditable events include attempted security breaches (e.g., attempts to access an unapproved device), authorized device accesses, and actions on approved devices (e.g., files copied). Note that all of the audit generation occurs on the clients and no actions (e.g., security management) are currently audited relative to the server.

2.2.2.2. Cryptographic support

The TOE, both the server and client, contains an instance of a FIPS 140-2 Level 1 evaluated cryptographic module. The cryptographic module performs symmetric key encryption and decryption of removable media data and is used to digitally sign the contents of removable media. The cryptographic module is operated in FIPS mode according to its FIPS security policy. The FIPS 140-2 Level 1 validated software module is implemented as a Microsoft Windows driver. It is loaded into computing system memory and resides at the kernel mode level of the Windows operating system. The TOE clients use the algorithms that are configured through the TOE server.

2.2.2.3. User data protection

The primary security function of the TOE is to control access to removable I/O devices and removable media. Beyond broad controls that may restrict access altogether or limit access to read-only operations, the TOE can also sign removable media device contents and encrypt removable media device and removable media (specifically, writeable CDs and DVDs) contents. Encryption is used to ensure data is protected even when it is removed from the scope of control of the TOE. Signatures are used to allow the TOE to determine when removable media devices have been modified outside its scope of control and to take preventative actions (e.g., disallow access, allow offending content to be deleted prior to access, or invoke a third-party media scanner to examine the changed contents for potentially unacceptable content).

2.2.2.4. Identification and authentication

The TOE identifies each TOE user and administrator, relying on the IT environment to perform user authentication. In addition, the TOE identifies the user's client computer. The combination of user and computer identification is used to select the user's device access control profile.

2.2.2.5. Security management

The TOE provides interfaces that users can use to manage the configuration of the TOE security policies. Policies are configured on the TOE server and then distributed to the associated TOE clients for enforcement. Note that the TOE enforces management permission restrictions based on user identity and group associations established by the IT environment.

2.2.2.6. Protection of the TSF

The TOE client works with its host operating system to actively protect its own files and registry keys. It does this by filtering access to its own files and registry keys within the kernel of the operating system. Whenever an attempt is made to delete or modify any of the files or registry keys known to be associated with the TOE, the TOE will block that attempt. However, the TOE is dependent on the IT environment (operating system) to not provide unfiltered access to those files or registry keys.

2.3. TOE Documentation

Check Point offers a series of documents that describe the installation process for the TOE as well as guidance for subsequent use and administration of the applicable security features. Refer to Section 6 for information about these and other documentation associated with the TOE.

3. Security Environment

This section summarizes the threats addressed by the TOE and assumptions about the intended environment of the TOE.

3.1. Threats

T.ACCOUNTABILITY	A user may not be held accountable for their actions.
T.ADMIN_TOOLS	An authorized administrator may not have tools suitable to allow the effective management of the TOE security functions.
T.MASQUERADE	An unauthorized user, process, or external IT entity may masquerade as an authorized entity to gain access to data or TOE resources.
T.TSF_COMPROMISE	A malicious user may cause the TOE or its configuration data to be inappropriately accessed (viewed, modified or deleted).
T.UNAUTH_ACCESS	A user may gain unauthorized access (view, modify, delete) to removable I/O devices or data stored within removable media devices.

3.2. Assumptions

A.LOCATE	The TOE server component will be located within controlled access facilities, which will prevent unauthorized physical access.
A.NO_EVIL	The TOE will be installed, configured, managed and maintained in accordance with its guidance documentation.

4. Security Objectives

This section summarizes the security objectives for the TOE and its environment.

4.1. Security Objectives for the TOE

O.AUDIT_GENERATION	The TOE will provide the capability create records of security relevant events associated with users.
O.AUDIT_REVIEW	The TOE will provide the capability to view audit information.
O.DEV_ACCESS	The TOE will provide the means to control access to removable I/O devices.
O.DEV_INTEG	The TOE will provide the means to measure the integrity of removable media contents and to limit access to the contents when the integrity has been violated.
O.ENCRYPTION	The TOE will provide the means to encrypt the content of removable media.
O.MANAGE	The TOE will allow administrators to effectively manage the TOE and its security functions.
O.ADMIN_ROLE	The TOE will provide authorized administrator roles to isolate and control administrative actions.
O.TOE_PROTECTION	The TOE will protect the TOE client and its assets from external interference, tampering, or bypass attempts.

4.2. Security Objectives for the IT Environment

OE.TIME_STAMPS	The IT Environment will provide a reliable source of timestamps to the TOE.
OE.TOE_ACCESS	The IT Environment will ensure that users gain only authorized access to the TOE and to the resources that the TOE controls.
OE.TOE_COMMS	The IT Environment will provide a means of protecting communication between distributed TOE components.
OE.TOE_PROTECTION	The IT Environment will protect the TOE and its assets from external interference, tampering, or bypass attempts.

4.3. Security Objectives for the Non-IT Environment

OE.CONFIG	The TOE will be installed, configured, managed and maintained in accordance with its guidance documentation.
OE.PHYCAL	The TOE server will be located within controlled access facilities, which will prevent unauthorized physical access.

5. IT Security Requirements

This section defines the security functional requirements for the TOE as well as the security assurance requirements against which the TOE has been evaluated. All of the requirements have been copied from version 2.3 of the applicable Common Criteria documents.

5.1. TOE Security Functional Requirements

The following table describes the SFRs that are satisfied by the TOE.

The TOE satisfies a minimum strength of function 'SOF-basic'. The only applicable (i.e., probabilistic or permutational) security functional requirements are FCS_COP.1, in the context of encryption of removable media or removable media devices using a password to derive the media encryption key, and FDP_ACF.1, in relation to the generation of removable media device digital signatures.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit data generation
	FAU_SAR.1: Audit review
	FAU_SAR.3: Selectable audit review
	FAU_SEL.1: Selective audit
FCS: Cryptographic support	FCS_CKM.1: Cryptographic key generation
	FCS_CKM.4: Cryptographic key destruction
	FCS_COP.1: Cryptographic operation
FDP: User data protection	FDP_ACC.1: Subset access control
	FDP_ACF.1: Security attribute based access control
FIA: Identification and authentication	FIA_ATD.1: User attribute definition
	FIA_UID.2: User identification before any action
	FIA_USB.1: User-subject binding
FMT: Security management	FMT_MOF.1: Management of security functions behavior
	FMT_MSA.1a: Management of security attributes
	FMT_MSA.1b: Management of security attributes
	FMT_MSA.1c: Management of security attributes
	FMT_MSA.3: Static attribute initialization
	FMT_MTD.1: Management of TSF data
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.1: Security Roles
FPT: Protection of the TSF	FPT_RVM.1a: Non-bypassability of the TSP
	FPT_SEP.1a: TSF domain separation

Table 1 - TOE Security Functional Components

5.1.1. Security audit (FAU)

5.1.1.1. Audit data generation (FAU_GEN.1)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events: a) Start-up and shutdown of the audit functions; b) All auditable events for the [***not specified***] level of audit; and c) [**access attempts subject to the device access control policy**].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [**no additional information**].

5.1.1.2. Audit review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide [**administrators**] with the capability to read [**all data**] 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.

5.1.1.3. Selectable audit review (FAU_SAR.3)

FAU_SAR.3.1 The TSF shall provide the ability to perform [***searches***] of audit data based on [**any content within audit records**].

5.1.1.4. Selective audit (FAU_SEL.1)

FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) [*event type*]
b) [no other attributes].

5.1.2. Cryptographic support (FCS)

5.1.2.1. Cryptographic key generation (FCS_CKM.1)

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**ANSI X9.31-based PRNG**] and specified cryptographic key sizes [**256 bits (for AES)**] that meet the following: [**FIPS 140-2 level 1**].

5.1.2.2. Cryptographic key destruction (FCS_CKM.4)

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**cryptographic key zeroization method**] that meets the following: [**FIPS 140-2 level 1**].

5.1.2.3. Cryptographic operation (FCS_COP.1)

FCS_COP.1.1 The TSF shall perform [**encryption and decryption, key wrap**] in accordance with a specified cryptographic algorithm [**AES**] and cryptographic key sizes [

a) encryption and decryption: 256 bits;
b) key wrap for automatic access: 256 bits; and
c) key wrap for password-protected access: 128 bits]

that meet the following: [**FIPS 197 (AES) in CBC mode**].

5.1.3. User data protection (FDP)

5.1.3.1. Subset access control (FDP_ACC.1)

FDP_ACC.1.1 The TSF shall enforce the [**Device Access Control SFP**] on [

a) subjects: client workstation processes;
b) objects: removable I/O (including media) devices, removable media (floppy disks, CDs, and DVDs) and workstation files; and,
c) operations: read and write access to removable I/O devices and removable media, creation and modification of workstation files].

5.1.3.2. Security attribute based access control (FDP_ACF.1)

FDP_ACF.1.1 The TSF shall enforce the [**Device Access Control SFP**] to objects based on the following: [

a) subject attributes: user identifier, profile (subject security settings), process name;
b) object attributes:

1) **for all removable I/O devices and removable media: device identity, device or media type, and media device digital signature;**
2) **for encrypted removable I/O devices and encrypted removable media: media owner (site identifier, user identifier), offline mode passwords (full access, read-only); and,**
3) **for workstation files: file type**].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

Device Manager Rules

a) the subject's profile must allow access to that device identity or type;

b) if the removable I/O device or removable media is not encrypted, the subject's profile may allow full access to the device or media (including write) or restrict access to read only or to read only and execute operation types;

Encryption Policy Manager Rule

c) if the removable media device or removable media is encrypted:

1) **the subject's profile must allow automatic access (available only when the ME client is connected online to the Endpoint Security Media Encryption Server), password-protected access, or challenge/response access to encrypted media;**
2) **automatic access is controlled in the subject's profile according to the media owner user identifier, as an ordered set of access authorizations:**
   i. **access may be explicitly allowed or denied if the media owner user identifier matches the subject's user identifier;**
   ii. **access may be explicitly allowed or denied if the media owner user identifier is associated in the TOE configuration with a user group that is allowed access in the subject's profile; and,**
   iii. **access authorization may be restricted to read only access;**
3) **password-protected access is controlled in the subject's profile according to the media owner site identifier, and to the media offline mode passwords:**
   i. **the media owner site identifier must be explicitly allowed in the profile;**
   ii. **the user is prompted for a password and must present either the correct full access password or the correct read-only access password; and,**
   iii. **access is restricted to read only if the read-only access password is presented;**
4) **a challenge/response offline authenticator generated by a key recovery officer in response to a device or media-specific challenge provides full access to the removable media device or removable media;**

Removable Media Manager Rule

d) a given removable media device or floppy disk with a missing or invalid media digital signature can be accessed only if the subject's profile does not require signature verification; and,

Program Security Guard Rule

e) a given workstation file can be created or modified only if its type is configured in the subject's profile to allow creation or modification].

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [

a) processes running as local SYSTEM processes, TOE processes, and supported antivirus applications[5] are exempt from the Removable Media Manager Rule; and,

b) if the workstation file is being modified or created by a subject whose process name is explicitly identified in the subject's profile as being allowed to modify or create restricted workstation file types, then the operation will be always be allowed;

c) the Device Manager, Removable Media Manager, and/or Program Security Guard rules may be disabled by an authorized user in accordance with FMT_MOF.1; and

d) a user that is also associated with the key recovery officer role is allowed automatic access to all encrypted media (available only when connected online)].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the [**no additional rules**].

FDP_ACF.1.5 The TSF shall provide the following:

- **a) each time the contents of a removable media device that can store data or floppy disk are modified, the TOE will digitally sign the contents and record that signature on the device;**
- **b) if a digital signature check fails, the subject's profile can be defined with any of the following actions:**
  - **1) allow the user to explicitly authorize (and re-sign) the media;**
  - **2) scan the media for executable content and optionally allow the user to delete any such content found in order to allow automatic authorization and re-signature; or,**
  - **3) invoke a configured third party application[5] and automatically authorize and re-sign the media if it returns a successful scan result; and,**
- **c) if the removable media device or removable media type is configured to be encrypted the contents will be automatically encrypted[6] during write operations and decrypted during read operations, in accordance with FCS_COP.1.**

Application note: FDP_ACF.1.5 was added to address actions taken by the TOE, e.g., to transform data, in conjunction with the access control policy. Note that a similar element is included within the CC information flow requirements.

5.1.4. Identification and authentication (FIA)

5.1.4.1. User attribute definition (FIA_ATD.1)

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [

- **a) user name;**
- **b) user identifier;**
- **c) group membership(s); and**
- **d) profile**].

5.1.4.2. User identification before any action (FIA_UID.2)

FIA_UID.2.1 The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

Application note: ME recognizes two types of users: human users, and client computers. When a client computer connects to the server, the server will always identify the client computer, and will also identify the human user if the IT environment (either Windows Domain Server or Novell Directory Server) has authenticated the user. The user identifier is unavailable to the TOE when the user is logged on to a local workstation user account.

5.1.4.3. User-subject binding (FIA_USB.1)

FIA_USB.1.1 The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [**user name, user identifier, computer name, profile**].

FIA_USB.1.2 The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [

- **a) the subject shall be associated with the computer name, and with the user name and user identifier if these are available from the IT environment;**
- **b) the subject's profile shall be selected as follows:**
  - **1) if the user identifier is unavailable, the computer profile;**

[5] The list of third-party content scanning applications (not considered part of the TOE) supported in the evaluated configuration is given in Appendix A – Supported Antivirus Solutions (pre-configured).

[6] Encryption is limited to removable media devices and the following removable media: CDs and DVDs. DVD encryption is supported only on the Windows Vista operating system. Also note that encryption of CDs and DVDs requires the use of the CD and DVD writing functions provided in the host operation system and will not work in conjunction with any other CD or DVD authoring tools. Other CD or DVD authoring tools will be blocked from writing to the media if only encrypted access is allowed to the media.

2) **else if the ME client is connected online to the ME server, a combination of the user and computer profiles;**
3) **else (when offline), as determined by the computer profile, either:**
   a. **a cached version of the user profile stored on the client computer; or**
   b. **a specially defined offline user profile; or (if neither of these is available)**
   c. **the computer profile**].

FIA_USB.1.3 The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [**no rules allowing change of attributes**].

5.1.5. Security management (FMT)

5.1.5.1. Management of security functions behaviour (FMT_MOF.1)

FMT_MOF.1.1 The TSF shall restrict the ability to [***determine the behaviour of, disable, enable, modify the behaviour of***] the functions [**all security functions**] to [**administrators and to users with appropriate profile authorizations**].

5.1.5.2. Management of security attributes (FMT_MSA.1a)

FMT_MSA.1a.1 The TSF shall enforce the [**Device Access Control SFP**] to restrict the ability to [***modify***] the security attributes [**encryption of removable media and removable media devices, media owner user identifier, offline mode access passwords and media digital signatures**] to [**users with appropriate profile authorizations**].

5.1.5.3. Management of security attributes (FMT_MSA.1b)

FMT_MSA.1b.1 The TSF shall enforce the [**Device Access Control SFP**] to restrict the ability to [***[generate]***] the security attributes [**challenge/response authenticators**] to [**key recovery officers**].

5.1.5.4. Management of security attributes (FMT_MSA.1c)

FMT_MSA.1c.1 The TSF shall enforce the [**Device Access Control SFP**] to restrict the ability to [***modify***] the security attributes [**all other security attributes**] to [**administrators**].

Application note: In FMT_MSA.1c, the assignment 'all other security attributes' refers to all Device Access Control SFP security attributes not covered by FMT_MSA.1a or FMT_MSA.1b.

5.1.5.5. Static attribute initialization (FMT_MSA.3)

FMT_MSA.3.1 The TSF shall enforce the [**Device Access Control SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [**administrators or users with appropriate profile authorizations**] to specify alternative initial values to override the default values when an object or information is created.

5.1.5.6. Management of TSF data (FMT_MTD.1)

FMT_MTD.1.1 The TSF shall restrict the ability to [***query, modify, delete***] the [**TSF data**] to [**administrators**].

5.1.5.7. Specification of Management Functions (FMT_SMF.1)

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

a) define the Device Access Control SFP for TOE clients;
b) modify user and group settings;
c) authorize removable media devices;
d) encrypt and decrypt removable media devices and removable media;
e) generate challenge/response authenticators;
f) change media offline mode passwords;
g) define the security audit policy configuration; and

h) archive audit records].

5.1.5.8. Security Roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles [**user, administrator, key recovery officer**].
FMT_SMR.1.2 The TSF shall be able to associate users with roles.

5.1.6. Protection of the TSF (FPT)

5.1.6.1. Non-bypassability of the TSP (FPT_RVM.1a)

FPT_RVM.1a.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.1.6.2. TSF domain separation (FPT_SEP.1a)

FPT_SEP.1a.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.
FPT_SEP.1a.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

5.2. IT Environment Security Functional Requirements

The following table describes the SFRs that are to be satisfied by the IT environment of the TOE.

Requirement Class	Requirement Component
FAU: Security audit	FAU_STG.1: Protected audit trail storage
FIA: Identification and Authentication	FIA_UAU.1: Timing of authentication
FPT: Protection of the TSF	FPT_ITT.1: Basic internal TSF data transfer protection
	FPT_RVM.1b: Non-bypassability of the TSP
	FPT_SEP.1b: TSF domain separation
	FPT_STM.1: Reliable time stamps

Table 2 - IT Environment Security Functional Components.

5.2.1. Security audit (FAU)

5.2.1.1. Protected audit trail storage (FAU_STG.1)

FAU_STG.1.1 The **IT environment** ~~TSF~~ shall protect the stored audit records from unauthorised deletion.
FAU_STG.1.2 The **IT environment** ~~TSF~~ shall be able to [***prevent***] unauthorised modifications to the audit records in the audit trail.

5.2.2. Identification and authentication (FIA)

5.2.2.1. Timing of authentication (FIA_UAU.1)

FIA_UAU.1.1 The **IT environment** ~~TSF~~ shall allow [**access to unencrypted media, password-protected access to encrypted removable media devices and removable media**] on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2 The **IT environment** ~~TSF~~ shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

5.2.3. Protection of the TSF (FPT)

5.2.3.1. Basic internal TSF data transfer protection (FPT_ITT.1)

FPT_ITT.1.1 The **IT environment** ~~TSF~~ shall protect TSF data from [***disclosure, modification***] when it is transmitted between separate parts of the TOE.

5.2.3.2. Non-bypassability of the TSP (FPT_RVM.1b)

FPT_RVM.1b.1 The **IT environment** ~~TSF~~ shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.2.3.3. TSF domain separation (FPT_SEP.1b)

FPT_SEP.1b.1 The **IT environment** ~~TSF~~ shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1b.2 The **IT environment** ~~TSF~~ shall enforce separation between the security domains of subjects in the TSC.

5.2.3.4. Reliable time stamps (FPT_STM.1)

FPT_STM.1.1 The **IT environment** ~~TSF~~ shall be able to provide reliable time stamps for **the TSF's** ~~its own~~ use.

5.3. TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 4 components as specified in Part 3 of the Common Criteria, augmented by the Part 3 component ALC_FLR.3. No operations are applied to the assurance components.

Requirement Class	Requirement Component
ACM: Configuration management	ACM_AUT.1: Partial CM automation
	ACM_CAP.4: Generation support and acceptance procedures
	ACM_SCP.2: Problem tracking CM coverage
ADO: Delivery and operation	ADO_DEL.2: Detection of modification
	ADO_IGS.1: Installation, generation, and start-up procedures
ADV: Development	ADV_FSP.2: Fully defined external interfaces
	ADV_HLD.2: Security enforcing high-level design
	ADV_IMP.1: Subset of the implementation of the TSF
	ADV_LLD.1: Descriptive low-level design
	ADV_RCR.1: Informal correspondence demonstration
	ADV_SPM.1: Informal TOE security policy model
AGD: Guidance documents	AGD_ADM.1: Administrator guidance
	AGD_USR.1: User guidance
ALC: Life cycle support	ALC_DVS.1: Identification of security measures
	ALC_FLR.3: Systematic flaw remediation
	ALC_LCD.1: Developer defined life-cycle model
	ALC_TAT.1: Well-defined development tools
ATE: Tests	ATE_COV.2: Analysis of coverage
	ATE_DPT.1: Testing: high-level design
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing - sample
AVA: Vulnerability assessment	AVA_MSU.2: Validation of analysis
	AVA_SOF.1: Strength of TOE security function evaluation
	AVA_VLA.2: Independent vulnerability analysis

Table 3 - Assurance Components

5.3.1. Configuration management (ACM)

5.3.1.1. Partial CM automation (ACM_AUT.1)

ACM_AUT.1.1d The developer shall use a CM system.
ACM_AUT.1.2d The developer shall provide a CM plan.
ACM_AUT.1.1c The CM system shall provide an automated means by which only authorised changes are made to the TOE implementation representation.
ACM_AUT.1.2c The CM system shall provide an automated means to support the generation of the TOE.
ACM_AUT.1.3c The CM plan shall describe the automated tools used in the CM system.
ACM_AUT.1.4c The CM plan shall describe how the automated tools are used in the CM system.
ACM_AUT.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.1.2. Generation support and acceptance procedures (ACM_CAP.4)

ACM_CAP.4.1d The developer shall provide a reference for the TOE.
ACM_CAP.4.2d The developer shall use a CM system.
ACM_CAP.4.3d The developer shall provide CM documentation.
ACM_CAP.4.1c The reference for the TOE shall be unique to each version of the TOE.
ACM_CAP.4.2c The TOE shall be labelled with its reference.
ACM_CAP.4.3c The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.
ACM_CAP.4.4c The configuration list shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.4.5c The configuration list shall describe the configuration items that comprise the TOE.
ACM_CAP.4.6c The CM documentation shall describe the method used to uniquely identify the configuration items that comprise the TOE.
ACM_CAP.4.7c The CM system shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.4.8c The CM plan shall describe how the CM system is used.
ACM_CAP.4.9c The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
ACM_CAP.4.10c The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
ACM_CAP.4.11c The CM system shall provide measures such that only authorised changes are made to the configuration items.
ACM_CAP.4.12c The CM system shall support the generation of the TOE.
ACM_CAP.4.13c The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.
ACM_CAP.4.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.1.3. Problem tracking CM coverage (ACM_SCP.2)

ACM_SCP.2.1d The developer shall provide a list of configuration items for the TOE.
ACM_SCP.2.1c The list of configuration items shall include the following: implementation representation; security flaws; and the evaluation evidence required by the assurance components in the ST.
ACM_SCP.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2. Delivery and operation (ADO)

5.3.2.1. Detection of modification (ADO_DEL.2)

ADO_DEL.2.1d The developer shall document procedures for delivery of the TOE or parts of it to the user.
ADO_DEL.2.2d The developer shall use the delivery procedures.
ADO_DEL.2.1c The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

ADO_DEL.2.2c The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer's master copy and the version received at the user site.

ADO_DEL.2.3c The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

ADO_DEL.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2.2. Installation, generation, and start-up procedures (ADO_IGS.1)

ADO_IGS.1.1d The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

ADO_IGS.1.1c The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.

ADO_IGS.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1.2e The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

5.3.3. Development (ADV)

5.3.3.1. Fully defined external interfaces (ADV_FSP.2)

ADV_FSP.2.1d The developer shall provide a functional specification.

ADV_FSP.2.1c The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.2.2c The functional specification shall be internally consistent.

ADV_FSP.2.3c The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.

ADV_FSP.2.4c The functional specification shall completely represent the TSF.

ADV_FSP.2.5c The functional specification shall include rationale that the TSF is completely represented.

ADV_FSP.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.2.2e The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.2. Security enforcing high-level design (ADV_HLD.2)

ADV_HLD.2.1d The developer shall provide the high-level design of the TSF.

ADV_HLD.2.1c The presentation of the high-level design shall be informal.

ADV_HLD.2.2c The high-level design shall be internally consistent.

ADV_HLD.2.3c The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.2.4c The high-level design shall describe the security functionality provided by each subsystem of the TSF.

ADV_HLD.2.5c The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

ADV_HLD.3.0c The high-level design shall identify all interfaces to the subsystems of the TSF.

ADV_HLD.2.7c The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

ADV_HLD.2.8c The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_HLD.2.9c The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.

ADV_HLD.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_HLD.2.2e The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.3. Subset of the implementation of the TSF (ADV_IMP.1)

ADV_IMP.1.1d The developer shall provide the implementation representation for a selected subset of the TSF.
ADV_IMP.1.1c The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.
ADV_IMP.1.2c The implementation representation shall be internally consistent.
ADV_IMP.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_IMP.1.2e The evaluator shall determine that the least abstract TSF representation provided is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.4. Descriptive low-level design (ADV_LLD.1)

ADV_LLD.1.1d The developer shall provide the low-level design of the TSF.
ADV_LLD.1.1c The presentation of the low-level design shall be informal.
ADV_LLD.1.2c The low-level design shall be internally consistent.
ADV_LLD.1.3c The low-level design shall describe the TSF in terms of modules.
ADV_LLD.1.4c The low-level design shall describe the purpose of each module.
ADV_LLD.1.5c The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.
ADV_LLD.1.6c The low-level design shall describe how each TSP-enforcing function is provided.
ADV_LLD.1.7c The low-level design shall identify all interfaces to the modules of the TSF.
ADV_LLD.1.8c The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.
ADV_LLD.1.9c The low-level design shall describe the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.
ADV_LLD.1.10c The low-level design shall describe the separation of the TOE into TSP-enforcing and other modules.
ADV_LLD.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_LLD.1.2e The evaluator shall determine that the low-level design is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.5. Informal correspondence demonstration (ADV_RCR.1)

ADV_RCR.1.1d The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.
ADV_RCR.1.1c For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.
ADV_RCR.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.3.6. Informal TOE security policy model (ADV_SPM.1)

ADV_SPM.1.1d The developer shall provide a TSP model.
ADV_SPM.1.2d The developer shall demonstrate correspondence between the functional specification and the TSP model.
ADV_SPM.1.1c The TSP model shall be informal.
ADV_SPM.1.2c The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.
ADV_SPM.1.3c The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4c The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

ADV_SPM.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4. Guidance documents (AGD)

5.3.4.1. Administrator guidance (AGD_ADM.1)

AGD_ADM.1.1d The developer shall provide administrator guidance addressed to system administrative personnel.

AGD_ADM.1.1c The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2c The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3c The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4c The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE.

AGD_ADM.1.5c The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6c The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7c The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8c The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

AGD_ADM.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.2. User guidance (AGD_USR.1)

AGD_USR.1.1d The developer shall provide user guidance.

AGD_USR.1.1c The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2c The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3c The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4c The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment.

AGD_USR.1.5c The user guidance shall be consistent with all other documentation supplied for evaluation.

AGD_USR.1.6c The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

AGD_USR.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.5. Life cycle support (ALC)

5.3.5.1. Identification of security measures (ALC_DVS.1)

ALC_DVS.1.1d The developer shall produce development security documentation.

ALC_DVS.1.1c The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.2c The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.
ALC_DVS.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ALC_DVS.1.2e The evaluator shall confirm that the security measures are being applied.

5.3.5.2. Systematic flaw remediation (ALC_FLR.3)

ALC_FLR.3.1d The developer shall provide flaw remediation procedures addressed to TOE developers.
ALC_FLR.3.2d The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.
ALC_FLR.3.3d The developer shall provide flaw remediation guidance addressed to TOE users.
ALC_FLR.3.1c The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.
ALC_FLR.3.2c The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.
ALC_FLR.3.3c The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.
ALC_FLR.3.4c The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.
ALC_FLR.3.5c The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.
ALC_FLR.3.6c The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.
ALC_FLR.3.7c The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.
ALC_FLR.3.8c The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.
ALC_FLR.3.9c The flaw remediation procedures shall include a procedure requiring timely responses for the automatic distribution of security flaw reports and the associated corrections to registered users who might be affected by the security flaw.
ALC_FLR.3.10c The flaw remediation guidance shall describe a means by which TOE users may register with the developer, to be eligible to receive security flaw reports and corrections.
ALC_FLR.3.11c The flaw remediation guidance shall identify the specific points of contact for all reports and enquiries about security issues involving the TOE.
ALC_FLR.3.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.5.3. Developer defined life-cycle model (ALC_LCD.1)

ALC_LCD.1.1d The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.
ALC_LCD.1.2d The developer shall provide life-cycle definition documentation.
ALC_LCD.1.1c The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.
ALC_LCD.1.2c The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.
ALC_LCD.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.5.4. Well-defined development tools (ALC_TAT.1)

ALC_TAT.1.1d The developer shall identify the development tools being used for the TOE.
ALC_TAT.1.2d The developer shall document the selected implementation-dependent options of the development tools.
ALC_TAT.1.1c All development tools used for implementation shall be well-defined.
ALC_TAT.1.2c The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ALC_TAT.1.3c The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

ALC_TAT.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6. Tests (ATE)

5.3.6.1. Analysis of coverage (ATE_COV.2)

ATE_COV.2.1d The developer shall provide an analysis of the test coverage.

ATE_COV.2.1c The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

ATE_COV.2.2c The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

ATE_COV.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.2. Testing: high-level design (ATE_DPT.1)

ATE_DPT.1.1d The developer shall provide the analysis of the depth of testing.

ATE_DPT.1.1c The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

ATE_DPT.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.3. Functional testing (ATE_FUN.1)

ATE_FUN.1.1d The developer shall test the TSF and document the results.

ATE_FUN.1.2d The developer shall provide test documentation.

ATE_FUN.1.1c The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2c The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3c The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4c The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5c The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

ATE_FUN.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.4. Independent testing - sample (ATE_IND.2)

ATE_IND.2.1d The developer shall provide the TOE for testing.

ATE_IND.2.1c The TOE shall be suitable for testing.

ATE_IND.2.2c The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_IND.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2e The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

ATE_IND.2.3e The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

5.3.7. Vulnerability assessment (AVA)

5.3.7.1. Validation of analysis (AVA_MSU.2)

AVA_MSU.2.1d The developer shall provide guidance documentation.
AVA_MSU.2.2d The developer shall document an analysis of the guidance documentation.
AVA_MSU.2.1c The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.
AVA_MSU.2.2c The guidance documentation shall be complete, clear, consistent and reasonable.
AVA_MSU.2.3c The guidance documentation shall list all assumptions about the intended environment.
AVA_MSU.2.4c The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).
AVA_MSU.2.5c The analysis documentation shall demonstrate that the guidance documentation is complete.
AVA_MSU.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_MSU.2.2e The evaluator shall repeat all configuration and installation procedures, and other procedures selectively, to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.
AVA_MSU.2.3e The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.
AVA_MSU.2.4e The evaluator shall confirm that the analysis documentation shows that guidance is provided for secure operation in all modes of operation of the TOE.

5.3.7.2. Strength of TOE security function evaluation (AVA_SOF.1)

AVA_SOF.1.1d The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.
AVA_SOF.1.1c For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.
AVA_SOF.1.2c For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.
AVA_SOF.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_SOF.1.2e The evaluator shall confirm that the strength claims are correct.

5.3.7.3. Independent vulnerability analysis (AVA_VLA.2)

AVA_VLA.2.1d The developer shall perform a vulnerability analysis.
AVA_VLA.2.2d The developer shall provide vulnerability analysis documentation.
AVA_VLA.2.1c The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for ways in which a user can violate the TSP.
AVA_VLA.2.2c The vulnerability analysis documentation shall describe the disposition of identified vulnerabilities.
AVA_VLA.2.3c The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.
AVA_VLA.2.4c The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.
AVA_VLA.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VLA.2.2e The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.
AVA_VLA.2.3e The evaluator shall perform an independent vulnerability analysis.

AVA_VLA.2.4e The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of additional identified vulnerabilities in the intended environment.

AVA_VLA.2.5e The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a low attack potential.

6. TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

6.1. TOE Security Functions

6.1.1. Security audit

The TOE client can generate audit events related to authorized use of controlled removable devices, access to data on controlled devices (e.g., copy files), and any attempted accesses denied by the TOE. The TOE client can be configured to either ignore (i.e., not record), record, or report-events-immediately (also resulting in a recorded event) based on their type. The TOE client stores recorded events in an internally implemented (i.e., within the client) database and forwards them to the associated TOE server at a configured interval, or immediately for events so configured. Each recorded event identifies the user associated with the applicable workstation process[7], the event type, the time, success or failure of the event, and other information relevant to the specific event (e.g., the name of the file copied). When the events are delivered to the server, the server also identifies the applicable client based on its unique identity. Note that each client is assigned a unique identifier after it is installed and first contacts the server. After that, the client will identify itself to the server during internal TOE communications.

Note that the audit function cannot be disabled at the server and therefore audit events related to start-up and shutdown of the audit function are not relevant.

The TOE server stores received events in an associated database (SQL or MSDE) where the database is expected to protect the audit data. TOE administrators using server functions can review, including searching audit records based on all content available in those records, and archive[8] the audit events. Alternately, users with access to the data in the database could use their own tools to examine the audit data, but that is outside the scope of TOE and this evaluation.

The Security audit function is designed to satisfy the following security functional requirements:

FAU_GEN.1: The TOE generates audit events as indicated above.

FAU_SAR.1: The TOE provides an interface to enable an administrator to review all of the audit data collected by the TOE server.

FAU_SAR.3: The TOE audit review interface allows the administrator to formulate queries where the entire contents of audit records can be searched.

FAU_SEL.1: The TOE allows audit events to be pre-selected based on event type by being configured to be ignored or recorded as described above.

6.1.2. Cryptographic support

The TOE includes a FIPS-certified cryptographic module embedded within its server and client components. The server component is primarily responsible for managing keys that will be used by the clients to encrypt and sign data. This allows recovery of data when a client failure occurs and also allows that the client doesn't have to store unprotected key materials where they might be accessible by unauthorized users.

[7] Note that while the TOE controls processes within a workstation, the specific process identifier isn't particularly meaningful in terms of accountability so the TOE records the user associated with the process by the host operating system.

[8] Archiving audit events results in the TOE copying the archived audit records to files stored outside the TOE, and then deleting them from the database. Archived audit events cannot be reviewed within the TOE.

When new removable media device or removable media type (writeable CD or DVD) is to be encrypted, the client generates a 256-bit AES key known as the Drive Encryption Key (DEK). That key is sent to the associated server along with the identity of the user (queried from the hosting operating system) that caused the key to be sent. If the user doesn't already have one, a 256-bit AES Key Encryption Key (KEK) will be generated for that user and stored in the server database. The DEK is then encrypted with the KEK and sent back to the client where the encrypted DEK will be stored on the new removable media device or removable media type.

If a removable media device or removable media is also to be accessible off-line, the applicable user will also be prompted to provide a password. The password is used to derive a transient 128-bit AES KEK, locally on the client, which is used to encrypt the DEK and that encrypted value is also stored on the removable media device or removable media. Note that this does not require server involvement.

When the removable media device or removable media becomes available for access, the client retrieves the encrypted DEK from the device media and provides it along with the user identity to the server. The server looks up the user's KEK and decrypts the DEK. The decrypted DEK is returned to the client for use during read and write operations. Of course, the KEK must be found and the DEK must be encrypted properly or access will not be allowed. AES is used in CBC mode for media encryption/decryption.

Note that writeable CDs and DVDs are handled a little differently than removable media devices. In the case of CDs and DVDs, the TOE modifies the CD and DVD authoring capabilities provided within the host operating system (i.e., the built in operating system function to write CD and DVD contents). As such, when a CD or DVD is authored using those built-in functions, the TOE intervenes in the authoring process and places all of the file objects being written to the CD or DVD in an encrypted container. The TOE blocks other CD or DVD authoring tools if removable media encryption is mandated. Once the contents of a CD or DVD are encrypted, decryption during read operations is handled basically the same as for other removable media devices.

If the client can't contact the server and the media has been configured for off-line access, the user will be prompted for a password to derive the (decrypted) DEK for the applicable cryptographic media I/O operations.

If the user forgets the off-line password and key recovery is enabled, the user generates a challenge string that encodes the encrypted DEK; based on this challenge, an authorized key recovery officer can generate a response (the challenge/response authenticator) that includes the decrypted DEK, and provide it to the user (using off-line means outside the scope of the TOE) for accessing the device. The TOE supports two challenge/response protocols: responses may be generated either by the TOE ME server or by a trusted Check Point WebRH Server with ME WebRH Extension in the IT environment.

Note that a program can be written onto the removable media device or removable media that can be invoked to facilitate off-line access to the encrypted data provided the user knows the applicable password or can present a valid challenge/response authenticator. However, since this application can be potentially removed from or modified on the media and alternate programs could be used to perform the same function, this application is not considered part of the TSF and no specific claims are made about that application, though it is part of the product and hence the TOE.

When keys are no longer necessary, they are immediately zeroized by the TOE as certified according to FIPS 140-2.

The Cryptographic support function is designed to satisfy the following security functional requirements:

FCS_CKM.1: The TOE creates keys using its FIPS certified cryptographic module.

FCS_CKM.4: The TOE destroys keys using its FIPS certified cryptographic module.

FCS_COP.1: The TOE provides its own FIPS-evaluated cryptographic engine which performs symmetric encryption and decryption of removable media data.

6.1.3. User data protection

The TOE implements a discretionary access control policy called the Device Access Control (DAC) Security Functional Policy (SFP) in the ST. The active entities that are subjects of the DAC policy are processes that run in

the context of the TOE client host operating systems and attempt to access protected objects. The objects controlled by the TOE include:

- removable I/O devices and removable media (floppy disks, CDs, and DVDs) some of which can store data, and
- files on the host workstation.

The TOE serves to control a number of operations including:

- read and write access to removable I/O devices and removable media,
- access (read, write, create) to removable I/O device and removable media contents, and
- creation and modification of workstation files.

The DAC policies are configured by an administrator using the TOE server. The policies are sent to TOE clients to be enforced. The TOE client drivers, which are installed as filter drivers within the host operating system kernel, enforce the policies by intercepting applicable activity within the operating system kernel. By doing so, the client is aware of new removable I/O devices and removable media (e.g., when a removable device is attached or when a DVD is inserted), attempts to access removable I/O devices or removable media, and attempts to access (open, create) objects (e.g., files, registry keys) on the devices or media as well as anywhere on the host workstation. Also, the TOE, serving as filter drivers, can restrict activity or modify that activity (e.g., to encrypt and decrypt data) as it occurs.

For any given removable I/O device or removable media or device type, the TOE can restrict access altogether, limit access to read-only, or require that the contents of the device or media are encrypted.

When a removable media device or removable media type (CD or DVD) is configured to require encryption, the TOE encrypts the media as described for the Cryptographic support security function. A client can gain access to the media contents in one of three ways:

- If the client is online, it requests the decrypted DEK from the server. The server relies on the operating system to provide the requesting user's authenticated identity, and will release the DEK if the user is either the media owner, or if the owner is associated with a user group that is included in an access authorization in the user's profile settings. In its response, the server may deny the request, allow full access, or indicate to the client to restrict access to read-only.
- If the client is offline and the administrator has allowed off-line access, full access and/or read-only password-encrypted DEKs are placed on the device or media that can be decrypted based on a user provided password. Storing two such DEKs allows differentiation between a full access and read-only password. Note that the TOE can be configured to enforce a minimum password size and composition complexity to ensure that the protection is adequate. In addition, a site identifier stamped within the encrypted file system is checked against a list of authorized site identifiers.
- If the user has forgotten the off-line password, the client can generate and display a challenge string that encodes the encrypted DEK; this string is provided to an authorized key recovery officer that connects to the server, enters the challenge, and receives a response string. The response string is communicated back to the user (by offline means), allowing the client to gain access to the DEK and decrypt the media. Challenge/response authenticators provide full access to the media (i.e. cannot be restricted to read-only).

Regardless, when the device is within the scope of control of the TOE, the TOE will obtain the applicable DEK (as described in the previous section) and will encrypt and decrypt data going to and coming from the device or media so that it is essentially transparent to the user's affected process. In other words, the user sees only clear-text data.

For a given client, the TOE can be configured to digitally sign removable media devices and floppy disks with a signature[9] of media contents that is written onto the device. When digital signatures are required, the TOE can be further configured to disallow access to media failing the signature check, to scan the media for executable content and optionally allow the user to delete offending content prior to allowing access, to invoke a third-party program in the IT environment (such as a virus scanner) that has to complete before allowing access, or to allow the user to explicitly allow the device. The signature on authorized media is updated by the TOE when the media contents are modified. A signature check failure would occur when the media is modified outside the scope of the TOE and then returned to the scope of the TOE.

In addition, to controlling access to removable I/O devices and removable media, the DAC policy also serves to control the ability to create or modify workstation files based on their type. By default, protected file types include types that represent executable code. Just like all the other DAC functions, the TOE accomplishes this by intercepting file creating and open (for modification) attempts within the kernel. The TOE examines the name of the file to determine whether its 'type' is allowed in accordance with the configured list. If it is not, then the attempt is blocked and fails unless the responsible process is running a program that is specifically identified as being exempt from this restriction. Programs, identified by application name, that are presumably trusted to perform creation and modification operations (e.g., will not tamper with the TOE), can be identified in the TOE configuration and when an offending access attempt occurs, the TOE can identify the program that is currently running in the context of the applicable process in order to determine whether an exception should be made.

The User data protection function is designed to satisfy the following security functional requirements:

FDP_ACC.1, FDP_ACF.1: The TOE provides the ability to restrict access to removable I/O devices and removable media and their contents, restrict the ability to create or modify specific file types in general, and also to encrypt content of removable media devices and removable media and sign the content of removable media devices for continued protection even when the removable media leaves the scope of control of the TOE.

FMT_MSA.3: By default, access to new removable I/O devices is restricted according to the Device Manager Rules. Programs are by default restricted from creating or modifying protected file types. The administrator can override these default values for individual devices or programs. Removable media devices newly introduced to the TOE do not contain a valid signature, until authorized by the user.

6.1.4. Identification and authentication

When a TOE client connects to the server for the first time, it is assigned a unique identifier that is used for all subsequent connections to identify the client. The server database contains an entry for each client that stores the client's host name, unique identifier, and may be associated by the authorized administrator with a computer group.

In addition, the client uses the operating system's SSPI facility to establish the connection to the server. If the client and server are members of the same Windows domain, the server will be able to extract the user's authenticated identity from the connection. Users may also be authenticated if they are logged in to a Novell directory. User entries are stored in the server database including the user name and account identifier, and may be associated with one or more user groups. User/group associations may optionally be synchronized with corresponding associations in Microsoft Active Directory or Novell Directory Services.

The computer host name, user identifier and user name are recorded in audit event records generated by the client.

The server generates a profile object containing security policy settings for the client, based on the client and user identity and group associations. Up to four different profile objects can be downloaded to the client: a client computer profile, a user profile (combining user and computer profile settings), and if configured by an authorized administrator, two offline profiles: offline user and offline admin. The client selects the appropriate profile as follows: if the client is online, it applies the user profile, or when the user profile is unavailable, the computer profile. When the client is off-line, depending on computer profile settings, the client uses either a cached version of

[9] Note that while the developer documentation refers to signatures and authentication, the applicable media integrity is verified using a proprietary hash that represents the media contents. This ST refers to the hash alternately as a signature to be consistent with the developer documentation.

the user profile, the appropriate off-line profile (depending on whether the user has local administrative privileges on the client computer), or a cached version of the computer profile.

The Identification and authentication function is designed to satisfy the following security functional requirements:

FIA_ATD.1: The TOE maintains user and computer security attributes in the server database.

FIA_UID.2: The TOE server will always identify the client computer, and will identify the human user if the IT environment has authenticated the user.

FIA_USB.1: The TOE binds the computer name, user name and identity, and appropriate subject profile with subjects on the client computer acting on the behalf of that user.

6.1.5. Security management

The Endpoint Security Media Encryption Server Administration Console provides a graphical user interface to manage the configuration of the TOE as well as to read audit records. The Administration Console is implemented as a MMC snap-in, and can be installed either locally or remotely to the server. The TOE uses Microsoft's SSPI (in the IT environment) to authenticate and encrypt the communications between the Administration Console and the server. The TOE relies on the server operating system to appropriately restrict access to TOE security functions, TOE security attributes, and TSF data. Administrators are required to follow TOE guidance to ensure that the TOE is installed and configured in its evaluated configuration.

The server extracts the user's identity, authenticated by the IT environment, and associates the user with management roles and permissions in accordance with security settings in the server database, corresponding to the user's group assignments in the IT environment. Administration permissions are grouped into two basic roles: the 'Special permissions' basic permission, mapped in this ST to the key recovery officer role, allows EPM Key Recovery; the 'Administrate' basic permission corresponds to the administrator role defined in this ST, allowing all other management functions. Users with no permissions cannot invoke the Administration Console, and are considered to be in the authorized user role if they have authorized access to a TOE client computer.

The Administration Console provides dialogs to manage all aspects of the device access control policy, including removable device access settings, encryption settings, and program security guard settings, management of user and groups settings, EPM Key Recovery, as well as to manage the audit function and other aspects of the TOE configuration. The TOE also provides the ability to review the audit records and to generate audit reports, though since they are stored in a database (SQL or MSDE) the user could alternately use their own tools to review the audit data.

The user role is restricted to TOE client administration interfaces. Depending on subject profile settings, the user can authorize removable media devices, import (encrypt) and export (decrypt) removable media devices and removable media, and set media offline mode passwords. The profile may also authorize the user to enable and disable Program Security Guard, Device Manager, or Removable Media Manager functionality.

Note that client access to removable devices is restrictive in the sense that the configuration is always enforced and is explicitly defined upon installation of the client on the host workstation. Only administrators working through the server can change the configuration of the TOE.

The Security management function is designed to satisfy the following security functional requirements:

FMT_SMF.1: The TOE provides interfaces that users can use to manage all aspects of the configuration of the TOE.

FMT_SMR.1: The TOE associates user with roles based on the user's identity as authenticated by the IT environment, in accordance with security settings in the server database.

FMT_MOF.1, FMT_MSA.1b, FMT_MSA.1c, FMT_MTD.1: The TOE's Administration Console restricts management of security behaviour and TSF data to the administrator role. Only key recovery officers can generate challenge/response authenticators for EPM Key Recovery.

FMT_MSA.1a: The TOE restricts user role management functions to users with appropriate profile authorizations.

FMT_MSA.3: The administrator can override default values for security attributes used to enforce the device access control policy. Users with appropriate profile settings can authorize removable media devices.

6.1.6. Protection of the TSF

The TOE protects itself from potential tampering and bypass attempts by working in the context of its host operating systems to actively control all attempts to access removable I/O devices and their contents, block attempts to create or modify controlled file types anywhere on the host workstation, or to block access to files and registry keys that belong to the TOE (note that this later function is not part of user data protection, but is implemented in the same manner via a filter driver designed to limit attempts to access TOE files and registry keys). When access is attempted through the TOE, the TOE will ensure that its rules are enforced prior to allowing the access attempt to proceed.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

FPT_RVM.1a: The TOE is designed to ensure its rules are applied when control passes through the TOE and the TOE works with the environment to ensure that control does pass through the TOE when it should.

FPT_SEP.1a: The TOE is designed to work with the host operating system to actively identify and block attempts to access the TOE and its data.

6.2. TOE Security Assurance Measures

6.2.1. Configuration management

The configuration management measures applied by Check Point ensure that configuration items are uniquely identified, and that documented procedures are used to control and track changes that are made to the TOE. Check Point ensures changes to the implementation representation are controlled with the support of automated tools and that TOE associated configuration item modifications are properly controlled. Check Point performs configuration management on the TOE implementation representation, design documentation, tests and test documentation, user and administrator guidance, delivery and operation documentation, life-cycle documentation, vulnerability analysis documentation, configuration management documentation, and security flaws.

These activities are documented in:

Check Point EndPoint Security Configuration Management Plan

The Configuration management assurance measure satisfies the following assurance requirements:

ACM_AUT.1

ACM_CAP.4

ACM_SCP.2

6.2.2. Delivery and operation

Check Point provides delivery documentation and procedures to identify the TOE, allow detection of unauthorized modifications of the TOE and installation and generation instructions at start-up. Check Point's delivery procedures describe all applicable procedures to be used to detect modification to the TOE. Check Point also provides documentation that describes the steps necessary to install the TOE in accordance with the evaluated configuration.

These activities are documented in:

Endpoint Security Media Encryption CC Evaluated Configuration Administrator's Guide

Check Point Endpoint Security Media Encryption Installation Guide

Check Point Endpoint Security Client Installation Guide

The Delivery and operation assurance measure satisfies the following assurance requirements:

ADO_DEL.2

ADO_IGS.1

6.2.3. Development

Check Point has numerous documents describing all facets of the design of the TOE. In particular, they have a functional specification that describes the accessible TOE interfaces; a high-level design that decomposes the TOE architecture into subsystems and describes each subsystem and their interfaces; a low-level design that further decomposes the TOE architecture into modules and describes each module and their interfaces; and, correspondence documentation that explains how each of the design abstractions correspond from the TOE summary specification in the Security Target to the actual implementation of the TOE. Furthermore, Check Point has a security model that describes each of the security policies implemented by the TOE. Of course, the implementation of the TOE itself is also available as necessary.

These activities are documented in:

Check Point Endpoint Security Media Encryption Functional Specification

Check Point Endpoint Security Media Encryption High-Level Design

Check Point Endpoint Security Media Encryption Low-Level Design

Check Point Endpoint Security Media Encryption Security Policy Model

Check Point Endpoint Security Media Encryption Analysis of Correspondence

The Development assurance measure satisfies the following assurance requirements:

ADV_FSP.2

ADV_HLD.2

ADV_IMP.1

ADV_LLD.1

ADV_RCR.1

ADV_SPM.1

6.2.4. Guidance documents

Check Point provides administrator and user guidance on how to utilize the TOE security functions and warnings to administrators and users about actions that can compromise the security of the TOE.

These activities are documented in:

Endpoint Security Media Encryption CC Evaluated Configuration Administrator's Guide

Check Point Endpoint Security Media Encryption Administration Guide

Endpoint Security Media Encryption User's Guide

The Guidance documents assurance measure satisfies the following assurance requirements:

AGD_ADM.1

AGD_USR.1

6.2.5. Life cycle support

Check Point ensures the adequacy of the procedures used during the development and maintenance of the TOE through the use of a comprehensive life-cycle management plan. Check Point applies security controls on the development environment that are adequate to provide the confidentiality and integrity of the TOE design and implementation that is necessary to ensure the secure development of the TOE. Check Point has a documented model of the TOE life cycle that ensures that the TOE is developed and maintained in a well-defined manner. Check Point uses well-defined development tools in order to ensure consistent and predictable results while developing the TOE.

Flaw tracking and remediation procedures and guidance addressed to TOE developers describe the procedures used to accept, track, and act upon reported security flaws and requests for corrections to those flaws, as well as the distribution of reports and corrections to registered users. Guidance addressed to TOE users describes means by which TOE users with a valid Software Subscription license report to the developer any suspected security flaws in the TOE, and receive security flaw reports and corrections.

These activities are documented in:

Check Point Endpoint Security Media Encryption Life-Cycle

The Life cycle support assurance measure satisfies the following assurance requirements:

ALC_DVS.1

ALC_FLR.3

ALC_LCD.1

ALC_TAT.1

6.2.6. Tests

Check Point has a test plan that describes how each of the necessary security functions is tested, along with the expected test results. Check Point has documented each test as well as an analysis of test coverage and depth demonstrating that the security aspects of the design evident from the functional specification and high-level design are appropriately tested. Actual test results are created on a regular basis to demonstrate that the tests have been applied and that the TOE operates as designed.

These activities are documented in:

Check Point Endpoint Security Media Encryption Test Plan

Check Point Endpoint Security Media Encryption Test Results

The Tests assurance measure satisfies the following assurance requirements:

ATE_COV.2

ATE_DPT.1

ATE_FUN.1

ATE_IND.2

6.2.7. Vulnerability assessment

The TOE administrator and user guidance documents describe the operation of the TOE and how to maintain a secure state. These guides also describe all necessary operating assumptions and security requirements outside the scope of control of the TOE. They have been developed to serve as complete, clear, consistent, and reasonable administrator and user references. Furthermore, Check Point has conducted a misuse analysis demonstrating that the provided guidance is complete.

Check Point has conducted a SOF analysis wherein all permutational or probabilistic security mechanisms have been identified and analyzed resulting in a demonstration that all of the relevant mechanisms (i.e., the use of passwords to protect encrypted media) fulfill the minimum SOF claim: SOF-basic.

Check Point performs regular vulnerability analyses of the entire TOE (including documentation) to identify weaknesses that can be exploited in the TOE.

These activities are documented in:

Check Point Endpoint Security Media Encryption Analysis of Guidance Documentation

Check Point Endpoint Security Media Encryption SOF Analysis

Check Point Endpoint Security Media Encryption Vulnerability Analysis

The Vulnerability assessment assurance measure satisfies the following assurance requirements:

AVA_MSU.2

AVA_SOF.1

AVA_VLA.2

6.3. Identification of Standards

The subject of criteria for the assessment of the inherent qualities of cryptographic algorithms is not covered in the CC. The SFRs in the Cryptographic Support (FCS) class stated in Section 5.1.2 therefore reference external standards that the implementation must meet when providing the required capabilities.

The following table summarizes the standards compliance claims made in Section 5.1.2 and states for each the method used to determine compliance (aside from development assurances). The method may be an applicable NIST certificate number, other third-party certification, or a vendor assertion.

Standard claimed	Cryptographic SFRs	Cryptographic operation	Method of determining compliance
X9.31-based PRNG	FCS_CKM.1	User data protection key generation	RNG Cert. #250
FIPS 140-2 level 1	FCS_CKM.1,	User data protection key	FIPS 140-2 cert. #784[10]

[10] FIPS 140-2 certificate #784, the Reflex Magnetics Cryptographic Library v.1.0, was validated on the Windows XP Professional operating system. However, note that FIPS 140-2 Implementation Guidance G.5 allows vendor porting and re-compilation of a validated software cryptographic module to a compatible operating system that was not included as part of the validation testing, when this does not require source code modifications, as is the case for

Standard claimed	Cryptographic SFRs	Cryptographic operation	Method of determining compliance
	FCS_CKM.4	generation and destruction	
FIPS 197 (AES) in CBC mode	FCS_COP.1	Encryption and decryption of stored user data	AES cert. #466

Table 4 - Cryptographic Standards and Method of Determining Compliance.

7. Protection Profile Claims

There is no Protection Profile claim in this Security Target.

the supported operating systems identified in section 2.1. The validation status is maintained in this case without re-testing.

8. Rationale

This section provides the rationale for completeness and consistency of the Security Target. The rationale addresses the following areas:

Security Objectives;

Security Functional Requirements;

Security Assurance Requirements;

Strength of Functions;

Requirement Dependencies;

TOE Summary Specification; and,

PP Claims.

8.1. Security Objectives Rationale

This section shows that all secure usage assumptions and threats are completely covered by security objectives. In addition, each objective counters or addresses at least one assumption or threat.

8.1.1. Security Objectives Rationale for the TOE and Environment

This section provides evidence demonstrating the coverage of threats and usage assumptions by the security objectives.

	T.ACCOUNTABILITY	T.ADMIN_TOOLS	T.MASQUERADE	T.TSF_COMPROMISE	T.UNAUTH_ACCESS	A.LOCATE	A.NO_EVIL
O.AUDIT_GENERATION	X						
O.AUDIT_REVIEW	X						
O.DEV_ACCESS					X		
O.DEV_INTEG					X		
O.ENCRYPTION					X		
O.MANAGE		X			X		
O.ADMIN_ROLE			X				
O.TOE_PROTECTION				X			
OE.TIME_STAMPS	X						
OE.TOE_ACCESS			X	X			
OE.TOE_COMMS				X			
OE.TOE_PROTECTION	X			X			
OE.CONFIG							X
OE.PHYCAL						X	

Table 5 - Environment to Objective Correspondence

8.1.1.1. T.ACCOUNTABILITY

A user may not be held accountable for their actions.

This Threat is satisfied by ensuring that:

- O.AUDIT_GENERATION: The TOE will include the capability to audit security-relevant user actions.
- O.AUDIT_REVIEW: The TOE will provide the means of reviewing the log of audit actions.
- O.TOE_PROTECTION: The TOE will protect the TOE client and its assets from external interference, tampering, or bypass attempts, thus ensuring the generation of auditable events.
- OE.TOE_PROTECTION: The IT environment will protect the TOE and its assets from external interference, tampering, or bypass, including protection of audit data generated by the TOE.
- OE.TIME_STAMPS: The IT environment will provide reliable time information to the TOE so that it can ensure that audit records are appropriately time stamped.

8.1.1.2. T.ADMIN_TOOLS

An authorized administrator may not have tools suitable to allow the effective management of the TOE security functions.

This Threat is satisfied by ensuring that:

- O.MANAGE: The TOE will provide administrators with the tool necessary to suitably manage the security functions of the TOE.

8.1.1.3. T.MASQUERADE

An unauthorized user, process, or external IT entity may masquerade as an authorized entity to gain access to data or TOE resources.

This Threat is satisfied by ensuring that:

- O.ADMIN_ROLE: The TOE will restrict access to TOE security management functions to those users authorized to manage the security functions of the TOE.
- OE.TOE_ACCESS: The IT environment will ensure that only authorized users gain access to the security functions and resources controlled by the TOE.

8.1.1.4. T.TSF_COMPROMISE

A malicious user may cause the TOE or its configuration data to be inappropriately accessed (viewed, modified or deleted).

This Threat is satisfied by ensuring that:

- O.TOE_PROTECTION: The TOE will protect itself to the extent it can, meaning via its own functions, from any attempts to bypass, tamper with or interfere with its security functions.
- OE.TOE_ACCESS: The IT environment will ensure that only authorized users gain access to the security functions and resources controlled by the TOE.
- OE.TOE_COMMS: The IT environment will provide a means of protecting communications between distributed parts of the TOE.
- OE.TOE_PROTECTION: The IT environment will protect itself and the TOE from any attempts to bypass, temper with or interfere with security functions of the IT environment or those of the TOE, to the extent that the IT environment can protect the TOE (i.e., via interfaces not controlled by the TOE).

8.1.1.5. T.UNAUTH_ACCESS

A user may gain unauthorized access (view, modify, delete) to removable I/O devices or data stored within removable media devices.

This Threat is satisfied by ensuring that:

- O.DEV_ACCESS: The TOE will provide mechanisms that can effectively restrict and control access to removable I/O devices.
- O.DEV_INTEG: The TOE will provide mechanisms that can determine whether the contents of a removable media storage device may have changed while outside the scope of control of the TOE.
- O.ENCRYPTION: The TOE will provide mechanisms that can encrypt removable media device contents so that they remain protected even when outside the scope of control of the TOE.
- O.MANAGE: The TOE will provide administrators with the tool necessary to suitably manage the security functions of the TOE.

8.1.1.6. A.LOCATE

The TOE server component will be located within controlled access facilities, which will prevent unauthorized physical access.

This Assumption is satisfied by ensuring that:

- OE.PHYCAL: The TOE will be instantiated in IT environments that are suitably protected from potential physical attacks.

8.1.1.7. A.NO_EVIL

The TOE will be installed, configured, managed and maintained in accordance with its guidance documentation.

This Assumption is satisfied by ensuring that:

- OE.CONFIG: The TOE will be installed and configured within its IT environment so that it is properly instantiated so that it can effectively enforce/apply its security policies.

8.2. Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that Table 6 indicates the requirements that effectively satisfy the individual objectives.

8.2.1. Security Functional Requirements Rationale

All Security Functional Requirements (SFR) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

	O.AUDIT_GENERATION	O.AUDIT_REVIEW	O.DEV_ACCESS	O.DEV_INTEG	O.ENCRYPTION	O.MANAGE	O.ADMIN_ROLE	O.TOE_PROTECTION	OE.TIME_STAMPS	OE.TOE_ACCESS	OE.TOE_COMMS	OE.TOE_PROTECTION
FAU_GEN.1	X											
FAU_SAR.1		X										
FAU_SAR.3		X										
FAU_SEL.1	X											

	O.AUDIT_GENERATION	O.AUDIT_REVIEW	O.DEV_ACCESS	O.DEV_INTEG	O.ENCRYPTION	O.MANAGE	O.ADMIN_ROLE	O.TOE_PROTECTION	OE.TIME_STAMPS	OE.TOE_ACCESS	OE.TOE_COMMS	OE.TOE_PROTECTION
FCS_CKM.1					X							
FCS_CKM.4					X							
FCS_COP.1					X							
FDP_ACC.1			X	X				X				
FDP_ACF.1			X	X				X				
FIA_ATD.1						X						
FIA_UID.2						X						
FIA_USB.1						X						
FMT_MOF.1							X					
FMT_MSA.1a							X					
FMT_MSA.1b							X					
FMT_MSA.1c							X					
FMT_MSA.3			X	X		X		X				
FMT_MTD.1							X					
FMT_SMF.1						X						
FMT_SMR.1						X						
FPT_RVM.1a								X				
FPT_SEP.1a								X				
FAU_STG.1												X
FIA_UAU.1										X		
FPT_ITT.1											X	
FPT_RVM.1b												X
FPT_SEP.1b												X
FPT_STM.1									X			

Table 6 - Objective to Requirement Correspondence

8.2.1.1. O.AUDIT_GENERATION

The TOE will provide the capability create records of security relevant events associated with users.

This TOE Security Objective is satisfied by ensuring that:

- FAU_GEN.1: The TOE is required to provide the means to audit security relevant actions attempted by users.
- FAU_SEL.1: The TOE is required to allow administrators to select the auditable events they wish to be audited, allowing the administrators to focus on those events they consider most important.

8.2.1.2. O.AUDIT_REVIEW

The TOE will provide the capability to view audit information.

This TOE Security Objective is satisfied by ensuring that:

- FAU_SAR.1: The TOE is required to provide a means whereby administrators can review generated audit records.

- FAU_SAR.3: The TOE is required to provide the ability to search audit records so that they can be more effectively reviewed.

8.2.1.3. O.DEV_ACCESS

The TOE will provide the means to control access to removable I/O devices.

This TOE Security Objective is satisfied by ensuring that:

- FDP_ACC.1, FDP_ACF.1: The TOE is required to enforce an access control policy limiting the ability to use removable I/IO devices based on the rules established by the TOE administrator.
- FMT_MSA.3: The TOE is required to implement well-defined default controls for removable I/O device access that can be overridden by administrators.

8.2.1.4. O.DEV_INTEG

The TOE will provide the means to measure the integrity of removable media contents and to limit access to the contents when the integrity has been violated.

This TOE Security Objective is satisfied by ensuring that:

- FDP_ACC.1, FDP_ACF.1: The TOE is required to enforce an access control policy where the TOE can determine when the contents of a removable media device have been modified outside the scope of control of the TOE.
- FMT_MSA.3: The TOE is required to implement restrictive default values for removable media device digital signatures that can be overridden by administrators.

8.2.1.5. O.ENCRYPTION

The TOE will provide the means to encrypt the content of removable media.

This TOE Security Objective is satisfied by ensuring that:

- FCS_CKM.1: The TOE is required to generate suitable keys for its cryptographic operations.
- FCS_CKM.4: The TOE is required to destroy cryptographic keys in an suitable manner when necessary.
- FCS_COP.1: The TOE is required to encrypt the contents of removable media devices when so configured.

8.2.1.6. O.MANAGE

The TOE will allow administrators to effectively manage the TOE and its security functions.

This TOE Security Objective is satisfied by ensuring that:

- FMT_SMF.1: The TOE is required to provide suitable means to administer the security functions to the administrator.
- FMT_SMR.1: The TOE associates users with the administrator role in the context of the use of the Endpoint Security Media Encryption Server Administration Console.
- FIA_UID.2, FIA_ATD.1, and FIA_USB.1: The TOE is required to support per-user management of security attributes, by ensuring that users are identified before they can perform any security-relevant functions, maintaining security attributes for users, and binding subjects acting on behalf of the user with appropriate security profile settings.

8.2.1.7. O.ADMIN_ROLE

The TOE will provide authorized administrator roles to isolate and control administrative actions.

This TOE Security Objective is satisfied by ensuring that:

- FMT_MOF.1: The TOE is required to restrict the ability to modify the behavior of TOE security functions to authorized administrators.
- FMT_MSA.1a, FMT_MSA.1b, and FMT_MSA.1c: The TOE is required to restrict the access to TOE security attributes to authorized administration roles.

- FMT_MTD.1: The TOE is required to restrict the access to security-relevant TOE data to authorized administrators.

8.2.1.8. O.TOE_PROTECTION

The TOE will protect the TOE client and its assets from external interference, tampering, or bypass attempts.

This TOE Security Objective is satisfied by ensuring that:

- FPT_RVM.1a: The TOE is required to ensure that its security functions cannot be bypassed so that its security functions are always enforced.
- FPT_SEP.1a: The TOE is required to protect itself from potential tamper attempts and also to distinguish its subjects so that it can enforce its policies on each subject appropriately.
- FDP_ACC.1, FDP_ACF.1: The TOE is required to enforce an access control policy limiting the ability to create or modify workstation files according to file type, thus protecting TOE client executable code from unauthorized modification.
- FMT_MSA.3: The TOE is required to implement restrictive values for program access to restricted workstation file types that can be overridden by administrators.

8.2.1.9. OE.TIME_STAMPS

The IT Environment will provide a reliable source of timestamps to the TOE.

This IT Environment Security Objective is satisfied by ensuring that:

- FPT_STM.1: The IT environment is required to provide reliable time stamps so that the TOE can include them in audit records.

8.2.1.10. OE.TOE_ACCESS

The IT Environment will ensure that users gain only authorized access to the TOE and to the resources that the TOE controls.

This IT Environment Security Objective is satisfied by ensuring that:

- FIA_UAU.1: The IT environment is required to always ensure that users are appropriately authenticated before they can perform any security-relevant functions.

8.2.1.11. OE.TOE_COMMS

The IT Environment will provide a means of protecting communication between distributed TOE components.

This IT Environment Security Objective is satisfied by ensuring that:

- FPT_ITT.1: The IT environment is required to protect data when communicated among distributed parts of the TOE.

8.2.1.12. OE.TOE_PROTECTION

The IT Environment will protect the TOE and its assets from external interference, tampering, or bypass attempts.

This IT Environment Security Objective is satisfied by ensuring that:

- FAU_STG.1: The IT environment is required to protect stored audit records generated by the TOE to ensure they are not inappropriately modified or deleted.
- FPT_RVM.1b: The IT environment is required to ensure that the security functions of itself and the TOE cannot be bypassed so that the security functions are always enforced.
- FPT_SEP.1b: The IT environment is required to protect itself and the TOE from potential tamper or interference attempts and also to distinguish its subjects so that it can enforce its policies on each subject appropriately.

8.3. Security Assurance Requirements Rationale

EAL 4 was selected as the base assurance level because the TOE is a commercial product whose users require a moderate to high level of independently assured security. The TOE is targeted at a relatively benign server environment with good physical access security and competent administrators and also at a client environment where the client users are willing subjects who benefit from the security offered by the TOE and would not actively attempt to circumvent or hamper the security functions provided by the TOE. Within such environments it is assumed that attackers will have little attack potential or motivation. As such, EAL 4 is appropriate to provide the assurance necessary to counter the limited potential /motivation for attack.

In addition, the assurance requirements have been augmented with ALC_FLR.3 (Systematic flaw remediation) to provide assurance that the TOE will be maintained and supported in the future, requiring the TOE developer to track and correct flaws in the TOE, and providing guidance to TOE users for how to submit security flaw reports to the developer, and how to register themselves with the developer so that they may receive these corrective fixes.

8.4. Strength of Function Rationale

The overall strength of function claim of SOF-basic is commensurate with the base assurance claim of EAL4.

The security functions for which a strength of function claim is appropriate are Cryptographic support, in relation to the use of passwords for deriving removable media encryption keys, and User data protection, for the generation of removable media device digital signatures.

The user selection of passwords is constrained by the TOE where the administrator is instructed by TOE guidance to configure minimum password length requirements. This supports a SOF-basic strength of function that meets the strength of function requirement for FCS_COP.1. The administrator can further define password composition rules.

The digital signature for removable media device contents is a 64 bit hash based on the volume bit map and on file time stamps, seeded with a per-site unique Media ID. The hash supports a SOF-basic strength of function that meets the strength of function requirement for FDP_ACF.1.

The analysis of the strength of these mechanisms is presented in the Check Point vulnerability analysis.

8.5. Requirement Dependency Rationale

The following table demonstrates that all dependencies among the claimed security requirements are satisfied, except FMT_MSA.2, and therefore the requirements work together to accomplish the overall objectives defined for the TOE. FMT_MSA.2 requires that only secure values are accepted in relation to the cryptographic security functional requirements included in this ST. However, the cryptographic mechanisms have been evaluated in accordance with FIPS 140-2 (certificate #784) and as such it is assumed that any requirements for accepting only secure values would have been addressed in that evaluation.

Note: dependencies satisfied by SFRs for the IT environment are indicated in *italicized* text in the table.

Note: dependencies of SFRs for the IT environment are not completely satisfied – in accordance with NIAP PD-0091 : Dependencies of Requirements on the IT Environment.

ST Requirement	CC Dependencies	ST Dependencies
FAU_GEN.1	FPT_STM.1	*FPT_STM.1*
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1
FAU_SAR.3	FAU_SAR.1	FAU_SAR.1
FAU_SEL.1	FAU_GEN.1 and FMT_MTD.1	FAU_GEN.1 and FMT_MTD.1
FCS_CKM.1	(FCS_CKM.2 or FCS_COP.1) and FCS_CKM.4 and FMT_MSA.2	FCS_COP.1and FCS_CKM.4 and **[FMT_MSA.2 – not included]**
FCS_CKM.4	(FDP_ITC.1 or FCS_CKM.1) and FMT_MSA.2	FCS_CKM.1 and **[FMT_MSA.2 – not included]**

ST Requirement	CC Dependencies	ST Dependencies
FCS_COP.1	FDP_ITC.1 or FCS_CKM.1) and FCS_CKM.4 and FMT_MSA.2	FCS_CKM.1 and FCS_CKM.4 and **[FMT_MSA.2 – not included]**
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1 and FMT_MSA.3	FDP_ACC.1 and FMT_MSA.3
FIA_ATD.1	None	none
FIA_UID.2	none	none
FIA_USB.1	FIA_ATD.1	FIA_ATD.1
FMT_MOF.1	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.1 and FMT_SMF.1
FMT_MSA.1a, FMT_MSA.1b, FMT_MSA.1c	FMT_SMR.1 and FMT_SMF.1 and (FDP_ACC.1 or FDP_IFC.1)	FMT_SMR.1 and FMT_SMF.1 and FDP_ACC.1
FMT_MSA.3	FMT_MSA.1 and FMT_SMR.1	FMT_MSA.1c and FMT_SMR.1
FMT_MTD.1	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.1 and FMT_SMF.1
FMT_SMF.1	none	none
FMT_SMR.1	FIA_UID.1	FIA_UID.2
FPT_RVM.1a	none	none
FPT_SEP.1a	none	none
FAU_STG.1	FAU_GEN.1	FAU_GEN.1
FIA_UAU.1	FIA_UID.1	FIA_UID.2
FPT_ITT.1	none	none
FPT_RVM.1b	none	none
FPT_SEP.1b	none	none
FPT_STM.1	none	none
ACM_AUT.1	ACM_CAP.3	ACM_CAP.4
ACM_CAP.4	ALC_DVS.1	ALC_DVS.1
ACM_SCP.2	ACM_CAP.3	ACM_CAP.4
ADO_DEL.2	ACM_CAP.3	ACM_CAP.4
ADO_IGS.1	AGD_ADM.1	AGD_ADM.1
ADV_FSP.2	ADV_RCR.1	ADV_RCR.1
ADV_HLD.2	ADV_FSP.1 and ADV_RCR.1	ADV_FSP.2 and ADV_RCR.1
ADV_IMP.1	ADV_LLD.1 and ADV_RCR.1 and ALC_TAT.1	ADV_LLD.1 and ADV_RCR.1 and ALC_TAT.1
ADV_LLD.1	ADV_HLD.2 and ADV_RCR.1	ADV_HLD.2 and ADV_RCR.1
ADV_RCR.1	none	none
ADV_SPM.1	ADV_FSP.1	ADV_FSP.2
AGD_ADM.1	ADV_FSP.1	ADV_FSP.2
AGD_USR.1	ADV_FSP.1	ADV_FSP.2
ALC_DVS.1	none	none
ALC_FLR.3	none	none
ALC_LCD.1	none	none
ALC_TAT.1	ADV_IMP.1	ADV_IMP.1
ATE_COV.2	ADV_FSP.1 and ATE_FUN.1	ADV_FSP.2 and ATE_FUN.1
ATE_DPT.1	ADV_HLD.1 and ATE_FUN.1	ADV_HLD.2 and ATE_FUN.1
ATE_FUN.1	none	none
ATE_IND.2	ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1 and ATE_FUN.1	ADV_FSP.2 and AGD_ADM.1 and AGD_USR.1 and ATE_FUN.1
AVA_MSU.2	ADO_IGS.1 and ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1	ADO_IGS.1 and ADV_FSP.2 and AGD_ADM.1 and AGD_USR.1
AVA_SOF.1	ADV_FSP.1 and ADV_HLD.1	ADV_FSP.2 and ADV_HLD.2
AVA_VLA.2	ADV_FSP.1 and ADV_HLD.2 and ADV_IMP.1 and ADV_LLD.1 and AGD_ADM.1 and AGD_USR.1	ADV_FSP.2 and ADV_HLD.2 and ADV_IMP.1 and ADV_LLD.1 and AGD_ADM.1 and AGD_USR.1

Table 7 – Requirement Dependency Rationale

8.6. Explicitly Stated Requirements Rationale

There are no explicitly stated requirements in this Security Target.

8.7. TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. Table 8 - Security Functions vs. Requirements Mapping demonstrates the relationship between security requirements and security functions.

	Security audit	Cryptographic support	User data protection	Identification and authentication	Security management	Protection of the TSF
FAU_GEN.1	X					
FAU_SAR.1	X					
FAU_SAR.3	X					
FAU_SEL.1	X					
FCS_CKM.1		X				
FCS_CKM.4		X				
FCS_COP.1		X				
FDP_ACC.1			X			
FDP_ACF.1			X			
FIA_ATD.1				X		
FIA_UID.2				X		
FIA_USB.1				X		
FMT_MOF.1					X	
FMT_MSA.1a, FMT_MSA.1b, FMT_MSA.1c					X	
FMT_MSA.3			X		X	
FMT_MTD.1					X	

FMT_SMF.1					X	
FPT_RVM.1a						X
FPT_SEP.1a						X

Table 8 - Security Functions vs. Requirements Mapping

8.8. PP Claims Rationale

See Section 7, Protection Profile Claims.

9. Appendix A – Supported Antivirus Solutions (pre-configured)

This section lists the third-party antivirus products that were included in the evaluation.

Vendor	Product and Version
McAfee	VirusScan 6.x-11.x
Symantec	Symantec AntiVirus Corporate Edition
Sophos	Anti-Virus 3.x-6.x
Trend Micro	Office Corporate Edition 5
Gri-Soft	AVG AntiVirus 7.x

