SecureDoc® Disk Encryption v4.3C

Security Target for Common Criteria EAL-4

Abstract: This document represents Security Target Document for Secure Doc Disk Encryption product v4.3C. It specifies treats, security objectives, functional requirements and assurance measures implemented in the TOE for evaluation for Common Criteria EAL-4

Author(s):	Rob deCarle, Alexandr Mazuruc
Reviewer(s):	Thi Nguyen-Huu
Status:	Version 1.11 / approved /
Revision Date:	May 14, 2007
Print Date:	31/07/2007 9:36 AM

Approvals:

Person approved the document	Date
Thi C. Nguyen-Huu, WinMagic Inc.	31 January 2005
Thi C. Nguyen-Huu, WinMagic Inc.	2 March 2005
Thi C. Nguyen-Huu, WinMagic Inc.	31 August 2005
Thi C. Nguyen-Huu, WinMagic Inc.	21 April 2006
Thi C. Nguyen-Huu, WinMagic Inc.	9 August 2006
Thi C. Nguyen-Huu, WinMagic Inc.	14 May 2007

Revision History Table:

Revision	Date	Changes Since Previous Revision	Revision Author
1.0 (Draft 1)	2 April 2004	This is the initial draft of the document.	Rob deCarle
1.1 (Draft 2)	18 January 2005	Internal consistence of the document revised	Alexandr Mazuruc
1.2	31 January 2005	Completeness of the document inspected.	Alexandr Mazuruc
1.3	2 March 2005	Incorporated changes made to TOE Security Policy document.	Alexandr Mazuruc
1.4	11 July 2005	Updated to correspond to SecureDoc v.4.1	Alexandr Mazuruc
1.5	30 August 2005	Physical Boundaries, Process Assurance updated	Alexandr Mazuruc
1.6	21 April 2006	Updated to correspond to SecureDoc v.4.2	Alexandr Mazuruc
1.7	9 June 2006	Section 8.2.5 updated	Alexandr Mazuruc
1.8	9 August 2006	Issues specified by DOMUS ITSL in SecureDoc ST OR-01, OR-02 addressed	Alexandr Mazuruc
1.9	19 December 2006	Cryptographic Support class FCS expanded. Requirements dependencies corrected. Issues specified in CB OR-1 by CSE addressed.	Alexandr Mazuruc
1.10	16 January 2007	Reference to Access Control SFP added to 6.1.5	Alexandr Mazuruc
1.11	14 May 2007	2.2.2 Physical Boundaries updated to v4.3 Version changed to 4.3C	Alexandr Mazuruc

Table of Contents

1. ST Introduction

This section identifies the Security Target and Target of Evaluation (TOE) identification, overview of ST, and the ST Conformance Claims.

1.1. Identification

ST Title: **SecureDoc™ Disk Encryption Security Target**

ST Version: **Version 1.11**

TOE: **SecureDoc™ Disk Encryption software product, version 4.3C**

Registration: <to be filled in by registry>

Keywords: Access control, Disk Encryption, Information Protection

1.2. Overview of Document

The WinMagic SecureDoc™ Disk Encryption software product, version 4.3C is a disk encryption product for use in a Windows 2000/XP/2003 environment running on a PC, workstation platform, or laptop. The SecureDoc™ product performs disk encryption using AES encryption algorithm on hard disks, logical segments of disks, or removable media such as floppy disks, flash disks, USB Drives, etc. The Common Criteria Evaluation Assurance Level 4 evaluation documented herein describes assumptions, threats, security objectives that pertain to the product in its normal use and presents findings that establish its functional security properties at that level.

1.3. Conformance Claim

The TOE conforms to the following specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, Version 2.2, January 2004, ISO/IEC 15408-02
    - Part 2 Conformant
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, Version 2.2, January 2004, ISO/IEC 154408-3
    - Part 3 Conformant
    - Evaluation Assurance Level 4 (EAL4)

2. TOE Description

SecureDoc™ Disk Encryption v.4.3C is a hard disk encryption application that is compliant with FIPS 140-2 Level 2 requirements for cryptographic modules, certified by NIST/CSE. The product is a software based security product for the Windows based PC platform that employs both authentication at pre-boot stage and transparent disk encryption to provide complete protection of information resources stored on fixed media in a workstation, or laptop.

2.1 Product Description

Protection of information assets from unauthorized access or disclosure is a major concern of organizations and forms a prominent component of the security policy of most IT systems. One of the most effective solutions to the problem of implementing an effective protection system is keeping "data-at-rest" in encrypted format.

Cryptographic access of information assets in the context of an operating system's file system can be implemented at the file, directory or disk level. While the file or directory level can be useful for certain low risk situations, or for systems that form a part of a communications system, they do not adequately address the problems imposed by contemporary personal computer and workstation operating systems and their applications.

The unpredictable presence and behaviour of temporary backup files, as well as caches and buffers that may reside on or be swapped to a hard drive entails the significant risk that sensitive information may be replicated in a plaintext repository that is unknown to the originator or legitimate owner of the principal file asset. The strategy of encrypting only at a file or directory level does not address this risk. Through the strategy of disk (logical segments or physical) encrypting, however, it is possible to mitigate this risk effectively. The SecureDoc™ Disk Encryption product requires the user to encrypt the entire hard disk.

SecureDoc™ software performs full disk encryption for personal computers and workstation platforms that run on Microsoft Windows 2000, XP, 2003 operating systems. SecureDoc™ will run on any platform that successfully supports these operating systems. The disk drives or partition can be protected by this product. The operating system is considered to be part of the environment, not part of the TOE.

The SecureDoc™ product performs disk encryption and decryption dynamically at the device driver level. The product also encrypts at a higher level like file and folder encryption. Upon installation, the product can be configured to encrypt all information stored on a physical disk drive or disk partition using a specified encryption algorithm (256-bit AES). User's encryption keys are stored in a key file that can be accessed only by entering the correct password, at login, that corresponds to it.

The product is installed and operated through the use of a specialized device driver that filters the standard disk read and write processes in the OS. This allows encrypted read and write operations to be performed at a low level to those disk sectors selected for encrypted I/O and plaintext operations to be performed for all

other disk sectors. The low-level boot login process is also modified to achieve integration with the OS password authentication process.

All low level write operations to a disk that has been protected by the SecureDoc™ product are encrypted using a Data Encryption Key (DEK), unique for that disk. The DEK is encrypted by the user's encryption key and stored in the "SecureDoc Encryption Header" together with the encrypted object. All low level read operations similarly involve decryption using the same DEK. This is done transparently by the installed device driver component of the product. Disk drives and partitions not selected by the user for protection are written and read as plaintext in the normal manner. Once the initial configuration is chosen regarding disk drives, the operation is transparent and requires no additional user intervention.

The SecureDoc™ product is installed through the standard installation wizard supplied on the installation CD or online executable. Some manual configuration and administration is essential to ensure that the normal operation of SecureDoc™ is sufficient to protect against leakage of sensitive information assets.

The above data objects must be configured to reside on the disk/partition under protection by the SecureDoc™ product. If they are sited on another unprotected drive, or elsewhere in a network, the encryption services of SecureDoc™ are unable to protect them from unauthorized access. Further consultation with specific application user manuals may be necessary to extend this list for a specific environment.

The system administrator is advised that only compression software recommended by WinMagic should be used to ensure that SecureDoc™ can protect the compressed files residing on encrypted disks.

As with any software product, it is advisable to have a state of the art virus-checking program to ensure viruses are not introduced. Only software approved by the system administrator should be installed on the platform that SecureDoc™ is protecting to prevent the importation of Trojan horses or other destructive software. This is especially true if SecureDoc is used in an enterprise-wide system with Internet access.

2.2 Security Environment TOE Boundary

2.2.1 Logical Boundaries

The logical boundaries include the interfaces necessary to incorporate the Security Functions.

- **Access Control**
  The TOE controls access by an identified and authenticated user to those objects that are files on a disk drive or sector that have been encrypted by the TOE. All read and write operations to the encrypted disk or sectors are mediated by the TOE.

The TOE enforces the policy of access to a specific drive or sector based on the identification and authentication of the user subject and possession of encryption key object.

- **Identification and Authentication**
  The TOE requires a user to authenticate at pre-boot prior to any other actions involving encryption/decryption access of protected data. The TOE allows the following means of authenticating:

  1) Fixed User ID and strong password

  The TOE requires the computer to restart after 3 consecutive failed login attempts. This prevents unauthorized login programs from running on the environment.

- **Security Management**
  The TOE provides sufficient Management Tools to implement and organize the Security Policies addressed by the TOE. These tools reflect on the authentication information, user and group policies, audit transaction information and design, and cryptographic data.

- **Security Audit**
  The TOE will track and save security related transactions and save them in a protected storage area. The audit log is restricted and can only be viewed by authorized users. The audit information provides a time stamp as to when the transaction was filed, and the event information of the transaction.

  The audit information can be viewed in an easily read format by authorized administrator. It is protected against malicious modifications or removal.

- **Session Control**
  The TOE shall maintain user session providing a way to set a trusted communication between itself and local user. A number of settings allow locking or termination of current session after a certain period of inactivity or pre-defined period of time. Session lock could also be initiated by user. The history of login attempt is visible for analysis on successful session establishment.

- **Cryptographic Support**
  The TOE uses cryptographic functionality based on the certification requirements for FIPS 140-2 Level 2. Encryption keys are generated, destructed, and protected.

- **Protection of the TSF**
  The TOE provides security mechanisms that prevent security functions from being tampered with, e.g. Access Control. The TOE also performs Disk Integrity checks for possible tampering.

- **Fault Tolerance**
  The TOE shall recover from an interrupted disk encryption operation (i.e., where only partial encryption of the disk has been achieved) where the interruption is due to loss of power, physical anomaly or attack, contention from another process in the OS environment or system faults require a reboot. The TOE should protect so no information is lost in the TOE environment.

2.2.2 Physical Boundaries

As for software product, physical boundaries of TOE are defined by the files and information stored on the computer where it is installed. The TOE can be installed on any x86 compatible computer running Microsoft Windows 2000/XP/2003. TOE functions are implemented uniformly across all listed OS platforms and do not interfere with any of them.

The TOE physical boundary consists of:

- SecureDoc code in MBR to load SecureDoc pre-boot authentication module
- The SecureDoc Space (a protected area on each hard drive or removable media where TOE-specific security information and user database are stored)

- All files within the C:\Program Files\WinMagic\SecureDoc-NT directory (or other folder in which SecureDoc resides):

  BINUPD.EXE
  bkgd.bin
  Boot_msg.txt
  BuildMBR.EXE
  chkboot.dat
  CSDUtil.EXE
  Default.ini
  E0.BIN
  ExtCode.bin
  Font.bin
  H1.bin
  H2.bin
  H3.bin
  H4.bin
  H5.bin
  Hands.bin
  L0.ovl
  L1.OVL
  L2.OVL
  L3.OVL
  MBRCODE.BIN
  menu.bin
  Radio.bin
  Radio_s.bin
  README.TXT
  RemoveBL.EXE
  RRUtil.exe
  SDEmgRec.exe
  SDFile.exe
  SDHelp.chm
  SDKey.exe
  SDLogo.bin
  SDPin.exe
  SDSecFolder.exe
  SdWipe.exe
  SecurDoc.exe
  SecureDoc.pdf
  WzdSetup.exe

- The following files in C:\ Windows \System32 (or C:\Winnt\ System32 as appropriate):

  chktoki.dat
  chkuser.dat
  CommonDlg.dll
  Sdc.dll
  Sdck.dll
  SDClient.dll
  Sdd.dll
  SDDisk.dll
  SDDllRes.dll
  SDFileCL.exe
  SDInst.dll
  SDocGina.dll
  SDToki.dll
  Sduser.dll
  Sdlibeay.dll
  Uninst.dll
  WMcv.dll
  WMpki.dll

- The following files in C:\ Windows \System32\Drivers (or C:\Winnt\ System32 Drivers as appropriate):

  Chkdxp.dat
  PinFile.sys
  SDdisk2k.sys

3. Security Environment

The Security Environment describes the security surrounding the TOE environment and how the TOE should be used correctly to prevent such.

This section will deal with the Threats, Assumptions and Organizational Policies for the TOE environment.

3.1 Assumptions

The list of assumptions regarding the security aspects of the environment in which the TOE is intended to be used is as follows:

Assumption	Description
A.NO_EMSEC	The sensitivity of information assets under protection by the TOE in its environment do not exceed that for which electromagnetic emissions countermeasures are mandatory or recommended by the environment system's responsible authority.
A.NO_EVIL	The selection of personnel for administrative roles with respect to the deployment of TOE and use in the organization must include a proper background check of the individual or be justified by mitigating circumstances that provide the organization with the assurance that administrators will demonstrate competence in their duties and not deliberately misuse or subvert the TOE for non-secure, fraudulent or other improper purposes.
A.NO_OBSERV	The physical environment allows users to enter passwords without being directly observable by other users or potential threat agents.
A.SENS_INFO	The sensitivity of information assets under protection by the TOE in its environment do not exceed that for which the symmetric encryption algorithms supported by the TOE (i.e., DES, DES3 and AES) are recommended by the environment system's responsible authority.

3.2 Threats

The list of threats that target the assets that the TOE and environment are protecting is as follows:

Threat	Description
T.ACCESS	A non-authorized user of the TOE may access information or resources without having the permission from the person who owns, or is responsible, for the information or resource.
T.DATA_DEST	Execution of a disk format in MS-DOS mode while the logged-in computer is unattended, thereby converting the disk back to an unencrypted resource.
T.EAVESDRP	In the temporary absence of the authorized user during a login session, an unauthorized insider or unescorted visitor may access protected information.
T.KEY_LOSS	A non-hostile user may inadvertently forget the password to the key database created to encrypt information assets, denying access to data from authorized users in the organization.
T.MOVE_FILES	If the computer is unattended, a threat agent could move backup and temporary directories/files to unprotected drives in the immediate environment or network.
T.NEGLECT	An unauthorized agent may attack the integrity of the key database or other security-critical asset without detection by the administrator or system authority
T.OS_FAULT	An unrelated user process may cause an operating system protection fault while disk encryption is taking place and halt the encryption process before completion, resulting in incomplete or corrupted output.
T.PHYSICAL	Security-critical parts of the TOE may be subject to an inadvertent or careless physical attack by privileged users that may compromise security, e.g., loss / destruction of key database.
T.POWER	A power loss results in failure and possible corruption of the encryption process.
T.PRIVILEGE	Compromise of IT assets may occur as a result of actions taken by careless, willfully negligent or hostile administrators or other privileged users.
T.PWD_SHARE	If user passwords are shared, contrary to the organizational policy, an unauthorized agent could access confidential assets
T.DELBOOT	A malicious or neglect user may disable or destroy the protective authentication MBR used by the TOE.

Threat	Description
T.TF_LOCN	An unauthorized agent may gain access to sensitive information in a temporary file not protected by the TOE.
T.ACT_TRACE	A user may perform unauthorized action or a system event may change security environment that goes undetected.
T.INFO_REUSE	A user may gain unauthorized access to protected resource by using out-of-date authentication and encipherment information.
T.SEC_OPER	A malicious agent may modify initial state of TOE (binary code or environment) to affect security enforcing functions or bypass them thereby gaining unauthorized access to resources.

3.3 Organizational Security Policies

The Organizational Security Policies enforced by the TOE are as follows:

Policy	Description
P.MANAGE	An authorized user must have possibility to effectively manage TOE security functions through the corresponding interfaces provided with the TOE.

4. Security Objectives

This section defines the Security Objectives of the TOE and surrounding environment. The Security Objectives are used to counter the Threats listed for the TOE as well as to support the Assumptions made about the TOE environment and to enforce specified Organizational Security Policies. The Security Objectives are grouped as either IT on Non-IT

4.1 TOE Security Objectives

The Security Objectives of the TOE comprise the following:

Security Objective	Description
SO.ACCESS_CTL	The TOE must prevent access to data that has been written to a drive or partition protected by it to all subjects unable to initiate a session with the password associated with the drive on which the data resides.
SO.CRYPT_STD	The TOE must provide a choice of cryptographic algorithms and strengths based on key sizes with which to protect all protected data.
SO.LOCKKF	The TOE should lockout or disable a key file after a certain date or time.
SO.KEY_BKUP	The security officer must retain a key database maintaining copies of all keys used by users having access to system protected drives.
SO.RESTORE	The TOE should detect and restore interrupted encryption processes at the next power resumption.
SO.LOCKSTATE	The TOE should lock an initial user session.
SO.SEC_STATE	In the event of an error occurring, the TOE should preserve a secure state.
SO.USER_I&A	Users of the TOE should be reliably identified and authenticated before being permitted access to the TOE and the cryptography-related IT assets therein.
SO.LATTEMPT	The TOE should log all unsuccessful login attempts to a computer being protected.
SO.SPASSWORD	The TOE should require a user set a strong password for their key file
SO.TRANS_LOG	All user transactions in TOE and system events related to TOE should be logged and saved for verifications.

Security Objective	Description
SO.MANAGE	An authorized user must have possibility to manage security functions of the TOE through a set of attributes.
SO.RESID_INFO	The TOE must prevent using out-of-date information released under TSF control.
SO.SEC_OPER	The TOE must recognize potential modifications of its initial state before starting normal operations and providing possible access to resources. It also prevent malicious user from attempt to bypass security mechanism.

4.2 Environmental Security Objectives

4.2.1 IT Environmental Security Objectives

The IT Environmental Security Objectives are as follows:

Security Objective	Description
SO.TIME_SRC	IT environment of the TOE must provide a reliable time source for the TOE to provide accurate date and time for audit records.
SO.MULTI_AUTH	IT environment of the TOE should provide possibility for multi-factor authentication.

4.2.2 Non-IT Environmental Security Objectives

The non-IT Environmental Security Objectives comprise the following:

Security Objective	Description
SO.INSPECT	The TOE and its key database should be regularly inspected for signs of errors or attacks.
SO.KDB_PROT	Procedural and physical measures should be taken to prevent unauthorized individuals from gaining access to the TOE key database.
SO.NO_EMSEC	The sensitivity of information assets under protection by the TOE in its environment do not exceed that for which electromagnetic emissions countermeasures are mandatory or recommended by the environment system's responsible authority.

Security Objective	Description
SO.NO_EVIL	The system administration roles must be staffed by adequately trained, responsible and honest individuals who are not motivated to disable, degrade or subvert the operation of the TOE in the environment for personal gain or other purposes that contradict the security policies of the organization.
SO.NO_OBSERV	The physical environment allows users to enter passwords without being directly observable by other users or potential threat agents.
SO.PHYS_ACC	The PC or workstation hosting protected drives/partitions must be located in a lockable cabinet or room, or be logged-out or powered down when unattended.
SO.PWD_SHARE	The sharing of passwords among users should be forbidden or, if required to enforce role or group access, strictly confined to the users who hold the specified organizational role or are members of the specified group authorized to access the information assets protected by the shared password.
SO.SEC_AWARE	Users should be properly trained in Organizational security policy and have awareness of security procedures
SO.SENS_INFO	The sensitivity of information assets under protection by the TOE in its environment do not exceed that for which the symmetric encryption algorithms supported by the TOE (AES) are recommended by the environment system's responsible authority.
SO.SYS_BKUP	The system must have regular backups that include protected drives.
SO.DELBOOT	Administrator should prepare the Emergency Diskette when it is recommended by TOE to be able to restore MBR damaged by malicious or neglect user.
SO.TF_LOCN	All temporary files that may contain sensitive information, and are either generated or used by their applications, must be located on encrypted drives in accordance with the recommendations of the TOE User Guide.
SO.UNATTEND	Users must be trained on correct procedures to follow when their PCs and workstations are unattended, and password-enabled screen savers and similar protective software should be used if their use is warranted

5. IT Security Requirements

5.1 Security Functional Requirements

This section contains the security functional requirements for the TOE. The following CC Part 2 Components are referenced, with definitions reproduced verbatim or completed where required. Completed definition text (i.e., assignments, selections or other added text not defined by the CC) is indicated below by *italics*.

Security Functional Class	Security Functional Components
Security Audit (FAU)	Security alarms (FAU_ARP.1)
	Audit data generation (FAU_GEN.1)
	User identity association (FAU_GEN.2)
	Potential violation analysis (FAU_SAA.1)
	Audit review (FAU_SAR.1)
	Restricted audit review (FAU_SAR.2)
	Protected audit trail storage (FAU_STG.1)
	Action in case of possible audit data loss (FAU_STG.3)
Cryptographic Support (FCS)	Cryptographic key generation (FCS_CKM.1)
	Cryptographic key access (FCS_CKM.3)
	Cryptographic key destruction (FCS_CKM.4)
	Cryptographic operation Data encryption and decryption (FCS_COP.1 (a))
	Cryptographic operation Cryptographic key encryption and decryption (FCS_COP.1 (b))
	Secure hash (FCS_COP.1 (c))
	Message authentication (FCS_COP.1 (d))
User Data Protection (FDP)	Complete access control (FDP_ACC.2)
	Security attribute based access control Disk access control (FDP_ACF.1)
	Subset residual information protection (FDP_RIP.1)
Identification and authentication (FIA)	Authentication failure handling (FIA_AFL.1)
	User attribute definition (FIA_ATD.1)
	Verification of secrets (FIA_SOS.1)
	User authentication before any action (FIA_UAU.2)
	Protected authentication feedback (FIA_UAU.7)
	User identification before any action (FIA_UID.2)

Security Functional Class	Security Functional Components
Security Management (FMT)	Management of security functions behavior Audit (FMT_MOF.1(a))
	Management of security functions behavior Identification and authentication (FMT_MOF.1(b))
	Management of security functions behavior Cryptographic support (FMT_MOF.1(c))
	Management of security attributes (FMT_MSA.1)
	Secure security attributes (FMT_MSA.2)
	Static attribute initialization (FMT_MSA.3)
	Management of TSF Data Audit trail (FMT_MTD.1 (a)
	Management of TSF Data Security relevant roles (FMT_MTD.1 (b))
	Management of TSF Data Encryption keys (FMT_MTD.1 (c))
	Management of TSF Data Authentication data (FMT_MTD.1 (d))
	Management of TSF Data Password policy (FMT_MTD.1 (e))
	Management of TSF Data Authentication failure (FMT_MTD.1 (f))
	Management of limits on TSF Data (authentication failure) (FMT_MTD.2)
	Revocation (FMT_REV.1)
	Time-limited authorization (FMT_SAE.1)
	Specification of Management Functions (FMT_SMF.1)
	Security management roles (FMT_SMR.1)
Protection of the TSF (FPT)	Abstract machine testing (FPT_AMT.1)
	Fail secure (FPT_FLS.1)
	Automated recovery (FPT_RCV.2)
	Non-bypassability of the TSP (FPT_RVM.1)
	TSF domain separation (FPT_SEP.1)
	Self testing (FPT_TST.1)
Resource Utilization (FRU)	Degraded fault tolerance (FRU_FLT.1)
TOE Access (FTA)	TSF-initiated session locking (FTA_SSL.1)
	User-initiated session locking (FTA_SSL.2)
	TOE access history (FTA_TAH.1)
Trusted Path/Channels (FTP)	Trusted Path (FTP_TRP.1)

5.1.1 Security Audit (FAU)

Security Alarms (FAU_ARP.1)

FAU_ARP.1.1

The TSF shall [***deny access to environment***] upon detection of a potential security violation.

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 [***minimum***] level of audit; and
c) [***All failed authentication with detailed information***];
d) [***Any disk integrity check failures***];

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;
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [***error specification***]

User identity association (FAU_GEN.2)

FAU_GEN.2.1

The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Potential violation analysis (FAU_SAA.1)

FAU_SAA.1.1

The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.

FAU_SAA.1.2

The TSF shall enforce the following rules for monitoring the audited events:

a) Accumulation or combination of [***access or authentication related events***] known to indicate a potential security violation;
b) [***Failure of the disk integrity check***];

Audit Review (FAU_SAR.1)

FAU_SAR.1.1

The TSF shall provide [***authorized users***] with the capability to read [***information about login attempts, integrity check events, and security related users' or administrators' actions***] 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.

Restricted audit review (FAU_SAR.2)

FAU_SAR.2.1

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

Protected audit trail storage (FAU_STG.1)

FAU_STG.1.1

The TSF shall protect the stored audit records from unauthorized deletions.

FAU_STG.1.2

The TSF shall be able to [***detect***] unauthorized modifications to the audit records in the audit trial.

Action in case of possible audit loss (FAU_STG.3)

FAU_STG.3.1

The TSF shall [***save all existing audit records and create a new trail for new records***] if the audit trail exceeds the [***available space limit***].

5.1.2 Cryptographic Support (FCS)

Cryptographic key generation (FCS_CKM.1)

FCS_CKM.1.1

The TSF shall generate cryptographic keys in accordance with cryptographic key generation algorithm ***ANSI X9.31 AES*** and specified cryptographic key sizes ***256 bit*** that meets the following:
[***FIPS 140-2, Section 4.7.2 Key Generation***].

Cryptographic key access (FCS_CKM.3)

FCS_CKM.3.1

The TSF shall perform [***cryptographic key archival***] in accordance with a specified cryptographic key access method [***duplicate***] that meets the following:
[***FIPS 140-2, Section 4.7.5 Key Storage***].

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 [***zeroization***] that meets the following: [***FIPS 140-2, Section 4.7.6 Key Zeroization***].

Cryptographic Operation (FCS_COP.1)

FCS_COP.1.1 (a)

The TSF shall perform [***data encryption and decryption***] in accordance with a specified cryptographic algorithm [**AES**] and cryptographic key size [**256 bits**] that meets the following: [***FIPS 197***].

FCS_COP.1.1 (b)

The TSF shall perform [***key encryption and decryption***] in accordance with a specified cryptographic algorithm [**AES**] and cryptographic key size [**256 bits**] that meets the following: [***FIPS 197***].

FCS_COP.1.1 (c)

The TSF shall perform [***secure hash***] in accordance with a specified cryptographic algorithm [**SHA-1, SHA-2**] and size [**20, 32, 48, 64 bytes**] that meets the following: [***FIPS 180-2***].

FCS_COP.1.1 (d)

The TSF shall perform [***message authentication***] in accordance with a specified cryptographic algorithm [**HMAC**] and cryptographic key size [**256 bits**] that meets the following: [***FIPS 198***].

5.1.3 User Data Protection (FDP)

Complete access control (FDP_ACC.2)

FDP_ACC.2.1

The TSF shall enforce the [***Access Control SFP***] on [***subjects: authorized users and objects: physical media, disk partitions, file system objects, security-relevant operations, security attributes***] and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2

The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

Security attribute based access control (FDP_ACF.1)

FDP_ACF.1.1

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

- ***authenticated users with user identity security attribute***
- ***encrypted objects with encryption key security attribute***
- ***cryptographic services with user identity and security-relevant privileges security attributes***

].

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled objects is allowed:

a) [***Authenticated users will be granted read/write access to an encrypted object (physical media, disk partition, file system object) if their identity is associated with the Data Encryption Key that decrypts the object***]

b) [***Authenticated users will be granted read access to key database if their identity is associated with the Key Encryption Key that protects it***]

c) [***Authenticated users will be granted access to their own security attributes to modify authentication data***]

FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

a) [***Authorized administrators may access(read/write, encrypt/decrypt) protected objects through the use of a controlled emergency key database***]

b) [***Authenticated administrators will be allowed to perform specific operations (media conversion, TOE configuration, key generation/backup, user and disk locking, viewing audit log) according to privileges assigned to their security-relevant role***]

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the following rules:

a) [***Authenticated user will be denied access to the protected system if that is prohibited by authorized administrator***]

b) [***Authenticated user will be denied access to the protected system if his key file is expired***]

c) [***Authenticated user will be denied access to the protected system as a result of exceeding of limit of failed login attempts***]

Subset Residual Information Protection (FDP_RIP.1)

FDP_RIP.1.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***de-allocation of the resource from***] the following objects: [***user key database***].

5.1.4 Identification and authentication (FIA)

Authentication failure handling (FIA_AFL.1)

FIA_AFL.1.1

The TSF shall detect when [***"an administrator configurable positive integer within 1 – 100"***] unsuccessful authentication attempts occur related to [***a user logging into a computer that is protected by the TOE***].

FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [***lock the key file of that user on this computer and block user's access to it until another user successfully login to computer***].

User attribute definition (FIA_ATD.1)

FIA_ATD.1.1

The TSF shall maintain the following list of security attributes belonging to individual users:
[
- ***User keys***
- ***User privileges***
- ***User key files***

]

Verification of Secrets (FIA_SOS.1)

FIA_SOS.1.1

The TSF shall provide a mechanism to verify that secrets meet the following:

a) [***For each attempt to use the fixed password authentication mechanism, the probability that a random attempt will succeed is less than one in 100,000,000,000***]

b) [***For multiple attempts to use the fixed password authentication mechanism during a one minute period,***

the probability that a random attempt during that minute will succeed is less than one in 10,000,000,000]

User authentication before any action (FIA_UAU.2)

FIA_UAU.2.1

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

Protected authentication feedback (FIA_UAU.7)

FIA_UAU.7.1

The TSF shall provide only [***asterisks to be displayed***] to the user while the authentication is in progress.

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.

5.1.5 Security Management (FMT)

Management of security functions behavior- Audit (FMT_MOF.1 (a))

FMT_MOF.1.1 (a)

The TSF shall restrict the ability to [***enable, disable, modify the behavior of***] the function [***audit***] to [***authorized administrators***].

Management of security functions behavior – Identification and authentication (FMT_MOF.1 (b))

FMT_MOF.1.1 (b)

The TSF shall restrict the ability to [***enable, disable, modify the behavior of***] the function [***identification and authentication***] to [***authorized administrators***].

Management of security functions behavior –Cryptographic support (FMT_MOF.1 (c))

FMT_MOF.1.1 (c)

The TSF shall restrict the ability to [***enable, disable, modify the behavior of***] the function [***cryptographic support***] to [***authorized administrators***].

Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1

The TSF shall enforce the [***Access Control SFP***] to restrict the ability to [***modify, delete, and create***] the security attributes [***user identifiers, encryption keys, security-relevant roles and privileges***] to [***authorized administrators***].

Secure security attributes (FMT_MSA.2)

FMT_MSA.2.1

The TSF shall ensure that only secure values are accepted for security attributes.

Static attribute initialization (FMT_MSA.3)

FMT_MSA.3.1

The TSF shall enforce the [***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 [***authorized administrator***] to specify alternative initial values to override the default values when an object or information is created.

Management of TSF Data – Audit Trail (FMT_MTD.1 (a))

FMT_MTD.1.1 (a)

The TSF shall restrict the ability to [***query, and export***] the [***audit trail***] to [***authorized administrators***].

Management of TSF Data – Security-relevant roles (FMT_MTD.1 (b))

FMT_MTD.1.1 (b)

The TSF shall restrict the ability to [***modify***] the [***security-relevant roles for users***] to [***authorized administrators***].

Management of TSF Data – Encryption keys (FMT_MTD.1 (c))

FMT_MTD.1.1 (c)

The TSF shall restrict the ability to [***initialize, delete***] the [***encryption keys***] to [***authorized administrators***].

Management of TSF Data – Authentication data (FMT_MTD.1 (d))

FMT_MTD.1.1 (d)

The TSF shall restrict the ability to [***modify***] the [***authentication data***] to [***authorized administrators, and users (for their own authentication data)***].

Management of TSF Data – Password Policy (FMT_MTD.1 (e))

FMT_MTD.1.1 (e)

The TSF shall restrict the ability to [***modify***] the [***requirements for password selection (such as composition, length, complexity, history, timing, etc.)***] to [***authorized administrators***].

Management of TSF Data – Authentication Failure (FMT_MTD.1 (f))

FMT_MTD.1.1 (f)

The TSF shall restrict the ability to [***modify***] the [***settings for handling authentication failures***] to [***authorized administrators***].

Management of limits of TSF Data – Authentication Failure (FMT_MTD.2)

FMT_MTD.2.1

The TSF shall restrict the specification of limits for [***the unsuccessful authentication attempts threshold***] to [***authorized administrators***].

FMT_MTD.2.2

The TSF shall take the following action, if the TSF data are at, or exceed the indicated limits: [***disable the user account for either a specified duration or until unlocked by an authorized administrator (as specified by an authorized administrator***].

Revocation (FMT_REV.1)

FMT_REV.1.1

The TSF shall restrict the ability to revoke security attributes associated with the [***users***] within the TSC to [***authorized administrators***].

FMT_REV.1.2

The TSF shall enforce the rules: [***Revocation will take place on the next login of the user***].

Time-limited authorization (FMT_SAE.1)

FMT_SAE.1.1

The TSF shall restrict the capability to specify an expiration time for [***access to restricted data, user accounts and user passwords***] to [***Security Administrators***].

FMT_SAE.1.2

For each of these security attributes, the TSF shall be able to [***lock access to restricted data, lock user accounts from accessing restricted data, force***

users to change passwords] after the expiration time for the indicated security attribute has passed.

Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1

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

a) [***Management of user accounts (create, delete, modify)***]
b) [***Management of security settings, including encryption***]
c) [***Review of audit trail***]

Security roles (FMT_SMR.1)

FMT_SMR.1.1

The TSF shall maintain the roles [***User and Administrator***].

FMT_SMR.1.2

The TSF shall be able to associate users with roles.

5.1.6 Protection of the TSF (FPT)

Abstract machine testing (FPT_AMT.1)

FPT_AMT.1.1

The TSF shall run a suite of tests [***during initial start-up***] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

Failure with preservation of secure state (FPT_FLS.1)

FPT_FLS.1.1

The TSF shall preserve a secure state when the following types of failures occur [***power failure or physical anomaly inflicting temporary failure of disk operations while disk conversion is occurring***].

Automated recovery (FPT_RCV.2)

FPT_RCV.2.1

When automated recovery from [***a failure or service discontinuity***] is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

FPT_RCV.2.2

For [**power failure or physical interruption]**, the TSF shall ensure the return of the TOE to a secure state using automated procedures.

Non-bypassability of the TSP (FPT_RVM.1)

FPT_RVM.1.1

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

TSF domain separation (FPT_SEP.1)

FPT_SEP.1.1

The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by un-trusted subjects.

FPT_SEP.1.2

The TSF shall enforce separation between the security domains of subjects in the TSC.

Self Testing (FPT_TST.1)

FPT_TST.1.1

The TSF shall run a suite of self tests [***during initial start-up or at the request of authorized user***] to demonstrate the correct operation of [***the TSF***].

FPT_TST.1.2

The TSF shall provide authorized users with the capability to verify the integrity of the [***TSF data***].

FPT_TST.1.3

The TSF shall provide the authorized users with the capability to verify the integrity of stored TSF executable code.

5.1.7 Resource Utilization (FRU)

Fault tolerance (FRU_FLT.1)

FRU_FLT.1.1

The TSF shall ensure the operation of encryption of [***disk, partition or removable media***] when the following failures occur: [***power failure or physical anomaly inflicting temporary failure of disk operation while encryption is occurring***].

5.1.8 TOE Access (FTA)

TSF-initiated session locking (FTA_SSL.1)

FTA_SSL.1.1

The TSF shall lock an interactive session after [***administrator set time of user inactivity***] by:

a) **clearing or overwriting display devices, making the current contents unreadable;**

b) **disabling any activity of the user's data access/display devices other than unlocking the session.**

FTA_SSL.1.2

The TSF shall require the following events to occur prior to unlocking the session:

a) [***User must authenticate itself to unlock the session***]

User-initiated session locking (FTA_SSL.2)

FTA_SSL.2.1

The TSF shall allow user-initiated locking of the user's own interactive session, by:

a) **clearing or overwriting display devices, making the current contents unreadable;**

b) **disabling any activity of the user's data access/display devices other than unlocking the session.**

FTA_SSL.2.2

The TSF shall require the following events to occur prior to unlocking the session:

a) [***User must authenticate itself to unlock the session***]

TOE access history (FTA_TAH.1)

FTA_TAH.1.1

Upon successful session establishment, the TSF shall display the: [***date, time method***] of the last session establishment of the user.

FTA_TAH.1.2

Upon successful session establishment, the TSF shall display the: [***date, time, method***] of the last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the last successful session establishment.

FTA_TAH.1.3

The TSF shall not erase the access history information from the user interface without giving the user an opportunity to review the information.

5.1.9 Trusted Path / Channels (FTP)

Trusted Path (FTP_TRP.1)

FTP_TRP.1.1

The TSF shall provide a communication path between itself and [***local users***] that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2

The TSF shall permit [***local users***] to initiate the communication via the trusted path.

FTP_TRP.1.3

The TSF shall require the use of the trusted path for [***initial user authentication***].

5.2 Security Assurance Requirements

5.2.1 Statement of Security Assurance Requirements

The following assurance requirements defined in CC Part 3 are applied to comply with the EAL 4 requirements (see CC Part 3, Table 6.2).

Security Assurance Class	Security Assurance Component
Configuration Management (ACM)	ACM_AUT.1 Partial CM Automation
	ACM_CAP.4 Generation Support and acceptance procedures
	ACM_ SCP.2 Problem tracking CM coverage
Delivery and Operation (ADO)	ADO_ DEL.2 Detection of modification
	ADO_IGS.1 Installation, generation, and start-up procedures
Development (ADV)	ADV_FSP.2 Fully defined external interfaces
	ADV_HLD.2 Security enforcing high-level design
	ADV_IMP.1 Subset of 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

Security Assurance Class	Security Assurance Component
Guidance Documents (AGD)	AGD_ADM.1 Administrator guidance
	AGD_USR.1 User guidance
Life Cycle Support (ALC)	ALC_DVS.1 Identifications of security measures
	ALC_LCD.1 Developer defined life-cycle model
	ALC_TAT.1 Well defined development tools
Tests (ATE)	ATE_COV.2 Analysis of coverage
	ATE_DPT.1 Testing: high-level design
	ATE_FUN.1 Functional Testing
	ATE_IND.2 Independent Testing - Sample
Vulnerability Assessment (AVA)	AVA_MSU.2 Validation of analysis
	AVA_SOF.1 Strength of TOE security function evaluation
	AVA_VLA.2 Independent vulnerability analysis

5.2.2 Statement of Strength of TOE Security Function

The Identification and Authentication function of TSF (see subsection 6.1.2) invokes user authentication mechanism enforced by FIA_UID.2 and FIA_UAU.2 requirements. The probabilistic fixed password mechanism implemented in TSF satisfies FIA_SOS.1 that provides SoF-medium level of strength. AVA_SOF.1 assurance requirement is applied to TSF to confirm the claimed SoF-medium level.

5.3 IT Environment Security Requirements

The IT environment security requirements define functional and/or assurance requirements to be satisfied by the IT environment. The requirements are satisfied through hardware, firmware and/or software external to the TOE needed in order to ensure that the security objectives for the TOE are archived.

5.3.1 Reliable time stamps (FPT_STM.1)

The TSF shall be able to provide reliable time stamps for its own use.

5.3.2 Multiple authentication mechanisms (FIA_UAU.5)

The TSF shall provide multi-factor authentication based on the use of the following cryptographic devices:

- **[Tokens]**
- **[Smartcards]**
- **[Biometrics]**
- **[TPM]**

6. TOE Summary Specification

6.1 Statement of TOE Security Functions

The TOE IT Security Functions and their specifications are listed as follows.

6.1.1 Access Control

The TOE controls access by an identified and authenticated user to those objects that are files on a disk drive or sector that have been encrypted by the TOE. All read and write operations to the encrypted disk or sectors are mediated by the TOE.

The TOE defines the ***Access Control SFP*** as a set of rules that determines what kind of access an authenticated user operating in a certain role, has to a security-relevant object. The rules require:

1) User identification and authentication
2) Association of user identity with encryption keys
3) Restriction of actions available for user to those specified for the role which user is operating in
4) Locking of user access due to expiration of security attributes, exceeding the limit of failed attempts or administrator's decision

The TOE enforces all low-level reading and writing operations to incorporate decryption, respectively encryption, operations that utilize the secret key and user-selected algorithm associated with the currently identified and authenticated user subject.

Note: At installation, the Administrator is prompted to make an emergency key database to access all disks/partitions and this must be updated with each new key generation and deletion.

The TOE keeps Data Encrypting Keys (DEK) encrypted with Key Encrypting Keys (KEK) which may be assigned to different user. The set of assigned keys is kept together with available privileges in user's personal key file and is encrypted once more with key file encryption key.

All key files regulating access to computer are stored in SecureDoc Space like a user database. The TOE enforces ***Access Control SFP*** preventing unauthorized users from access to that area by encrypting sensitive information inside it. The TOE can assign security-related roles to access this stored information. Authorized administrators can perform the following actions:

1) Modify key database in key file
2) Lock/Unlock users
3) Lock/Unlock partitions
4) Restrict operations on removable media
5) Change TOE active configuration and options

The Access Control function is designed to satisfy the following security functional requirements:

- FDP_ACC.2
- FDP_ACF.1
- FDP_RIP.1

6.1.2 Identification and Authentication

The TOE enforces **Identification & Authentication SFP** to require user to be authenticated by the TOE at pre-boot prior to any other actions involving encryption/decryption access of protected data.

There are the following methods that authentication can take place:

1) User ID and strong password

As part of **Identification & Authentication SFP** the administrator enforces the following password selection policy through the password rules to provide SoF-medium level of security for password-based authentication mechanism:

1) Password length must be at least 8 characters to resist the "brute force attack"
2) Password must include letters (both cases), digits and special symbols to resist "dictionary attack"
3) Password must have reasonable validity period
4) Password history must keep several last passwords to prevent user from using the same password once again
5) Password Hint and Self-Help Password recovery must not be available for user

The TOE provides only asterisks (*) display as user inputs characters of authentication strings at pre-boot.

The TOE will lock the computer after 3 unsuccessful login attempts. The computer will be required to be restarted to continue log attempts.

The TOE will log all unsuccessful authentications at pre-boot. This information will be saved in the audit trail. A message will also display in Windows for the user, showing the number of unsuccessful login attempts since the last successful attempt was made.

The TOE can be configured to lock down the computer if the maximum number of unsuccessful logins has been reached. If this occurs, only the TOE administrator will be able to unlock the computer. Only the administrator of the TOE can configure this.

Each user that logs into the TOE has individual privileges that are set for them by the TOE administrator.

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

- FIA_AFL.1
- FIA_ATD.1

- FIA_SOS.1
- FIA_UAU.2
- FIA_UAU.7
- FIA_UID.2

6.1.3 Security Management

Users of the TOE are assigned rights on "User Privileges" and "Administrator Privileges".

What a user can do is enforced by **Access Control SFP** and all depends on which privileges were assigned to the user.

By default, a user is only assigned "User Privileges". These rights will only allow a user to log into their computer to view information, and change the password to their key file.

The "Administrative Privileges" will allow for much more detailed rights. Such rights include:

1) Encrypting/decrypting;
2) Selective security modifications;
3) Creation/deletion of users;
4) Lock of disks/partitions/removable media;
5) View of audit tracking, etc.

The TOE allows for expiry dates to be set on privileges and passwords.

In addition to normal privileges set, the TOE allows for local administrators to have a hierarchy of rights. These rights may include:

1) Creation of emergency disks;
2) Encryption of either: hard disks, floppy disks, etc.;
3) Disk Integrity Changes to a user's system;
4) Modifying specific profiles on users machines;
5) Setting specific Disk Access Lock configurations;
6) Modifications to the Audit Log information;
7) Creation and modifications to encryption keys;

The administrator must have the protection key to perform any functions on a typical user computer. Without the protection key, the administrator cannot access the computer.

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

- FMT_MOF.1
- FMT_MSA.1
- FMT_MSA.2
- FMT_MSA.3
- FMT_MTD.1
- FMT_MTD.2
- FMT_REV.1

- FMT_SAE.1
- FMT_SMF.1
- FMT_SMR.1

6.1.4 Security Audit

The TOE enforces **Audit SFP** by recording all related user events in Windows and at pre-boot. In Windows, the TOE related events are recorded. At pre-boot, all user authentication events are recorded.

The audit function starts when the computer running the TOE starts-up. The audit function shutdowns when the operating system shutdowns / restarts the computer.

Events that will be audited include:

1) Start-up and shutdown of the audit functions
2) All Disk Integrity events at pre-boot and Windows
3) User actions within the TOE in Windows
4) Administration actions within the TOE in Windows

The audit record consists of:

1) Date/Time
2) User ID
3) Action
4) Result (*Successful or Failure)*
5) Error Number (*If an error is detected.)*
6) Error Text (*If an error is detected.)*

When the TOE creates an audit transaction, it saves the information, as outlined above, to an audit file. This file is used locally to track, as well, is sent to the TOE administrator to track activity. The information in the audit file is encrypted, and cannot be tampered with. The audit file cannot be deleted or changed. If the TOE detects that the audit file was tampered with, it will alarm the administrator.

Only the administrators of the TOE can access and view audit transactions. A successful login must be obtained for access.

Audit transactions can be viewed using the TOE installed on a user's computer, or the IT Environment. Audit transactions can be searched and sorted.

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

- FAU_ARP.1
- FAU_GEN.1
- FAU_GEN.2
- FAU_SAA.1
- FAU_SAR.1
- FAU_SAR.2
- FAU_STG.1
- FAU_STG.3

6.1.5 Session control

The TOE shall provide control of user's session. It set trusted path between itself and local users after computer restarts by embedding into regular boot procedure initiated by BIOS before any other software has chance to get control. Initial authentication at that stage immediately involves TSP enforcing functions and maintains the following operation in secure manner.

The TOE enforces ***Access Control SFP*** preventing other users from accessing an active session of currently authenticated user.

The current session may be interrupted by user or due to certain period of inactivity. In this case the session is locked under control of TOE and new authentication is needed to resume the session. The session could also be terminated after certain pre-set period of time.

The TOE shows the access history to the user upon successful session establishment providing that it won't be erase otherwise as on user direction.

The Session control security function is designed to satisfy the following security functional requirements:

- FTP_TRP.1
- FTA_SSL.1
- FTA_SSL.2
- FTA_TAH.1

6.1.6 Cryptogr*a*phic Support

TOE performs all cryptographic operations through the Cryptographic Engine that enforces **Cryptographic Keys SFP**. The Cryptographic Engine is certified for FIPS 140-2 Level 1 (Certificate #699) and Level 2 (Certificate #698).

The following NIST certified cryptographic algorithms are available:

Algorithm	Cryptographic Function	Modes / Mechanisms	Output Block (bytes)	Key Size (bits)	Certificate #
AES	Encipherment	ECB, CBC	16	256	359
SHA	Hashing	SHA-1,256, 384, 512	20, 32, 48, 64		434
HMAC	Message authentication	SHA-1, 256, 384, 512	20, 32, 48, 64	256	158
PRNG	Random number generation	ANSI X9.31 AES	16	256	172

The TOE generates keys via PRNG specified in the table above.

The TOE deletes keys by zeroization.

The TOE supports the use of key archiving. The encryption keys can be copied to a remote location for recovery purposes. The file containing the encryption key is protected by encryption.

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

- FCS_CKM.1
- FCS_CKM.3
- FCS_CKM.4
- FCS_COP.1

6.1.7 Protection of the TSF

The TOE shall recover from an interrupted disk encryption operation (i.e., where only partial encryption of the disk has been achieved) where the interruption is due to loss of power, physical anomaly or attack, contention from another process in the OS environment or system faults require a reboot.

Cryptographic keys and encrypted data could not be accessed unless the user has been successfully authenticated and has appropriate privileges.

The code running in pre-boot, kernel and user modes is separated from the each other. The code is also not accessible by other applications.

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

- FPT_AMT.1
- FPT_FLS.1
- FPT_RCV.2
- FPT_RVM.1
- FPT_SEP.1
- FPT_TST.1

6.1.8 Fault Tolerance

The TOE shall recover from an interrupted disk encryption operation (i.e., where only partial encryption of the disk has been achieved) where the interruption is due to loss of power, physical anomaly or attack, contention from another process in the OS environment or system faults require a reboot.

The Resource Utilization function is designed to satisfy the following security functional requirements:

- FRU_FLT.1

6.2 TOE Security Assurance Measures

The following assurance measures are applied to satisfy the Common Criteria EAL4 assurance requirements:

- **Process Assurance**
- **Delivery and Guidance**
- **Design Documentation**
- **Tests**
- **Vulnerability Assessment**

6.2.1 Process Assurance

The configuration Management system is applied at WinMagic with the following goals in mind:

- Procedures are in place to track and control all changes made to the TOE
- Specific tools are used to control access and changes to the TOE
- All transactions are followed using strict configuration management policies for the implementation representation, design, tests, user and administrator guidance, the CM documentation, and security flaws.
- Bugs and issues reported by customers of during production testing are controlled by bug tracking environment.

These events are addressed in detail in the following main documents:

- **WinMagic Software Development Process**
- **SecureDoc Configuration Management Plan**
- **SecureDoc Disk Encryption Configuration List**
- **SecureDoc Versioning and Release System**
- **SecureDoc Disk Encryption Building Procedure**
- **WinMagic Software Acceptance Plan**
- **WinMagic Software Development Security**
- **WinMagic Bug Tracking Process**

WinMagic guarantees the adequacy of the specified development and maintenance procedures through the life-cycle management plan of the TOE. WinMagic enforces security controls for the development environment to provide confidentiality and integrity during the TOE design and implementation. WinMagic ensures these results through an in-depth model of life-cycle of the TOE and reliable development tools. WinMagic documents all processes during the implementation; all procedures and information are documented in the documents listed above.

Assurance Requirements: ACM_AUT.1, ACM_CAP.4, ACM_SCP.2, ALC_DVS.1, ALC_LCD.1, ALC_TAT.1

6.2.2 Delivery and Guidance

WinMagic provides the necessary documentation and procedural information necessary for the identification of the TOE, its secure installation and start-up as well as for the detection of any unauthorized changes to the TOE. The delivery procedures describe the instructions used to determine if any modifications were made to the TOE during delivery.

WinMagic provides the necessary administrative and user guidance to guarantee that the TOE is configured using a secure manner. The proper information will prevent any compromise of security the TOE provides. The administrator and user guidance is documented in the SecureDoc Disk Encryption v.4.3C Administrative and User's Guide.

Corresponding documents are the following:

- **WinMagic Software Delivery Procedure** document describes process of delivery TOE to customers' sites
- **SecureDoc Disk Encryption Setup Guide** document contains instructions on initial installation of the TOE
- **WinMagic SecureDoc Corporate Edition Guide** document provides a manual for administrators and regular users on installation, usage and maintenance of the TOE

Assurance Requirements: ADO_DEL.2, ADO_IGS.1, AGD_ADM.1, AGD_USR.1

6.2.3 Design Documentation

WinMagic ensures that an extensive set of design documents are written during the development stage of SecureDoc Disk Encryption v.4.3C that describe all related aspects of the TOE security design, architecture, mechanisms, and interfaces. The following comprise the full documentation list:

- **SecureDoc Disk Encryption v.4.3C Functional Specification** document describes the interfaces and functionality of the TOE.
- **SecureDoc Disk Encryption v.4.3C High Level** Design document represents the architecture of the TOE through a set of subsystems.
- **SecureDoc Disk Encryption v.4.3C Low Level Design** document details the functionality of subsystems into modules.
- **SecureDoc Disk Encryption v.4.3C Implementation** document contains source code implementing the TOE.
- **SecureDoc Disk Encryption v.4.3C Representation Correspondence** document analyzes the correspondence between different representations the TOE.
- **SecureDoc Disk Encryption v.4.3C Security Policy Model** document presents an informal security model for the TOE.

Analysis of correspondence between the different representations of the TOE in design documents with corresponding evidence is provided separately. This will be accomplished by mapping between the TSF and components comprising the TOE.

Assurance Requirements: ADV_FSP.2, ADV_HLD.2, ADV_LLD.1, ADV_IMP.1, ADV_RCR.1, ADV_SPM.1

6.2.4 Tests

The test documentation for the TOE has been designed to describe how all security interfaces have been tested through the use of test cases. These procedures are in place to ensure the implementations are correct and functioning properly. The test documentation describes the actual step-by-step tests, the procedures to successfully execute the tests, the expected results, and a set of results from running the tests on the evaluated product in the amount enough to provide an independent testing. The test documentation also includes analysis of the coverage of TSF and functional requirements as well as depth of testing till the level of system architecture.

The documents supporting this assurance measure are:

- **SecureDoc Disk Encryption v.4.3C Validation Plan** document details test procedures and cases
- **SecureDoc Disk Encryption v 4.3C Testing Analysis** document analyzes the completeness of the testing of the TOE

Test results that prove the correct behaviour of the TOE, are provided as a set of separate documents in the Appendix to SecureDoc Disk Encryption v.4.3C Validation Plan.

Assurance Requirements: ATE_COV.2, ATE_DPT.1, ATE_FUN.1, ATE_IND.2

6.2.5 Vulnerability Assessment

The administrator guidance documentation describes the operation of SecureDoc Disk Encryption v.4.3C regarding how to maintain a secure state of TOE. The administrator guide is also useful to fully understand operating modes and security functions inside the scope of control of the TOE. The administrator document will provide consistent and reliable administrative reference.

The vulnerability analysis document addresses flaws that might be exploited in the TOE due to misuse of the product, weakness of underlying security mechanism or inability to resist penetration attack. The document addresses potential vulnerabilities and advises how to avoid them by correctly managing the TOE.

The SOF and vulnerability analysis are documented in the **SecureDoc Disk Encryption v.4.3C Vulnerability Analysis** document.

Assurance Requirements: AVA_MSU.2, AVA_SOF.1, AVA_VLA.2

7. Protection Profile Claims

This TOE does not claim conformance to a protection profile claim.

8. Rationale

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

- Security Objectives
- Security Requirements
- Requirements Dependencies
- Strength of Function
- Internal Consistency
- TOE Summary Specification

8.1 Security Objectives Rationale and Traceability

The purpose of this section is to show that the security objectives of the TOE and environment are appropriate to the security problem defined in the security environment section (see Section 3). This is accomplished through a set of tables that cross-reference threats, security policies and assumptions against the security objectives that address them. Each threat, policy or assumption is addressed by one or more security objective. Each security objective of the TOE (described in Section 4) addresses at least one threat, policy or assumption. An informal argument is provided to show, for each threat, policy or assumption, why the identified security objective provides an effective countermeasure that prevents an attack or mitigates risk to acceptable levels.

8.1.1 Security Objectives Rationale for Assumptions

The table below traces security objectives to assumptions as follows:

Assumption	Security Objective	Rationale
A.NO_EMSEC	SO.NO_EMSEC	Sensitivity of information protected by the TOE doesn't demand any countermeasures against electromagnetic emissions.
A.NO_EVIL	SO.NO_EVIL	Personnel authorized to maintain and administer the TOE must not be a source of disclosure or other compromising sensitive information of TOE and users.
A.NO_OBSERV	SO.NO_OBSERV	The TOE operates secure on condition that any sensitive information like passwords cannot be compromised by direct observation of user's input.
A.SENS_INFO	SO.SENS_INFO	Protection provided by symmetric algorithms supported by the TOE is appropriate for sensitivity of the information encrypted with those algorithms.

8.1.2 Security Objectives Rationale for Organization Policies

The following organization policies are addressed by TOE security objectives:

Policy	Security Objective	Rationale
P.MANAGE	SO.MANAGE	The organization security policies could be enforced through configuring TOE security attributes by authorized personnel.

8.1.3 Security Objectives Rationale for Threats

The TOE addresses the following security threats.

Threat	Security Objective	Rationale
T.ACCESS	SO.ACCESS_CTL SO.CRYPT_STD SO.LOCKKF SO.LOCKSTATE SO.USER_I&A SO.TRANS_LOG SO.LATTEMPT SO.SPASSWORD	The TOE controls access, restricting access to authorized users when installed correctly and administered by a security officer.
T.KEY_LOSS	SO.KEY_BKUP	The TOE provides a master key database to enable the administrator to access encrypted disks if a user's password is forgotten or unknown.
T.OS_FAULT	SO.RESTORE SO.SEC_STATE	The TOE has the capability to restore the interrupted encryption process when a system fault occurs. The TOE maintains a secure state when a system fault occurs.
T.POWER	SO.RESTORE SO.SEC_STATE	The TOE has the capability to restore the interrupted encryption process when a power failure has occurred. The TOE maintains a secure state when a power interruption occurs.
T.ACT_TRACE	SO.TRANS_LOG	The TOE records all user transactions and event that could affect TOE security functions into audit log.
T.INFO_REUSE	SO.RESID_INFO	The TOE makes unavailable any previous information content of a resource upon de-allocation.

Threat	Security Objective	Rationale
T.SEC_OPER	SO.SEC_OPER	The TOE doesn't allow bypassing its security enforcing functions and also blocks normal operations in case of possible modification of its initial state.

8.1.4 Environmental Security Objectives Rationale for Threats

The traceability of threats to the Environmental Security Objectives is as follows:

Threat	Security Objective	Rationale
T.ACCESS	SO.UNATTEND SO.MULTI_AUTH	The Organization must inform and train users in the proper procedures for unattended sessions in which sensitive information is accessible on their PC or workstations. Means for multi-factor authentication must be provided by TOE environment.
T.DATA_DEST	SO.SEC_AWARE SO.UNATTEND SO.PHYS_ACC	The Organization must train users on correct procedures to follow when their PCs and workstations are unattended, and must provide password-enabled screen savers and similar protective software if warranted
T.EAVESDRP	SO.PHYS_ACC	The Organization must provide adequate protection of the installed TOE PC or workstation to prevent access of a login session while the device is unattended.
T.MOVE_FILES	SO.UNATTEND SO.PHYS_ACC	The Organization must train users on correct procedures to follow when their PCs and workstations are unattended, and must provide password-enabled screen savers and similar protective software if warranted.
T.NEGLECT	SO.INSPECT, SO.KDB_PROT, SO.SEC_AWARE	The Organization must enforce regular inspection of the key databases and must protect such assets from theft or duplication. Users must be aware of consequences of mistreating access control information.

Threat	Security Objective	Rationale
T.PHYSICAL	SO.SYS_BKUP SO.SEC_AWARE	The Organization must enforce effective protection of TOE-critical information such as the key database to prevent loss of security-critical information on a hard drive or diskette.
T.PRIVILEGE	SO.SEC_AWARE SO.NO_EVIL	The Organization must provide administrators with adequate security awareness and training to prevent careless, willfully negligent, or hostile actions on the part of administrators.
T.PWD_SHARE	SO.SEC_AWARE SO.PWD_SHARE	The Organization must train users and provide security awareness that prevents the sharing of passwords among users unless sanctioned by security policy.
T.DELBOOT	SO.DELBOOT	The Organization must force administrators create SecureDoc Emergency Diskette
T.TF_LOCN	SO.TF_LOCN	The Organization must make sure that users properly locate application-specific temporary files and swap-areas on the protected disk / partition, if they contain sensitive information.
T.ACT_TRACE	SO.TIME_SRC	IT environment of the TOE must provide accurate date and time for audit records through a reliable time source.

The security objectives of the environment are considered effective in countering the effect of the threats cited if correctly applied by the organization. Security objectives of the TOE are effective in countering the threats identified in section 3.2 that are not found above. This traceability of threats to TOE-specific security objectives is found in section 8.1.3.

8.2 Security Requirements Rationale

8.2.1 Security Functional Requirements (SFRs) Rationale

The rationale for the TOE SFRs against the security objectives of the TOE is given in the table below. For each security objective of the TOE, a list of assigned TOE SFRs is given, followed by an argument stating how each SFR addresses or satisfies the security objective in question.

Security Objective	SFR	Rationale
SO.LATTEMPT	FAU_ARP.1	The TSF shall deny access to environment upon detection of a potential security violation:
	FAU_GEN.1	The TSF will generate an auditable event on each unsuccessful login attempt.
	FAU_GEN.2	The TSF shall be able to associate each auditable event with the identity of the user that caused the event.
	FTA_TAH.1	The TOE keeps the history of authentication attempt and provides user with the details on each successful session establishment.
SO.ACCESS_CTL	FDP_ACC.2	Access control to protected objects is based on cryptographic separation on file objects that are written to a protected disk drive or partition. All operations with the objects involved are covered and controlled by TSP via corresponding SF.
	FDP_ACF.1	FDP_ACF.1.1 provides that the access control to protected objects is based on their location (logical disk drive) attribute. FDP_ACF.1.2 provides that if a subject is correctly identified and authenticated, then read and write access to a controlled object (i.e., a file on a protected drive/partition) is granted through encryption/decryption operations employing a correct cryptographic key.
	FAU_SAR.2	No users except the ones with explicitly granted access could read audit information.

Security Objective	SFR	Rationale
	FAU_STG.1	Records in audit trial should be protected against unauthorized modifications.
SO.CRYPT_STD	FCS_COP.1	FCS_COP.1 provides that the TSF performs cryptographic operations in accordance with FIPS standards 180-2 (secure hash), 197 (AES encryption), and 198 (HMAC).
	FCS_CKM.1	FCS_CKM.1 provides key generation for encryption and decryption processes.
	FCS_CKM.3	FCS_CKM.1 provides key archival for the goal of emergency access to protected resources.
	FCS_CKM.4	FCS_CKM.4 provides key deletion for keys used in encryption and decryption processes by zeroization.
SO.KEY_BKUP	FDP_ACF.1	FDP_ACF.1.3 provides the administrator access to all disks/partitions with the use of an emergency key database, which requires no password in the event an employee, leaves without decrypting the disk or a password is forgotten.
SO.RESTORE	FRU_FLT.1	FRU_FLT.1 provides that the TOE can continue correct operation (following a reboot) and an automated recovery in the event of a power failure or system fault while encryption of a disk/partition is in progress.
SO.SEC_STATE	FPT_FLS.1	FPT_FLS.1 provides that an interrupted disk encryption process will result in the TOE returning to a secure state as defined in the partial security policy model.
	FPT_RCV.2	FPT_RCV.2 provides that for at least one type of service discontinuity (e.g., power failure) automated recovery to secure state, without human intervention will be performed by the TOE. For other types of failures or discontinuities, the TOE will require re-booting.
SO.LOCKKF	FDP_ACF.1	The TSF shall allow administrator to assign a validity period for a key file and lock the key file once it is expired.

Security Objective	SFR	Rationale
SO.LOCKSTATE	FTA_SSL.1	The TSF shall lock an interactive session after [a specified time has been reached or a user physically locks the interactive session] by: c) Invoking the locking screensaver where the user authentication is required to log in. The TSF shall require the following events to occur prior to unlocking the session: b) User must authenticate itself to unlock the session.
	FTA_SSL.2	The TSF shall allow user-initiated locking of a user's own interactive session by: Pressing ctrl-alt-del initiate the session lock. User must authenticate itself to unlock the session.
SO.TRANS_LOG	FIA_AFL.1	The TSF shall detect when a specified number set by the administrator unsuccessful authentication attempts occur related to a user logging into a computer that is protected by the TOE.
	FAU_GEN.1	The TSF will generate a record in the audit log on each auditable event.
	FAU_SAR.1	An authorized user must be able to read and interpret audit information.
	FAU_STG.3	Audit mechanism has to prevent loss of audit records.
SO.USER_I&A	FIA_UID.2	FIA_UID.2 provides that identification (by selection of user ID from the login form) must be done prior to any other TSF actions, such as encryption / decryption of protected data.
	FIA_UAU.2	FIA_UAU.2 provides that the user cannot perform actions such as encryption / decryption of protected data (other than selection of user ID from the login form) prior to authentication of the user's identity.

Security Objective	SFR	Rationale
	FIA_UAU.7	FIA_UAU.7 provides that the authentication feedback to the user be limited, allowing specifically that the characters of the user authentication string could be represented by asterisks (*).
	FTA_TRP.1	Upon initial boot of computer TOE sets trusted communication path between itself and local user.
SO.MANAGE	FAU_SAA.1	The TOE is able to set rules which provide in audit log indication of potential security violation.
	FIA_ATD.1	The TOE maintains lists of security attributes for individual users.
	FMT_MOF.1	The ability to change behavior of any security function inside TSC is restricted to authorized administrators.
	FMT_MSA.1 FMT_MSA.2 FMT_MSA.3	The TSF restrict ability to change security attribute to authorized administrators, ensuring secure values of attributes and providing default values.
	FMT_MTD.1	The TSF restrict ability to change security objects (such as audit records, privileges, keys, authentication data and mechanisms and handling of authentication failure) to authorized administrators (or users if the latter modify their own authentication data).
	FMT_MTD.2	The TSF restricts specification of unsuccessful authentication attempts threshold to authorized administrators and takes corresponding action if threshold is reached or exceeded.
	FMT_REV.1	The TOE restricts the ability to revoke user's security attributes to authorized administrators and enforces revocation on next user's login.
	FMT_SAE.1	The TSF restricts the ability to specify expiration time for user's objects and attributes to authorized administrators and is able to lock them or force user for pre-defined action after expiration.

Security Objective	SFR	Rationale
	FMT_SMF.1	The TOE specifies the list of security management functions which TSF is capable performing of.
	FMT_SMR.1	The TOE defines and maintains User and Administrator roles and is able to associate users with them.
SO.RESID_INFO	FDP_RIP.1	Releasing any informational resource under its control, the TOE makes its content unavailable.
SO.SPASSWORD	FIA_SOS.1	The TOE allows administrator to enforce strong password for user via password rules.
SO.SEC_OPER	FPT_AMT.1 FPT_TST.1	The TOE performs a suite of tests on start-up to ensure it may secure maintain normal operations.
	FPT_RVM.1 FPT_SEP.1	The TOE protects its own execution space and doesn't allow others to bypass functions that enforce TSP within TSC.

The coverage of the above table against the SFRs satisfies the following properties:

- for every security objective of the TOE, there is at least one SFR that satisfies it
- for every SFR, there is at least one security objective of the TOE that it addresses
- for every security objective of the TOE, an informal argument as to why the identified SFRs are sufficient to meet it is provided

8.2.2 IT Environment Security Requirements Rationale

The table below gives the rationale for the IT environment SFRs against the security objectives of the IT environment. For each security objective of the IT environment there is an IT environment SFR that addresses the security objective in question, as well as an explanation why this SFR is sufficient.

Security Objective	SFR	Rationale
SO.TIME_SRC	FPT_STM.1	The TOE requires a reliable source of time to provide accurate records in audit log

Security Objective	SFR	Rationale
SO.MULTI_AUTH	FIA_UAU.5	Multi-factor user authentication should be available to strengthen access control provided by the TOE.

8.2.3 Security Assurance Requirements Rationale

This ST contains the assurance requirements from the CC EAL4 assurance package and is based on good rigorous commercial development practices. This ST has been developed for a generalized environment with a medium level of risk to the assets.

The TOE will be used to protect attractive information assets and it is assumed that possible attackers will have a medium level of expertise, resources and motivation—an attack potential of medium. The Security Objectives for the TOE were derived to resist attackers with these characteristics, and CC EAL4 was found to be sufficient to provide the assurance for the environment.

8.2.4 Requirements Dependencies Rationale

Functional Requirements dependency

Among the above IT Security Requirements, some have dependencies that are either included in 5.1 or are satisfied via hierarchical components of the same family as shown below:

IT Security Requirement	Dependencies	Remarks
FAU_ARP.1	FAU_SAA.1	Included
FAU_GEN.1	FPT_STM.1	Included as requirement for IT environment
FAU_GEN.2	FAU_GEN.1	Included
	FIA_UID.1	Satisfied via FIA_UID.2
FAU_SAA.1	FPT_GEN.1	Included
FAU_SAR.1	FPT_GEN.1	Included
FAU_SAR.2	FPT_GEN.1	Included
FAU_STG.1	FPT_GEN.1	Included
FAU_STG.3	FPT_STG.1	Included
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Included
FCS_CKM.3	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Included
FCS_CKM.4	FCS_CKM.1	Included
	FMT_MSA.2	Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Included

IT Security Requirement	Dependencies	Remarks
FDP_ACC.2	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Satisfied via FDP_ACC.2
	FMT_MSA.3	Included
FIA_AFL.1	FIA_UAU.1	Satisfied via FIA_UAU.2
FIA_UAU.2	FIA_UID.1	Satisfied via FIA_UID.2
FIA_UAU.7	FIA_UAU.1	Satisfied via FIA_UAU.2
FMT_MOF.1	FMT_SMF.1	Included
	FMT_SMR.1	Included
FMT_MSA.1	FDP_ACC.1	Satisfied via FDP_ACC.2
	FMT_SMF.1	Included
	FMT_SMR.1	Included
FMT_MSA.2	ADV_SPM.1	Included
	FDP_ACC.1	Satisfied via FDP_ACC.2
	FMT_MSA.1	Included
	FMT_SMR.1	Included
FMT_MSA.3	FMT_MSA.1	Included
	FMT_SMR.1	Included
FMT_MTD.1	FMT_SMF.1	Included
	FMT_SMR.1	Included
FMT_MTD.2	FMT_MTD.1	Included
	FMT_SMR.1	Included
FMT_REV.1	FMT_SMR.1	Included
FMT_SAE.1	FMT_SMR.1	Included
	FPT_STM.1	Included as requirement for IT environment
FMT_SMR.1	FIA_UID.1	Satisfied via FIA_UID.2
FPT_FLS.1	ADV_SPM.1	Included
FPT_RCV.2	AGD_ADM.1	Included
	ADV_SPM.1	Included
FPT_TST.1	FPT_AMT.1	Included
FRU_FLT.1	FPT_FLS.1	Included
FPT_FLS.1	ADV_SPM.1	Included
FTA_SSL.1	FIA_UAU.1	Satisfied via FIA_UAU.2
FTA_SSL.2	FIA_UAU.1	Satisfied via FIA_UAU.2

Assurance Requirements dependency

Among the above Assurance Requirements, some have dependencies that are either included in 5.2 directly or through higher level family components as shown below:

Assurance Measure	Dependencies	Remarks
ACM_AUT.1	ACM_CAP.3	Included via ACM_CAP.4
ACM_CAP.4	ACM_SCP.1	Included via ACM_SCP.2
	ALC_DVS.1	Included
ACM_SCP.2	ACM_CAP.3	Included via ACM_CAP.4
ADO_DEL.2	ACM_CAP.3	Included via ACM_CAP.4
ADO_IGS.1	AGD_ADM.1	Included
ADO_DEL.2	ACM_CAP.3	Included via ACM_CAP.4

Assurance Measure	Dependencies	Remarks
ADV_FSP.2	ADV_RCR.1	Included
ADV_HLD.2	ADV_FSP.1	Included via ADV_FSP.2
	ADV_RCR.1	Included
ADV_IMP.1	ADV_LLD.1	Included
	ADV_RCR.1	Included
	ALC_TAT.1	Included
ADV_LLD.1	ADV_HLD.2	Included
	ADV_RCR.1	Included
ADV_SPM.1	ADV_FSP.1	Included via ADV_FSP.2
AGD_ADM.1	ADV_FSP.1	Included via ADV_FSP.2
ADV_USR.1	ADV_FSP.1	Included via ADV_FSP.2
ALC_TAT.1	ADV_IMP.1	Included
ATE_COV.2	ADV_FSP.1	Included via ADV_FSP.2
	ATE_FUN.1	Included
ATE_DPT.1	ADV_HLD.1	Included via ADV_HLD.2
	ATE_FUN.1	Included
ATE_IND.2	ADV_FSP.1	Included via ADV_FSP.2
	AGD_ADM.1	Included
	AGD_USR.1	Included
	ATE_FUN.1	Included
AVA_MSU.2	ADO_IGS.1	Included
	ADV_FSP.1	Included via ADV_FSP.2
	AGD_ADM.1	Included
	AGD_USR.1	Included
AVA_SOF.1	ADV_FSP.1	Included via ADV_FSP.2
	ADV_HLD.1	Included via ADV_HLD.2
AVA_VLA.2	ADV_FSP.1	Included via ADV_FSP.2
	ADV_HLD.1	Included via ADV_HLD.2
	ADV_IMP.1	Included
	ADV_LLD.1	Included
	AGD_ADM.1	Included
	AGD_USR.1	Included

8.2.5 Functional Claims Rationale

The selected functionality for this ST is consistent with and appropriate for the security objectives for the TOE. There are 4 main categories of security service that the TOE provides:

- User Identification and Authentication must precede all other access to protected information, providing binding between the user and the symmetric key used in read and write access to protected information stores;
- All access to the protected information is through decryption using specified algorithms and key lengths that are of appropriate strength for business, financial and personal private data under a broad class of applications;
- Restoration of a partially encrypted disk that has been created when a disk encryption process is interrupted either through power failure or operating system fault or interruption through contention for resources is automated.

These security services embody the security objectives of the TOE and are consistent with the level of capability and motivation that a threat agent would be expected to possess, given the assumptions regarding data sensitivity of information assets and sophistication of threat agent. Elimination of all potential threat agents clearly requires environmental support, procedural security and training. The latter safeguards are complementary security objectives that the environment is expected to supplement the TOE functional properties with in order to obtain an overall acceptable level of risk. They do not constitute weaknesses or omissions in the TOE, as the majority of the environmental security objectives are beyond the scope of any conceivable software solution. In addition, not all may represent serious risk to the average system in which the TOE is deployed.

8.2.6 Strength of Function Rationale

The TOE minimum strength of function level of SOF-medium was chosen to be consistent with the risk to assets defined within the TOE. The explicit strength of function claimed for the password authentication mechanism specified in FIA_UAU.2 and FIA_UID.2 is adequate to the probability of guessing a fixed password. The password selection policy enforced through the TOE password rules supports FIA_SOS.1 requirements to claim SoF-medium strength level.

The SOF-medium strength level is sufficient to meet the security objectives of the TOE given the security environment described in the ST.

8.2.7 TOE Consistency Rationale

The ST includes no instance of a requirement that contradicts another requirement in the ST. In instances where different requirements apply to the same events or types of data, the requirements and the operations performed within the requirements do not contradict each other, but provide supporting functionality ensuring that the TOE is internally consistent.

The combination of several different supporting security functions, and the inclusion of all dependencies as illustrated in 8.2.1 and 8.2.3, ensures that together the selected requirements form a mutually supportive whole.

This claim is also supported by the following:

1) requirements are suitably mapped to security objectives and TOE security functions as demonstrated above
2) architectural requirements FPT_RVM.1 (Non-bypassability of TSP) and FPT_SEP.1 (TOE domain separation) provide protection of the TSF
3) TOE Security Policy are covered by corresponding security function
4) audit requirements prevent undetected attacks of other security functional requirements
5) security management requirements are included to ensure proper configuration and control of other security functional requirements.

8.3 TOE Summary Specification Rationale

8.3.1 IT Security Functions Rationale (SFRs)

The TOE IT Security Functions are listed with cross-references to the SFRs, described in section 6, that are provided by the defined IT Security Function. Specifications of IT Security Functions are provided in section 6.

Security Function	Functional Requirement	Rationale
Access Control	FDP_ACC.2 FDP_ACF.1 FDP_RIP.1	It is required that complete access control is in place for all operations between subjects, on objects in the TOE. This is covered in the functionality of the TOE in that all file objects on the protected disk / partition are successfully read from or written to only if the user has authenticated him/her prior to the specified read/write operations and related transactions in the login session. If the authentication is unsuccessful, the correct key will not be initialized for the session and all specified operations will fail. Once a key is deleted from the user key file it is not available for that user any more. The information content of the key file removed from the computer is not available either.
User Identification and Authentication	FIA_AFL.1 FIA_ATD.1 FIA_SOS.1 FIA_UAU.2 FIA_UAU.7 FIA_UID.2	It is required that the user be successfully authenticated before any action, such as read or write to/from the protected disk, be permitted. This is covered in the functionality of the TOE by prompting the user for his/her secret password to authenticate the user and retrieve his/her secret key. It is required that only asterisks (*) be presented to the user while authentication is in progress. This is covered in the functionality of the TOE, i.e., while the user is entering his/her password, only a single asterisk appears for each character entered.

Security Function	Functional Requirement	Rationale
Security Management	FMT_MOF.1 FMT_MSA.1 FMT_MSA.2 FMT_MSA.3 FMT_MTD.1 FMT_MTD.2 FMT_REV.1 FMT_SAE.1 FMT_SMF.1 FMT_SMR.1	Various security policies could be enforced by the TOE through configuration of security objects and their attributes. Suitable interfaces are supported by the TOE for this purpose. There is a number of restrictions on access to these interfaces as well as the functionality involved and security data affected. The TOE provides the functionality above through maintaining different roles, which differ in rights and privileges.
Security Audit	FAU_ARP.1 FAU_GEN.1 FAU_GEN.2 FAU_SAA.1 FAU_SAR.1 FAU_SAR.2 FAU_STG.1 FAU_STG.3	The TOE requires that all user transaction related together with system events that might affect its operation were monitored. The result of the monitoring is kept in audit log in the form that allows unambiguous identification of users caused the record as well as the details of the event or transaction performed. Only authorized staff has access to audit record. TOE prevents the records from any modifications by user and also from loss due to audit limit exceeding.
Session Control	FTP_TRP.1 FTA_SSL.1 FTA_SSL.2 FTA_TAH.1	The TOE control interaction with local user from the beginning of operations by setting trusted communication path and later providing automated or manual lock of current user's session. New authentication procedure precedes resuming session activity. The history of login attempt is available for further analysis.
Cryptographic Support	FCS_CKM.1 FCS_CKM.3 FCS.CKM.4 FCS_COP.1	It is required that the TOE performs cryptographic operations in accordance with a specified algorithm, cryptographic key size and standard. The functionality of the TOE includes algorithms for encryption, secure hash and message authentication, that are defined by standards and employ key lengths appropriate to the algorithms and standards referenced. The TOE performs cryptographic operations in accordance with the following standards: FIPS 180-2, 197, 198.

Security Function	Functional Requirement	Rationale
Protection of TSF	FPT_AMT.1 FPT_FLS.1 FPT_RCV.2 FPT_RVM.1 FPT_SEP.1 FPT_TST.1	The TOE controls its own integrity performing a suite of tests during each start-up. This prevents against attempt to modify TOE or its environment in order to bypass or weak security of TOE operational environment. It is required that the TOE have the capability to recover from an interrupted disk encryption operation (i.e., where only partial encryption of the disk has been achieved) where the interruption is due to loss of power, unrelated process running in the OS or system fault requiring reboot. This functionality covers this requirement through the automated recovery operation of the TOE. TOE key management denies access to cryptographic keys and encrypted data unless the user is successfully authenticated and has appropriate privileges. The code is separated while running in pre-boot, kernel and user modes, where first two are not accessible from the latter one
Fault Tolerance	FRU_FLT.1	It is required that the TOE have the capability to recover from an interrupted disk encryption operation (i.e., where only partial encryption of the disk has been achieved) where the interruption is due to loss of power, unrelated process running in the OS or system fault requiring reboot. This functionality covers this requirement through the automated recovery operation of the TOE.

The combined aggregate of the TOE security functions satisfy the set of identified TOE SFRs as shown above. Given that the cryptographic power of the TOE (in terms of algorithm choice and key size) is sufficient to protect information assets within the requirements of the organization / environment, then it can be concluded that the security functionality of the TOE is effective in applying that cryptographic protection to a restricted user-selected set of information assets. It is clear that the problems of protecting residual objects and temporary application-created objects containing sensitive information is effectively solved through the direction of the TOE to logical drives and partitions, rather than to individual files. The embedding of cryptographic services in the device driver layer of the environment ensures application transparency. Certain utilities such as compression software must be selectively chosen. These criteria are identified in the user documentation. Provided the configuration and maintenance of the TOE is carried out in a secure way, following vendor recommendations, the TOE security functional claims are valid.

8.3.2 Assurance Measures Rationale

The compliance of the TOE with the required assurance measures is established in the table below.

Assurance Measures	Assurance Requirement	Compliance
Process Assurance	ACM_AUT.1 ACM_CAP.4 ACM_SCP.2 ALC_DVS.1 ALC_LCD.1 ALC_TAT.1	Process assurance measures cover life cycle of the TOE controlling its development through a number of regulation documents and procedures. Those procedures provide configuration management for the TOE.
Delivery and Guidance	ADO_DEL.2 ADO_IGS.1 AGD_ADM.1 AGD_USR.1	It is required that procedures for secure delivery, installation and start-up were defined for the TOE and the guidance documents contained all necessary information.
Design Documentation	ADV_FSP.2 ADV_HLD.2 ADV_LLD.1 ADV_IMP.1 ADV_SPM.1 ADV_RCR.1	Design documentation of various levels of details demonstrates methodology and architectural decisions for refinement of initial functional requirements into final implementation.
Tests	ATE_COV.2 ATE_DPT.1 ATE_FUN.1 ATE_IND.2	Testing procedures follow the development process and verify TOE operability according its functional requirements and design.
Vulnerability Assessment	AVA_MSU.2 AVA_SOF.1 AVA_VLA.2	Vulnerability analysis is provided for the TOE to figure out where vulnerability might exist for its secure operation and estimate how they are addressed by TOE and corresponding documentation.