Encryption Plus® Hard Disk 7.0 Security Target

Common Criteria EAL1
Version 1.02, March 24, 2003

PC Guardian
1133 East Francisco Boulevard
San Rafael CA 94901 USA
http://www.pcguardian.com
+1 415-459-0190

Contents

1. ST Introduction

1.1. ST Identification

ST Title: Security Target Version 1.02 for Encryption Plus Hard Disk 7.0
Date: March 24, 2003
TOE Title: Encryption Plus Hard Disk 7.0
Vendor: PC Guardian, San Rafael, California

1.2. ST Overview

The Encryption Plus Hard Disk 7.0 software package (hereinafter referred to as EP Hard Disk) is a hard disk encryption system that encrypts entire disks or partitions at the disk driver level so that normal applications can use the EP Hard Disk confidentiality services transparently. EP Hard Disk includes features for site installation, administration, and recovery from lost passwords.

1.3. CC Conformance

This Security Target (ST) is defined with reference to Common Criteria for Information Technology Security Evaluation, Version 2.1, August 1999, ISO/IEC 15408 (hereinafter referred to in abbreviated form as CC).

This ST document conforms to CC part 2.

The TOE conforms to CC part 3.

The TOE will be evaluated to Evaluation Assurance Level 1 (EAL1).

This document presents the evaluation evidence for the claim that EP Hard Disk conforms to Common Criteria Evaluation Assurance Level 1.

EAL1 was chosen as appropriate to the security needs of customers the TOE is used by and will be marketed to.

2. TOE Description

EP Hard Disk is a hard disk encryption system that encrypts entire disks or partitions at the disk driver level so that normal applications can use the EP Hard Disk confidentiality services transparently. EP Hard Disk is available for Windows 2000, XP, and NT versions of the Microsoft Windows family of operating systems.

The following table shows the application components, main user-visible functions within those components, and the user role expected to use each function. This table is intended to clarify the relationships between the components and functions. The component names, function names, and role names used in the table are used throughout this document.

Application Component	Application Function	Intended User
Administrator Program	Administrator Logon function	EP Hard Disk Administrator
	User Program Setup wizard	
	Configuration Update function	
User Program	Initial Encryption	User
	Initial Encryption (Pre-installed)	EP Hard Disk Administrator
	On-the-Fly Encryption function	User
	User Logon function	
	Authenti-Check Logon function	
	Access Recovery function	
	Full Decryption function	User
		Corporate Administrator
		Local Administrator
	User Program Admin Logon function	Corporate Administrator
		Local Administrator
One-Time Password Program	Access Recovery function	Corporate Administrator
		Local Administrator
Recover Program	Hard Disk Repair function	User
		Corporate Administrator
		Local Administrator

The data written to and read from the partition or disk are respectively encrypted and decrypted on-the-fly as required, driven by operating system use of the storage device. The encryption algorithm used is the Advanced Encryption Standard (AES) **[AES]** in Cipher Block Chaining (CBC) mode **[AES-MODES]** with 256-bit keys. The Disk Key, which is used to encrypt the data on the disk, is randomly generated and stored encrypted under the Disk Key Encryption key (Disk KEK), which is derived from the user name and password using the key derivation function PBKDF2 defined in **[PBKDF2].**

It is recommended that all disk partitions be encrypted with EP Hard Disk to minimize the risk of swap files and other application and operating system generated temporary files being stored in plaintext on an unprotected disk partition.

Another source of risk — this one outside the scope of the product — is the use of hibernation modes common on laptops, where the state of the machine's memory is stored onto disk, typically in a separate disk partition outside the control of EP Hard Disk. It is recommended that these features be disabled in order to avoid the risk that a stolen laptop or machine could have user data recovered from the hibernation partition. Stand by mode — where the machine's state is retained in RAM but other components are powered down — is supported, though the operating system should be configured to require a password on resume from this state.

EP Hard Disk includes administrative functions and roles to facilitate use in a corporate environment. There are three classes of administrator: EP Hard Disk Administrator, Corporate Administrator, and Local Administrator. To authenticate themselves to EP Hard Disk, administrators have their own passwords. The EP Hard Disk Administrator is the master administrator and delegates tasks to Corporate and Local Administrators. The EP Hard Disk Administrator can assume the role of any Corporate or Local Administrator by entering the respective administrator's user name and password. The EP Hard Disk Administrator also creates the user installation package using the User Program Setup wizard, and creates configuration update messages. (The Configuration Update function is described further below.)

Local Administrators are assigned a domain of control (for example, a department within the company) by the EP Hard Disk Administrator and are only able to fulfill the Access Recovery and User Program Admin Logon functions within their domain of control; Corporate Administrators, on the other hand, can access the entire domain of control covered by the installation. All users in an installation are under the administrative control of the Corporate Administrator; each user is under the administrative control of one of the Local Administrators.

(Note: In principle, a company could have multiple installations, each with a separate EP Hard Disk Administrator and Corporate Administrator with control within that domain. In a small site, the Local Administrator role may not be used, and those tasks normally carried out by a Local Administrator are instead carried out by the sole Corporate Administrator. Similarly, in a small site, the EP Hard Disk Administrator and Corporate Administrator roles could be fulfilled by the same person.)

EP Hard Disk also includes an access recovery procedure that allows designated administrators to remotely assist users in regaining access to their data when they forget their passwords. The administrators use the Access Recovery function of the One-Time Password Program to do this. The Access Recovery function recovers the Disk Key the disk is encrypted with: this allows the user to regain access to their data. Once access is regained, EP Hard Disk allows the user to choose a new password. The messages exchanged between the user and the administrator during the recovery procedure are compact so that the messages can be communicated verbally (for example, over a telephone). The One-Time Password Program does not require the administrator to log on. The administrator private key is stored in the One-Time Password Program installation. The administrator must retain good physical security of the machine the One-Time Password Program is installed on; the machine should preferably not have a network connection, or at minimum should have good network security measures.

In addition, the Corporate Administrator and Local Administrator are able to log on to the User Program and gain access to user data without user assistance (given physical access to the machine).

Terminology note — the term *recovery* is used in three different contexts in this document:

1. The low-level cryptography-related use in the term *key recovery;*

2. The application-related use in the term *access recovery*, which assists users in regaining access to their data when they forget their password (this is a function of the One-Time Password Program); and

3. The reliability-related use in the term *Recover Program,* which is a hard drive repair tool.

The access recovery procedure technically works as follows. The Disk Key is encrypted under the Elliptic Curve Key Recovery Key (ECKRK) with AES in CBC mode. The ECKRK is derived by first negotiating a key with Elliptic Curve Diffie-Hellman (ECDH) encryption using the Corporate Administrator's and the Local Administrator's public ECDH keys, and then deriving the ECKRK from the negotiated key and the user name with the KDF2 key derivation function. The ECKRK-encrypted blocks are called recovery blocks, and there are two recovery blocks: the Corporate Administrator recovery block and the Local Administrator recovery block. The AES-encrypted Disk Key is also stored with the recovery block. During the access recovery protocol, the user — with the assistance of an administrator — identifies and authenticates himself or herself to the administrator, and then transfers part of an administrator recovery block to the corresponding administrator. The administrator uses the recovery block part to negotiate a shared key, then derives the ECKRK from the shared key and the authenticated user's user name using the KDF2 key derivation function, and transfers the ECKRK back to the user. The user recovers the Disk Key by decrypting the Disk Key with AES using the ECKRK. The Disk Key allows the user to access his or her files, and the user can then choose a new password. The recovery blocks are over-written with new recovery blocks, so that recovery messages captured by a threat agent eavesdropping on the recovery messages do not help the threat agent to subsequently recover user data if he or she were to gain physical access to the user machine.

The same underlying key recovery cryptographic mechanism is used to allow the administrator to gain access to user data (given physical access to the user machine).

As a final precaution to ensure availability of the Access Recovery and User Program Admin Logon functions in the event that EP Hard Disk administrators leave the company without telling the company their passwords, there is a backup procedure for the Administrator Database Key that is completed under the auspices of the EP Hard Disk Administrator.

EP Hard Disk contains an alternative key recovery mechanism called Authenti-Check® that enables the user to recover their Disk Key without assistance from an administrator. The user is asked to provide a list of questions and answers during setup of the User Program. The Authenti-Check Key Recovery Key (KRK) is derived from the answers to the questions provided by the user. The Authenti-Check KRK is used to encrypt the Disk Key.

Once the User Program setup is complete, the user then invokes the Initial Encryption process. This process first prompts the user for a password and then encrypts the disk. The EP Hard Disk Administrator has a few options relating to Initial Encryption that provide security and reliability vs. speed trade-offs as the initial encryption of a large disk can be time-consuming. One option is to encrypt only used space: unused space will be encrypted on-the-fly as it is populated with user data during later use. This trades off some security for speed of initial encryption. To use this feature safely, the user should be confident that there is no sensitive user data on unused portions of the disk at time of encryption. The other option is to disable power-loss protection during initial encryption. This trades off some reliability for speed of initial encryption. To safely disable this feature, the user should either have no user data on the machine or have backups before starting, and should ensure a reliable source of power with sufficient capacity to complete the operation.

To facilitate pre-installation of EP Hard Disk on laptops and workstations by trusted administrators, a feature is available where the administrator can give a default user name and password during initial encryption. This user name and password is used to signify that the machine is in a pre-installed state. The configuration and use of Authenti-Check and Access Recovery functions are disabled until the user chooses his or her own password. The user is reminded at each application start to choose a password, until a password is provided. If applicable, Authenti-Check configuration proceeds. The application is then fully set up: the Authenti-Check and Access Recovery functions are made available.

Users can change their passwords at any time if the EP Hard Disk Administrator has allowed them to make the change. If Corporate and Local Administrators wish to have their passwords changed, there is a password update feature available to the EP Hard Disk Administrator in the Administrator Program. This feature creates a signed password update that can be installed on existing installations of the User Program. The User Program then updates the recovery blocks with the new public keys corresponding to the new administrator passwords.

There are a number of configurable User Program options related to security, such as messages to display at various points in the EP Hard Disk dialogs (for example, phone numbers or methods of contacting the administrators) and options relating to the number of incorrect entries allowed during password entry. The EP Hard Disk Administrator configures these options into the User Program setup files, which are then installed on user workstations. There is support for automated network installations, for example via network logon scripts. Configuration changes can also be made to installations of the User Program by the EP Hard Disk Administrator, using a signed configuration change package. Both configuration changes and administrator password changes can be automatically updated on the installations of the User Program using, for example, a network logon script.

Configuration changes are signed with the current EP Hard Disk Administrator's Elliptic

Curve Digital Signature Algorithm (ECDSA) signature key. Administrator password updates are signed with the old EP Hard Disk Administrator ECDSA and the new EP Hard Disk Administrator ECDSA key, signifying transfer of authority from the old to the new key. The old ECDSA key will always be available even though the EP Hard Disk Administrator password may have been lost since the ECDSA key and the ECDH key are stored in the Administrator Database.

The signed update message includes all signed ECDSA public key updates from the installation time, to ensure that a user who is offline for some time and misses some of the updates can verify signatures on updates in a chain of signatures with the previous key on the replacement key leading to the current ECDSA public key.

The EP Hard Disk Administrator ECDSA key scalars and the Corporate and Local Administrator ECDH private key scalars are computed from the corresponding administrator passwords. The EC curve and public parameters used are Koblitz curves of size 233 bits from ECDSA **[FIPS-DSA]**. The ECDH private key is derived from the administrator password.

There is a decryption feature to allow users to convert an encrypted partition back into plaintext (Full Decryption). This feature can be disabled by the EP Hard Disk Administrator at setup or by one of the signed update messages described above. This feature is provided to assist in implementing security policies that call for all user data to be encrypted.

As a convenience to the user, a Single Sign-On feature is provided. The logon to the User Program is displayed before the Windows logon window. If the Single Sign-On option is selected, EP Hard Disk manages authentication to Windows so that the Windows logon window will not be displayed. EP Hard Disk stores the Windows logon name and password encrypted inside the encrypted partition and supplies them to the Windows logon in order for Sign Sign-On to function.

Another related user convenience feature is a password synchronization option, which updates the EP Hard Disk password when the Windows password changes so that the two passwords will continually match.

Finally, there is the Recover Program, which can be used in scenarios where the hard disk suffers data loss. It reconstructs EP Hard Disk related data from redundant copies that are kept by the application. After the Recover Program has completed — presuming that the data loss was not too extensive — the normal logon and access functions are available to the user.

2.1. Conformance

In this document where conformance to security standards is claimed, conformance is determined by the developer.

3. TOE Security Environment

3.1. Assumptions

The following is a list of assumptions made of the security environment the TOE operates in:

A.TRU_ADM	Personnel fulfilling administrative roles in the TOE's operation are trustworthy. If EP Hard Disk administrators have privileges allowing them to gain access to user data, it is assumed that they are trustworthy and do not attempt to make unauthorized disclosures of confidential data or disclose administrator passwords allowing recovery of confidential data.
A.SHO_SUR	It is not possible for the users or administrators to have their passwords or Authenti-Check answers compromised by a threat agent observing them typing it in ("shoulder surfing"). This includes threat agents in the immediate vicinity of the user or administrator as well as use of surveillance equipment such as telescope where the user's keyboard is observable at some distance, and hidden video cameras. It is also assumed that closed circuit TV if any is operated by the company is not positioned so as to make keyboard entry viewable, or if it is viewable, that the trusted personnel include personnel operating internal CCTV systems.
A.PHY_CTL	The computer the User Program is installed on should not fall under temporary and undetected physical control of a threat agent. Appropriate physical security measures and physical security policies are in place to manage risk of this event occurring.
A.REC_PHY	The computer the One-Time Password Program is installed on should not fall under the physical control of a threat agent.
A.TRU_SW.	The software environment runs only trusted software that has been approved by the security officer. This also presumes appropriate protections against malicious installation of non-approved software such as viruses and Trojan horses by the appropriate deployment of firewalls, bastion hosts, and anti-virus software as well as potentially use of the distinction between administrator account and user account on Windows NT where Windows NT is used to further reduce risk by ensuring that users do not have privileges to install software.
A.MOD_SW.	If the risk of undetected software modifications is insufficiently mitigated by physical security policy and measures practical in the

	environment, it is assumed appropriate technical measures are taken to reduce risk of or detect software modification.
A.MOD_HW.	If the risk of undetected hardware modifications is insufficiently mitigated by physical security policy and measures practical in the environment, it is assumed that the threat agent able to gain temporary and undetected physical access to the machine has insufficient expertise and/or resources to install a hardware keyboard sniffer or other hardware modification to allow passwords or data to be recovered.
A.BAK_SEC	Backups taken of the user's data are assumed to be separately encrypted or physically protected to ensure data security is not compromised via theft of or unauthorized access to backed up information.
A.BAK_AVA	Regular and complete backups are assumed to be taken so that user data can be recovered in the event that a threat agent gains temporary physical access to the machine and deletes or otherwise damages the integrity of the data for example by formatting the disk after rebooting from read only media. Similarly good backups ensure that data remains available even in the event that the threat agent steals and/or physically destroys the equipment in an attempt to deny availability.
A.BAK_DB.	The Administrator Database is assumed to be adequately backed up to ensure availability of functions of the Administrator Program.
A.NET_ACC	If the computer is connected to a network, it is assumed that either file sharing and other network services offering remote access to data stored on the computer are disabled, or that appropriate authentication and confidentiality services are used in combination with those services and that the authenticated remote users are considered to be within the domain of authorized users.
A.NET_SCR	If network logon scripts or other mechanisms involving automatic execution of remotely downloaded software are used, it is assumed that this software is trusted and approved by the security officer and that either this feature is only used on a trusted physically controlled network, or that appropriate security and authentication mechanisms are used to prevent a threat agent modifying or inserting additional software to be run by this mechanism.
A.HIB_STO	Hibernation features common on laptops can result in portions of user data in memory being used by applications, or cached in

	operating system disk caches, being stored to a separate disk partition not under the control of EP Hard Disk. It is assumed that such features are disabled.
A.USR_ATH	When the administrator assists the user in recovering access to their data with the One-Time Password Program, the administrator must assure himself or herself of the user's identity. This is to prevent a threat agent — who has stolen or gained unattended access to the user's machine while it is not logged on — from using the access recovery procedure by pretending to be the user. The administrator is assumed to use some reliable and secure method to authenticate users.
A.NO_UAT	It is assumed that the EP Hard Disk software is not left unattended while encrypted partitions are mounted. This is to avoid a threat agent using the opportunity of temporary access to the machine to make unauthorized copies of user data, or to make undetected configuration changes to put the system in an insecure state so that user data can be later copied at leisure.
A.INI_SEC	While the machine is in the pre-installed state with the default password, before the user has changed the password, it is assumed that adequate physical security precautions are taken to ensure that a threat agent is not able to use the default user password to obtain the encryption keys.
A.NT_PWD	If the Windows password synchronization option is used on a network connected machine that is using remote password management via a Windows NT domain server, security of the TOE encrypted data will depend on the security of the password management protocol used by Windows NT, and the security of the configuration of the server.
A.USD_SPC	When the option to perform initial encryption of used space only is selected, data that was used but deleted, or left on currently unused areas of the disk when data was migrated due to de-fragmentation, will not be encrypted. It is assumed that this option will only be used where there is, at the time of encryption, no sensitive information on the disk.
A.PWR_LOS	When the option to disable power-loss recovery during initial encryption is used it is assumed that no user data (or no non-backed up user data) is on the disk, and that the machine is connected to a reliable source of power with sufficient capacity to complete the operation.

3.2. Threats

The following is the list of threats that may target the assets the TOE is protecting. The asset under attack in all of the following attacks is user data stored in encrypted form by EP Hard Disk. Some of these attacks are indirect in so far as a password or key that protects the data is the immediate information under attack. The motivation of the threat agent is either to gain access to the user the user data, or to deny the user access to their own data (known as a "denial of service" attack).

T.PAS_LOS	The user may forget their password, making data unavailable. There is no third party threat agent with this threat; rather a memory lapse on the part of an authorized user presents the threat that the user will lose access to their data.
T.DSK_COR	The disk may become corrupted due to mechanical failure or unclean operating system shutdown due to power interruption. There is no third party threat agent with this threat, though the mechanical failure could potentially be intentionally induced by a threat agent with the intent of denying the user access to his data.
T.DAT_SEC	A threat agent who has exploited an opportunity to gain physical access to the machine may try to examine data stored on disk to find user data that is stored in protected partitions.
T.USR_LOG	A threat agent who has exploited an opportunity to gain physical access to the machine may try to abuse the User Program logon mechanism to gain access to the user's data.
T.UAD_LOG	A threat agent who has exploited an opportunity to gain physical access to the machine the User Program is installed on may try to execute the User Program Admin Logon protocol in an attempt to gain access to the user's data.
T.ADM_LOG	A threat agent who has exploited an opportunity to gain physical access to the machine the Administrator Program is installed on may try to abuse the Administrator Program logon mechanism to gain access to the functions of the Administrator Program.
T.REC_USR	The threat agent may be another EP Hard Disk user who has stolen or otherwise gained physical access to the target user's machine. The threat agent may try to execute the access recovery procedure authenticating as himself in an attempt to gain access to the target user's data.
T.REC_EAV	The threat agent may eavesdrop on the telephone or other communications between the user and the administrator to capture the messages exchanged during the access recovery procedure. The

	threat agent will then after the fact attempt to steal or otherwise gain detectable access to the computer and try to use the recovery information eavesdropped to gain access to the user's data by using it to execute the recovery procedure.
T.ATK_LOG	A threat agent who has exploited an opportunity to gain physical access to the machine may try to gain access to the machine via the Authenti-Check logon function with the aim of gaining unauthorized access to user data.
T.UPD_MOD	The threat agent may try to modify configuration and password updates the User Program receives from the administrators. If the threat agent could modify the administrator password update, it could replace the administrator's new EC public key in the update message and hence be able itself to execute the User Program Admin Logon protocol on the user machine if he could gain physical access. Configuration option updates are also of relevance to security, in that if the threat agent could modify contact information in the application he may be able to more easily socially engineer passwords or other sensitive information from the users who may then incorrectly assume the threat agent is a trusted company administrator. The aim of these attacks is to gain unauthorized access to user data.
T.ADM_CFG	The administrator may unintentionally select insecure configuration parameters or insecure default configuration parameters for the user. The risk if insecure configuration parameters are selected is that a threat agent could attempt to gain access to user data with fewer restrictions than intended by the administrator.
T.USR_CFG	The user may unintentionally select insecure configuration parameters, reducing the security of the TOE. The user may try to select values that the EP Hard Disk Administrator considers inappropriate for the environment the installation is used in. The risk if insecure configuration parameters are selected is that a threat agent could attempt to gain access to user data with fewer restrictions than intended by the user.
T.SW_BUG	The TOE may exhibit a software bug and fail to protect the user data.
T.DAT_LEK	If the user configures the software to have some encrypted and some unencrypted partitions the user may accidentally write data intended to be protected to an unprotected partition. Application

	software may write user data to unprotected partitions without the user's knowledge.
T.DB_SEC	If the Administrator Database contents were obtained by a threat agent, the threat agent could execute the User Program Admin Logon protocol on any installation of the User Program in the Corporate Administrator's domain of control and thereby gain unauthorized access to user data.
T.BAK_DBK	If the Administrator Database key were obtained by a threat agent who was also able to copy the Administrator Database, the threat agent could execute the User Program Admin Logon protocol on any installation of the User Program in the Corporate Administrator's domain of control, thereby gaining unauthorized access to user data. If the Administrator Database key were lost and the corresponding passwords forgotten, the Administrator Program functions would become unavailable. In this event availability of user data could be lost if the user forgets their password as the recovery function and admin logon would no longer be available.

3.3. Organizational Security Policies

Security objectives are derived from assumptions and threats only, so this section is left blank.

4. Security Objectives

4.1. Security Objectives for the TOE

SO.DAT_AVA	The TOE must ensure continued availability of user data in event that the user forgets his or her password. This security objective is to counter threat T.PAS_LOS.
SO.DSK_COR	The TOE must always be in a state that can be resumed or recovered into a secure and consistent state in event that the power is interrupted. This security objective is to counter the threat of disk corruption and loss of data availability described in threat T.DSK_COR.
SO.DAT_SEC	The TOE should encrypt user data on the disk so that a threat agent who does not have the password or key will be unable to gain access to the user data by directly analyzing data on the disk.
SO.USR_LOG	The TOE should provide a secure logon function where only authorized users are able to gain access to user data via the logon function.

SO.ADM_LOG	The TOE should provide a secure logon function where only the EP Hard Disk Administrator is able to gain access to the Administrator Program.
SO.UAD_LOG	The TOE should provide a logon function where only authorized administrators with administrative control over the domain of the machine are able to gain access to user data via the User Program Admin Logon function.
SO.REC_SEC	The TOE should allow the user with assistance from an administrator to regain access to his machine and set a new password after forgetting his password. Only the authorized user whose data is protected on the machine and an administrator with physical control of the One-Time Password Program should be able to successfully exercise the access recovery protocol. The threat agent is presumed to have access to the previous messages the user may have exchanged with the administrator by having eavesdropped on the exchange.
SO.ATK_LOG	The TOE should provide a secure Authenti-Check logon function where only authorized users can authenticate themselves.
SO.ATK_SEC	The Authenti-Check procedure allows a user to regain access to their machine and choose a new password if they forget their password. After the user has authenticated himself or herself with the Authenti-Check logon function as described under SO.ATK_LOG, a key is derived from the authentication process and this key used to regain access to the user data.
SO.UPD_ATH	The TOE should provide mechanisms to authenticate the administrator password and configuration update messages so that a threat agent cannot undetectably modify them.
SO.ADM_CFG	The TOE should restrict the configuration values the administrator can set to secure values.
SO.USR_CFG	The TOE should restrict the configuration options the user may select to secure values. Default secure values and ranges of restricted values should be stored in the application; the Corporate Administrator should be able to modify some of the default values and restricted ranges of values to suit the environment and the organization’s policies.
SO.SW_TST	The TOE should perform self-tests to verify correct operation of sensitive operations.

SO.ENC_ALL The TOE could encourage the user to encrypt all partitions to counter the threat of sensitive user data being accidentally written to unprotected partitions.

SO.DB_ENC The TOE should encrypt data in the Administrator Database to prevent unauthorized users reading the information stored in it.

SO.BAK_DBK The Administrator Database Key must be stored on removable media and the media stored in a physically secure location such as a safe.

4.2. Security Objectives for the Environment

4.2.1. IT Environmental Security Objectives

The following are security objectives for the IT environment the TOE operates in.

The Environmental Security Objectives below will not be used to evaluate the security of the TOE. The TOE has no programmatic dependencies on these security objectives, so these IT environmental security objectives are not further refined into Security Functional Requirements.

However, these security objectives they may be useful to prospective users of the TOE to evaluate any changes that it may be desirable to make to their environment to improve operational security of the TOE, and to evaluate the operational security implications on the TOE of any desired changes that are not addressed.

Related policy and personnel requirements on the environment are given in the section below on Non-IT Environmental Security Objectives.

SO.TRU_SW. The software environment must run only trusted software that has been approved by the security officer. Appropriate protections against malicious installation of non-approved software such as viruses and Trojan horses must be employed. Examples of protection systems include the appropriate deployment of firewalls, bastion hosts, and anti-virus software as well as potentially use of the distinction between administrator account and user account on Windows NT where Windows NT is used to further reduce risk of breach of policy by ensuring that users do not have privileges to install software.

SO.MOD_SW. If the risk of undetected software modifications on the machine the User Program is installed on are insufficiently mitigated by physical security policy (see SO.PHY_CTL below) and measures practical in the environment, appropriate technical measures should be taken to detect software modification. Measures include:

- Storing cryptographic checksums of data stored on the machine's disks on removable media, and using a removable media boot disk with integrity checking software to compare a checksum with the data on the disk

SO.MOD_HW. If the risk of undetected hardware modifications is insufficiently mitigated by physical security policy and measures practical in the environment, it is assumed that the threat agent able to gain temporary and undetected physical access to the machine has insufficient expertise and/or resources to install a hardware keyboard sniffer or other hardware modification to allow passwords or data to be recovered.

SO.BAK_SEC Backups taken of the user's data must be separately encrypted or physically protected to ensure data security is not compromised via theft or unauthorized access of backed up information.

SO.BAK_AVA Regular and complete backups must be taken so that user data can be recovered in the event that a threat agent gains temporary physical access to the machine and deletes or otherwise damages the integrity of the data for example by formatting the disk after rebooting from read only media. Similarly good backups ensure that data remains available even in the event that the threat agent steals and/or physically destroys the equipment in an attempt to deny availability.

SO.BAK_DB. Backups should be taken of the Administrator Database to ensure continued availability of the Administrator Program functions.

SO.NET_ACC If the computer is connected to a network, either file sharing and other network services offering remote access to data stored on the computer must be disabled, or appropriate authentication and confidentiality services must be used in combination with those services and the authenticated remote users must be considered to be within the domain of authorized users.

SO.NET_SCR If network logon scripts or other mechanisms involving automatic execution of remotely downloaded software are used, this software is considered trusted and only software approved by the security officer may be used. Also, either this feature must only used on a trusted physically controlled network, or appropriate security and authentication mechanisms must be used to prevent a threat agent from modifying or inserting additional software to be run by this mechanism.

SO.HIB_STO	Hibernation features common on laptops can result in portions of user data in memory being used by applications, or cached in operating system disk caches being stored to a separate disk partition not under the control of EP Hard Disk. Such features must be disabled.
SO.NO_UAT	The users of the TOE should not leave the software unattended in a logged in state. They should either log off before leaving the terminal, or employ the built in Windows screen saver or a third-party screen saver and configure that screen saver to require a password to disable.
SO.NT_PWD	If the Windows password synchronization option is used on a network connected machine that is using remote password management via a Windows NT domain server, security of the TOE encrypted data will depend on the security of the password management protocol used by Windows NT, and the security of the configuration of the server. Users of the TOE should evaluate the applicability of this risk and the security implications in their environment when deciding whether to use the Windows password synchronization option.
SO.USD_SPC	The user should only use the option to perform initial encryption of used space where there is no sensitive information on the disk at the time of encryption.
SO.PWR_LOS	When the option to disable power-loss recovery during initial encryption is used it is assumed that no user data (or no non-backed up user data) is on the disk, and that the machine is connected to a reliable source of power with sufficient capacity to complete the operation.

4.2.2. Non-IT Environmental Security Objectives

This section contains policy and personnel related security requirements of the environment the TOE operates in.

SO.TRU_ADM	Personnel fulfilling administrative roles in the TOE's operation must be trustworthy. If EP Hard Disk administrators have privileges allowing them to gain access to user data, they must be trustworthy not to attempt to make unauthorized disclosures of confidential data or of the administrator passwords allowing recovery of confidential data.
SO.SHO_SUR	It must not be possible for users or administrators to have their passwords compromised by a threat agent observing them typing it in ("shoulder surfing"). This includes threat agents in the

	immediate vicinity of the user or administrator as well as use of surveillance equipment such as telescope where the user's keyboard is observable at some distance, and hidden video cameras. If closed circuit TV is operated by the company it must either not be positioned to make keyboard entry viewable, or if it is viewable, personnel operating internal CCTV systems must be trustworthy and trusted not to try to obtain and disclose passwords.
SO.PHY_CTL	The computer the User Program is installed on must not fall under temporary and undetected physical control of a threat agent. Appropriate physical security measures and physical security policies must be in place to manage risk of this event occurring.
SO.REC_PHY	The computer the One-Time Password Program is installed on must not fall under the physical control of a threat agent.
SO.USR_ATH	When the administrator assists the user in recovering access to their data with the access recovery procedure, the administrator must assure himself or herself of the user's identity. This is to prevent a threat agent who has stolen or gained unattended access to the users machine while it is not logged in from using the access recovery procedure by pretending to be the user. The administrator must use a reliable and secure method to authenticate users.
SO.INI_SEC	While the machine is in the pre-installed state with the default password, before the user has changed the password, adequate physical security precautions should be taken to ensure that a threat agent is not able to use the default user password to obtain the encryption keys.

5. IT Security Requirements

5.1. TOE Security Requirements

5.1.1. TOE Security Functional Requirements

The following definitions and templates are taken from CC part 2. Completed template parts are shown in *italics*. Unless otherwise specified the components are hierarchical to no other components. Unless otherwise specified the components have no dependencies.

5.1.1.1. General Application Functionality

This section gives the security functional requirements for the application in general.

FPT_FLS.1	**Failure with preservation of secure state**

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: *power failure, or physical failure of machine or its storage devices*].

Dependencies: ADV_SPM.1 Informal TOE security policy model

FPT_RCV.4 Function recovery

FPT_RCV.4.1 The TSF shall ensure that [assignment: *power failure, or physical failure of the machine or its storage devices, or intentional power down and startup*] have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

Dependencies: ADV_SPM.1 Informal TOE security policy model

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles [assignment: *user, EP Hard Disk Administrator, Corporate Administrator, Local Administrator*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Dependencies: FIA_UID.1 Timing of identification

FPT_TST.1 TSF testing

FPT_TST.1.1 The TSF shall run a suite of self-tests [selection: during initial start-up] 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 TSF data.

FPT_TST.1.3 The TSF shall provide authorized users with the capability to verify the integrity of stored TSF executable code.

Dependencies: FPT_AMT.1 Abstract machine testing

FTA_TAB.1 Default TOE access banners

FTA_TAB.1.1 Before establishing a user session, the TSF should display an advisory warning message regarding unauthorized use of the TOE.

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FPT_FLS.1	ADV_SPM.1	Not Included – EP Hard Disk does not try to protect against undetected software and security state modification, only against user data recovery from a machine that physical access is gained to while the application is locked, or while the machine is powered down. No security relevant data that it is possible to cryptographically protect is written to the disk in unprotected form. Therefore, the system should never be in an insecure state.
FPT_FLS.1	ADV_SPM.1	Not Included
FPT_RCV.4	ADV_SPM.1	Not Included
FMT_SMR.1	FIA_UID.1	Included in the administrator password and user password sections. Note: In the user password section, FIA_UID.1 is not explicitly included – but the TSF captures and records the user's identity at configuration time.
FPT_TST.1	FPT_AMT.1	Not Included – There are no security assumptions made of the underlying hardware that could reasonably or reliably be automatically tested.

5.1.1.2. Initial Encryption

This section gives the security functional requirements for the Initial Encryption function.

Cryptography related components are referenced in the sections on User Password, Disk Key, and Disk KEK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the pre-installed SFP*] on [assignment: *the configuration and use of Authenti-Check, and the Access Recovery function*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *pre-installed SFP*] to objects based on [assignment: *whether or not the installation is in the pre-installed state*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the user name and password have default values, the installation will be considered to be in the pre-installed state, and access to the Authenti-Check configuration and recovery, and the Access Recovery function will be disabled*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control function so FMT_MSA.3 is not relevant. The only criterion for access is that the user has chosen a password.

5.1.1.3. On-the-Fly Encryption

This section gives the security functional requirements for the main function of EP Hard Disk: the on-the-fly encryption and decryption of user data.

Cryptography related components are referenced in the sections on User Password, Disk Key, and Disk KEK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the mandatory access control SFP*] on [assignment: *user data stored on encrypted partitions*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *mandatory access control SFP*] to objects based on [assignment: *user authentication*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the user, the Corporate Administrator or the Local Administrator have successfully authenticated themselves then access to modify, read, and create files on the encrypted partition shall be granted*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control function so FMT_MSA.3 is not relevant. The only criterion for access is successful authentication.

5.1.1.4. Full Decryption

This section gives the security functional requirements for user and administrator decryption of user partitions. Full decryption is the process that is gone through to convert an encrypted partition back into a plaintext partition with no on-the-fly decryption necessary to read.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the decryption access control SFP*] on [assignment: *access to the decryption function*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *the decryption access control SFP*] to objects based on [assignment: *the current setting of the allow user decrypt attribute*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the allow user decrypt attribute is set then the user will be allowed to decrypt partitions; if the allow user decrypt attribute is not set, the user shall not be allowed to decrypt partitions*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *if the user is an administrator then the user will be allowed to decrypt partitions regardless of the setting of the allow user decrypt attribute*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the [assignment: *access control SFP*] to provide [selection: *permissive*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [assignment: *EP Hard Disk Administrator*] to specify alternative initial values to override the default values when an object or information is created.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks

FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	Included
FMT_MSA.3	FMT_MSA.1	Included in section on Administrator Configuration.
	FMT_SMR.1	Included in section on General Application Functionality.

5.1.1.5. User Password

This section gives the security functional requirements for selection and use of user passwords.

The user password is used to fill the high-level security objective of authenticating the user.

FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [assignment: *the minimum length requirements set by the administrator*]

FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow [assignment: *recovery of the user disk key*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.7 Protected authentication feedback

FIA_UAU.7.1 The TSF shall provide only [assignment: *display of asterisks in place of characters typed*] to the user while authentication is in progress.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when [assignment: *the administrator specified number*] of unsuccessful authentication attempts occur related to [assignment: *user logon*].

Dependencies: FIA_UAU.1 Timing of authentication

FMT_SAE.1 Time-limited authorization

FMT_SAE.1.1 The TSF shall restrict the capability to specify an expiration time for [assignment: *user passwords*] to [assignment: *the EP Hard Disk Administrator*].

FMT_SAE.1.2 For each of these security attributes, the TSF shall be able to [assignment: *deny further logon until the user has chosen a new password*] after the expiration time for the indicated security attribute has passed.

Dependencies: FMT_SMR.1 Security roles
FPT_STM.1 Reliable time stamps

FPT_STM.1 Reliable time stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the password history SFP*] on [assignment: *selection of new user passwords*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *password history SFP*] to objects based on [assignment: *whether or not the password is contained in the recently used password list*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the new password is not in the recently used password list then the new password shall be accepted and shall replace the old password for authentication purposes*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FIA_UAU.1	FIA_UID.1	Not Included – The TSF captures and records the user's identity at configuration time.
FIA_UAU.7	FIA_UAU.1	Included
FIA_AFL.1	FIA_UAU.1	Included
FMT_SAE.1	FMT_SMR.1	Included in general application functionality section.
	FPT_STM.1	Included
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	The password history is not statically assigned; it is initialized by EP Hard Disk to an empty list so FMT_MSA.3 is not relevant.

5.1.1.6. Disk Key

This section gives security functional requirements for the management and use of the symmetric AES Disk Key used by EP Hard Disk to encrypt the data stored on the user's disk.

The Disk Key helps satisfy the high-level security objective of: data separation of user data.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *the cryptographic random number generator from* **[RNG]**] and specified cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *no standard*].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.3 Cryptographic key access

FCS_CKM.3.1 The TSF shall perform [assignment: *key recovery*] in accordance with a specified cryptographic key access method [assignment: *EP Hard Disk key recovery method*] that meets the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *overwriting with new key*] that meets the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *user data encryption and user data decryption*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Not Included – The Disk Key is a raw AES key and has no security attributes.
FCS_CKM.3	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Not Included
FCS_CKM.4	FCS_CKM.1	Included
	FMT_MSA.2	Not Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FCS_MSA.2	Not Included

5.1.1.7. Disk KEK

This section gives security functional requirements for the management and use of the symmetric Disk Key Encryption Key (KEK) used by EP Hard Disk to perform cryptographic key encryption and cryptographic key decryption of the Disk Key.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived using the PBKDF2 key derivation function from the user name and password*] and specified cryptographic key sizes [assignment: *256-bit keys though overall strength depends upon user password selection*] that meet the following: [assignment: *PKCS#5 standard* **[PBKDF2]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *cryptographic key encryption and cryptographic key decryption*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The Disk KEK is never stored persistently; it is freshly computed from the user name and password and retained temporarily only in memory.
	FMT_MSA.2	Not Included – The Disk KEK is a raw AES key and has no security attributes.
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.8. Authenti-Check Logon

This section gives the security functional requirements for the Authenti-Check Logon mechanism. Requirements about the Authenti-Check key recovery procedure are given in the section on Authenti-Check Key Recovery.

FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [assignment: *the minimum length requirements set by the administrator*]

FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow [assignment: *recovery of the user disk key*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FIA_UAU.1	FIA_UID.1	Not Included – the TSF captures and records the user's identity at configuration time.

5.1.1.9. Authenti-Check Key Recovery

This section gives the security functional requirements for the Authenti-Check Key Recovery function.

Cryptography related components are referenced in the section on Authenti-Check KRK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the Authenti-Check recovery access SFP*] on [assignment: *users attempting to recover access using the Authenti-Check mechanism after forgetting their passwords*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *Authenti-Check recovery access SFP*] to objects based on [assignment: *user authentication*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the user is able to authenticate himself using the Authenti-Check mechanism then access shall be granted*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control function so FMT_MSA.3 is not relevant. The only criterion for access is successful administrator authentication.

5.1.1.10. Authenti-Check KRK

This section gives security functional requirements for the management and use of the Authenti-Check Key Recovery Key (Authenti-Check KRK) used by EP Hard Disk to decrypt the Disk Key that is stored in encrypted form in the recovery block.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *the Authenti-Check Key Recovery Key is computed as the SHA-256 hash of the concatenation of all the answers*] and specified cryptographic key sizes [assignment: *256-bit keys for the Authenti-Check KRK*] that meet the following: [assignment: *SHA-256 standard* **[SHA-256]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *encryption and decryption of the Authenti-Check portion of the recovery block*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys though actual strength will depend on the amount of entropy in the user-selected answers given knowledge of the questions*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The Authenti-Check KRK is never stored persistently; it is computed from the user question answers and retained temporarily only in memory.
	FMT_MSA.2	Not Included – The Authenti-Check KRK is a raw key and has no security attributes.
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.11. Administrator Configuration

This section gives the security functional requirements for the administrator configuration of the User Program prior to installation, and subsequent signed updates to User Program configuration after installation.

Cryptography related components are referenced in the sections on Administrator Password, Administrator Database Key, Administrator Database KEK, Administrator ECDSA private key, and Administrator ECDSA public key.

FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to [selection: *modify the behavior of*] the functions [assignment: *logon function and On-the-Fly Encryption function, via administrator configuration update messages*] to [assignment: *the EP Hard Disk Administrator*].

Dependencies: FMT_SMR.1 Security roles

FMT_MSA.2 Secure security attributes

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes.

Dependencies: ADV_SPM.1 Informal TOE security policy model
FDP_ACC.1 Subset access control

FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the [assignment: *safe configuration settings SFP*] to restrict the ability to [selection: *modify*] the security attributes [assignment: *logon message, optimization option, administrator public keys, allowed number of incorrect password entries, allow user decrypt*] to [assignment: *EP Hard Disk Administrator*].

Dependencies: FDP_ACC.1 Subset access control
FMT_SMR.1 Security roles

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the safe configuration settings SFP*] on [assignment: *administrators setting and modifying configuration parameters*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *safe configuration settings SFP*] to objects based on [assignment: *user authentication: only the EP Hard Disk Administrator can modify; and the settings shall be restricted to secure values*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the user is the EP Hard Disk Administrator, and is authenticated and the attempted modification is a secure value allow the modification*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FMT_MOF.1	FMT_SMR.1	Included in the section on general application functionality.
FCS_MSA.2	ADV_SPM.1	Not Included – EP Hard Disk does not try to protect against undetected software and security state modification, only against user data recovery from a machine that physical access is gained to while the application is locked, or while the machine is powered down. No security relevant data that it is possible to cryptographically protect is written to the disk in unprotected form. Therefore, the system should never be in an insecure state.
	FDP_ACC.1	Included
	FMT_MSA.1	Included
	FMT_SMR.1	Included in the general application functionality section.
FCS_MSA.1	FDP_ACC.1	Included
	FMT_SMR.1	Included
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control function so FMT_MSA.3 is not relevant. The only criterion for access is successful administrator authentication.

5.1.1.12. Administrator Database Encryption

This section gives the security functional requirements for the encryption of the Administrator Database.

Cryptography related components are referenced in the sections on Administrator Database Key and Administrator Database KEK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the mandatory access control SFP*] on [assignment: *administrator keys and related data stored in the encrypted Administrator Database*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *mandatory access control SFP*] to objects based on [assignment: *administrator authentication*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the EP Hard Disk Administrator has successfully authenticated himself or herself, then access to modify, read, or create files on the Administrator Database shall be granted*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control function so FMT_MSA.3 is not relevant. The only criterion for access is successful authentication.

5.1.1.13. Administrator Password

This section gives the security functional requirements for selection and use of EP Hard Disk Administrator, Corporate Administrator, and Local Administrator passwords.

The administrator password is used to provide the high-level security objective of authenticating the EP Hard Disk Administrator to the Administrator Program, and of authenticating Local Administrators and the Corporate Administrator to the administrator logon function of the User Program.

Note: The user referred to in the non-italicized parts of components in this section refers to the EP Hard Disk Administrator, Corporate Administrator, and/or Local Administrator (as appropriate given the context), and should not be confused with a reference to the user. User in this context is meant in the sense that the EP Hard Disk Administrator, Corporate Administrator and Local Administrators are users of the TOE.

FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [assignment: *the minimum password length requirements required by EP Hard Disk for administrator passwords*].

FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1

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.

Dependencies: FIA_UID.1 Timing of identification

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1

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.

FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when [assignment: *the EP Hard Disk specified number*] of unsuccessful authentication attempts occur related to [assignment: *the User Program Admin Logon function*].

Dependencies: FIA_UAU.1 Timing of authentication

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FIA_UAU.1	FIA_UID.1	Included (by inclusion of hierarchical component FIA_UID.2)
FIA_AFL.1	FIA_UAU.1	Included (by inclusion of hierarchical component FIA_UAU.2).

5.1.1.14. Administrator Database Key

This section gives security functional requirements for the management and use of the symmetric AES Administrator Database key used by EP Hard Disk to encrypt the data (passwords) stored in the Administrator Database.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *the cryptographic random number generator from* **[RNG]**] and specified cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *no standard*].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.3 Cryptographic key access

FCS_CKM.3.1 The TSF shall perform [assignment: *key backup*] in accordance with a specified cryptographic key access method [assignment: *EP Hard Disk key backup to removable media for physically secure storage*] that meets the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: *overwriting the encrypted database key with the new encrypted database key*] that meets the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *database data encryption and database data decryption*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Not Included – The Administrator Database Key is a raw AES key and has no security attributes.
FCS_CKM.3	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Not Included
FCS_CKM.4	FCS_CKM.1	Included
	FMT_MSA.2	Not Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FCS_MSA.2	Not Included

5.1.1.15. Administrator Database KEK

This section gives security functional requirements for the management and use of the symmetric Administrator Database Key Encryption Key (KEK) used by EP Hard Disk to perform cryptographic key encryption and cryptographic key decryption of the Administrator Database Key.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived using PBKDF2 key stretching function from the EP Hard Disk Administrator's user name and password*] and specified cryptographic key sizes [assignment: *256-bit keys though overall*

strength depends upon EP Hard Disk Administrator's password selection] that meet the following: [assignment: *PKCS#5 standard* **[PBKDF2]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *cryptographic key encryption and cryptographic key decryption*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The Administrator Database KEK is never stored persistently; it is freshly computed from the EP Hard Disk Administrator user name and password and retained temporarily only in memory.
	FMT_MSA.2	Not Included – The Administrator Database KEK is a raw AES key and has no security attributes.
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.16. Administrator ECDSA Private Key

This section gives security functional requirements for the management and use of the administrator DSA private signing key used by EP Hard Disk to create signatures on administrator public key updates and to create signatures on configuration update messages.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived using PBKDF2 from the administrator user name and password*] and specified cryptographic key sizes [assignment: *160-bit keys though overall strength depends upon administrator password selection*] that meet the following: [assignment: *PKCS#5 standard* **[PBKDF2]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.3 Cryptographic key access

FCS_CKM.3.1 The TSF shall perform [assignment: *key backup*] in accordance with a specified cryptographic key access method [assignment: *storage in encrypted Administrator Database*] that meets the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *signatures on update messages*] in accordance with a specified cryptographic algorithm [assignment: *ECDSA signatures*] and cryptographic key sizes [assignment: *160-bit keys*] that meet the following: [assignment: *DSA standard* **[ECDSA]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included

	FCS_CKM.4	Not Included – The administrator ECDSA private key is encrypted when stored persistently.
	FMT_MSA.2	Not Included – The administrator ECDSA private key is a raw ECDSA key and has no security attributes.
FCS_CKM.3	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.17. Administrator ECDSA Public Key

This section gives security functional requirements for the management and use of the administrator DSA public signing key used by EP Hard Disk to verify signatures on administrator public key updates and to verify signatures on configuration update messages.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived according to the ECDSA key generation algorithm from the administrator ECDSA private key*] and specified cryptographic key sizes [assignment: *233-bit curve and public key*] that meet the following: [*DSA standard* **[ECDSA]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.2 Cryptographic key distribution

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *included with configuration parameters at installation*] that meets the following: [assignment: *no standard*].

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *a signed update message*] that meets the following [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *verification of signatures on update messages*] in accordance with a specified cryptographic algorithm [assignment: *ECDSA signatures*] and cryptographic key sizes [assignment: *233-bit curve and public key*] that meet the following: [assignment: *DSA standard* **[ECDSA]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The administrator ECDSA public key is a public key so destruction is unnecessary.
	FMT_MSA.2	Not Included – The administrator ECDSA public key is a raw ECDSA key and has no security attributes.
FCS_CKM.2	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.18. Elliptic Curve Key Recovery

This section gives security functional requirements for the Elliptic Curve-based key recovery and Elliptic Curve-based User Program Admin Logon functions.

Cryptography related components are referenced in the sections on Administrator ECDH Private Key, Administrator ECDH Public Key, and Elliptic Curve KRK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the Elliptic Curve recovery access control SFP*] on [assignment: *the key recovery procedure for the Elliptic Curve-based key recovery procedure*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *Elliptic Curve recovery access control SFP*] to objects based on [assignment: *administrator authentication*].

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

(1) *if the Corporate Administrator or Local Administrator has physical possession of the One-Time Password Program, then access to the Access Recovery function shall be granted*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No attributes are used in the access control functions for the Access Recovery function, so FMT_MSA.3 is not relevant. The only criterion for access to both functions is successful authentication.

5.1.1.19. User Program Admin Logon

This section gives the security functional requirements for the logon function of the User Program that is available to administrators.

Cryptography related components are referenced in the sections on Administrator ECDH Private Key, Administrator ECDH Public Key, and Elliptic Curve KRK below.

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *the User Program Admin Logon access control SFP*] on [assignment: *user data stored on encrypted partition for the User Program Admin Logon function*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *User Program Admin Logon access control SFP*] to objects based on [assignment: *administrator authentication*].

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

(1) if the Corporate Administrator or Local Administrator has successfully authenticated himself to the User Program then access via the User Program Admin Logon function shall be granted].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included

	FMT_MSA.3	No attributes are used in the access control functions for the User Program Admin Logon function, so FMT_MSA.3 is not relevant. The only criteria for access to the functions is successful authentication.

5.1.1.20. Administrator ECDH Private Key

This section gives security functional requirements for the management and use of the administrator ECDH private decryption key used by EP Hard Disk to recover the ECKRK from the recovery message by deriving the same key using ECDH; the ECKRK is used to decrypt the recovery block to recover the Disk Key.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived using PBKDF2 key derivation function from the Corporate or Local Administrator's user name and password*] and specified cryptographic key sizes [assignment: *233-bit keys*] that meet the following: [assignment: *PKCS#5 standard* **[PBKDF2]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *decryption of the recovery message*] in accordance with a specified cryptographic algorithm [assignment: *ECDH key negotiation*] and cryptographic key sizes [assignment: *233-bit keys*] that meet the following: [assignment: *IEEE P1363 standard* **[ECDH]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included

	FCS_CKM.4	Not Included – The ECDH private key is never stored in encrypted or plaintext form on disk, and is freshly computed from the Corporate or Local Administrator's user name and password prior to use.
	FMT_MSA.2	Not Included – The administrator ECDH private key is a raw ECDH key and has no security attributes.
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FCS_MSA.2	Not Included

5.1.1.21. Administrator ECDH Public Key

This section gives security functional requirements for the management and use of the administrator ECDH public encryption key that is used by EP Hard Disk to derive the ECKRK and that is used to encrypt the Disk Key.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *derived from the administrator ECDH private key*] and specified cryptographic key sizes [assignment: *233-bit keys*] that meet the following: [assignment: *IEEE P1363 standard* **[ECDH]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_CKM.2 Cryptographic key distribution

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *included with configuration parameters at installation*] that meets the following: [assignment: *no standard*].

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [assignment: *a signed update message*] that meets the following [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *encryption of the recovery block (also extracted into the recovery message)*] in accordance with a specified cryptographic algorithm [assignment: *ECDH encryption as specified in IEEE P1363 standard*] and cryptographic key sizes [assignment:*233-bit keys*] that meet the following: [assignment: *no standard*].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The administrator ECDH public key is a public key and no security value is attributable to its destruction.
	FMT_MSA.2	Not Included – The administrator ECDH public key is a raw ECDH key and has no security attributes.
FCS_CKM.2	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.22. Elliptic Curve KRK

This section gives security functional requirements for the management and use of the Elliptic Curve Key Recovery Key (ECKRK) used by EP Hard Disk to decrypt the Disk Key that is stored in encrypted form in the recovery block.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: *first a shared key is negotiated with ECDH using the administrator's ECDH public key, then the ECKRK is derived using KDF2 from the user name and the negotiated key*] and specified cryptographic

key sizes [assignment: *for the negotiated key: 233-bit keys and for the ECKRK: 256 bits*] that meet the following: [assignment: ***IEEE P1363** standard* **[ECDH]** *and* ***IEEE P1363a** standard* **[KDF2]**].

Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 Cryptographic Operation

FCS_COP.1.1 The TSF shall perform [assignment: *encryption and decryption of the recovery block*] in accordance with a specified cryptographic algorithm [assignment: *AES in CBC mode*] and cryptographic key sizes [assignment: *256-bit keys*] that meet the following: [assignment: *AES standard* **[AES]** *and AES modes of operation standard* **[AES-MODES]**].

Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Not Included – The ECKRK is never stored persistently; it is computed by the administrator and retained temporarily only in memory.
	FMT_MSA.2	Not Included – The ECKRK is a raw key and has no security attributes.
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Not Included
	FCS_MSA.2	Not Included

5.1.1.23. User Configuration

This section gives the security functional requirements for the user configuration of the User Program.

Some of the user configuration options are restricted in the values they may hold or their availability by the defaults set or updated by the administrator configuration options. The administrator configuration security functional requirements are given in section Administrator Configuration.

Note: It is possible for users, Corporate Administrators, and Local Administrators to logon to the User Program and edit the user configuration. The same restrictions apply equally to all roles when they are using the User Program, so in some contexts in this section the word *user* is used to refer to the entity currently logged into the User Program where this user could be the user, the Corporate Administrator, or a Local Administrator.

FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to [selection: *modify the behavior of*] the functions [assignment: *logon function, and on-the-fly encryption function*] to [assignment: *the user, Corporate Administrator, and Local Administrator*].

Dependencies: FMT_SMR.1 Security roles

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce [assignment: *user configuration access SFP*] on [assignment: *users and administrators attempting to set or modify user-settable configuration options using the User Program*].

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [assignment: *user configuration access SFP*] to objects based on [assignment: *user authentication and administrator default configuration settings*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *if the user is able to authenticate himself and the setting or modification does not conflict with the administrator default configuration options then the modification shall be approved*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [assignment: *none*].

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

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

The following table documents which dependencies of the components referenced in this section are included, and where dependencies are not included the rationale for their omission.

IT Security Requirement	Dependencies	Remarks
FDP_ACC.1	FDP_ACF.1	Included
FMT_MOF.1	FMT_SMR.1	Included in the section on general application functionality.
FDP_ACF.1	FDP_ACC.1	Included
	FMT_MSA.3	No user attributes are used in the access control function so FMT_MSA.3 is not relevant. The inputs to the access decision are user authentication and the administrator default configuration settings. The security functional requirements for the administrator default configuration settings are specified in section Administrator Configuration.

5.1.2. TOE Security Assurance Requirements

This section lists the assurance requirements the TOE must meet to be evaluated at Evaluation Assurance Level 1. The following components are taken from CC part 3. The components in the following section have no dependencies unless otherwise noted. These components are included by reference only as there are no parameters to be assigned; the body can be found in CC part 3.

ACM_CAP.1 **Version numbers**

ADO_IGS.1 **Installation, generation, and start-up procedures**

Dependencies: AGD_ADM.1 Administrator guidance

ADV_FSP.1 **Informal functional specification**

Dependencies: ADV_RCR.1 Informal correspondence demonstration

ADV_RCR.1 **Informal correspondence demonstration**

AGD_ADM.1 **Administrator guidance**

Dependencies: ADV_FSP.1 Informal functional specification

AGD_USR.1 **User guidance**

Dependencies: ADV_FSP.1 Informal functional specification

ATE_IND.1 **Independent testing – conformance**

Dependencies: ADV_FSP.1 Informal functional specification
AGD_ADM.1 Administrator guidance
AGD_USR.1 User guidance

5.2. Security Requirements for the IT Environment

The TOE has no programmatic asserted dependencies on the IT environment, so this section is left blank. The policy and configuration requirements of the IT environment are discussed in the Section 4.2.2 Non-IT Environmental Security Objectives.

6. TOE Summary Specification

6.1. TOE Security Functions

TSF.NO_STO The TOE does not store any sensitive information on disk — specifically user plaintext, user passwords, user question answers, or cryptographic keys — to disk in unprotected form at any time

	provided the related security assumptions are met. This security function is implemented in all areas of all of the application functions: Administrator Logon function, User Program Setup wizard, Configuration Update function, Initial Encryption, Initial Encryption (Pre-installed), On-the-Fly Encryption function, User Logon function, Authenti-Check Logon function, Access Recovery function, Full Decryption function, User Program Admin Logon function, Access Recovery function and Hard Disk Repair function with the exception of circumstances where the function's defined behavior is to permanently decrypt data. The related security objectives are SO.ENC_ALL (that all partitions are encrypted with the TOE) and SO.HIB_STO (that laptop hibernation features are disabled).
TSF.KEY_OVR	As an additional counter-measure in addition to the function described in TSF.NO_STO, when keys are changed or decommissioned, the areas of disk where the encrypted form was stored are over-written with zeros, or are over-written with the new encrypted key. This helps ensure that even if a key derived from a weak password were used to encrypt the key, security is improved by later choosing a stronger password, or by later uninstalling or reinstalling the software. This security function is implemented in all areas of all of the application functions: Administrator Logon function, User Program Setup wizard, Configuration Update function, Initial Encryption, Initial Encryption (Pre-installed), On-the-Fly Encryption function, User Logon function, Authenti-Check Logon function, Access Recovery function, Full Decryption function, User Program Admin Logon function, Access Recovery function and Hard Disk Repair function where any keys are changed or decommissioned.
TSF.ENC_REL	The TOE makes reasonable efforts to ensure that no data is lost if encryption, decryption, or other TOE functions are unexpectedly interrupted due to power loss, or temporary failure of a storage device that does not itself lose data. This security function is implemented in all areas of all of the application functions: Administrator Logon function, User Program Setup wizard, Configuration Update function, Initial Encryption, Initial Encryption (Pre-installed), On-the-Fly Encryption function, User Logon function, Authenti-Check Logon function, Access Recovery function, Full Decryption function, User Program Admin Logon function, Access Recovery function and Hard Disk Repair function.

	Note: The option to disable power-loss protection during initial encryption described in SO.PWR_LOS, if selected, temporarily invalidates this assurance.
TSF.SEC_ROL	The TOE maintains the following roles: the user, EP Hard Disk Administrator, Corporate Administrator, and Local Administrator. The roles are used throughout the application functions.
TSF.USR_LOG	The TOE provides a User Logon function where the user must type in his password. This function is implemented in the User Logon application function. The password is not displayed as it is typed; instead, asterisks are displayed to reduce the risk of a threat agent observing the password as it is typed. The Disk KEK is derived from the user password. If the user incorrectly types their password more than an EP Hard Disk Administrator-configured maximum, the application will lockup. If the machine is rebooted, the user can continue trying more passwords. The feature is designed to frustrate attempts by a threat agent to guess passwords by repeated entry of guesses. The user will also be locked out and forced to choose a new password if the user has not changed their password more recently than an EP Hard Disk Administrator-configured minimum password change time interval from the time of last password change. See TSF.PWD_HST for a description of TSF functions relating to changing user passwords.
TSF.PWD_HST	When the user changes their password, if the password history configuration option is enabled by the EP Hard Disk Administrator in the User Program Setup wizard, the user will not be able to choose a new password that is the same as the recent passwords stored in the password history log. The number of passwords retained in the password history log is also configured by the EP Hard Disk Administrator. This function is implemented as part of the User Logon function component of the User Program.
TSF.TIM_STP	The TSF uses the system clock to obtain the time for evaluating the rules involving time such as the user password expiry function. No extra steps are taken to prevent the user modifying the time on their machine. This function is implemented as part of the User Logon function.
TSF.ATK_LOG	The TOE provides the Authenti-Check Logon function where the user must type in answers to questions he provided during configuration. The Authenti-Check Key Recovery Key (KRK) is derived from the answers as described in TSF.ATK_KRK. This function is implemented as part of the Authenti-Check Logon function of the User Program.

TSF.ATK_REC	The TOE provides an Authenti-Check Key Recovery function to allow the user to choose a new password in the event that they forget their password. The user logs in using the Authenti-Check question and answer mechanism described in TSF.ATK_LOG, and the Authenti-Check Key Recovery Key (KRK) is derived as described in TSF.ATK_KRK. The KRK is used to decrypt a copy of the Disk Key encrypted under the Authenti-Check KRK stored in the recovery block using the AES algorithm in CBC mode, which conforms to AES **[AES]** and AES modes of operation **[AES-MODES]** standards. This function is implemented in the Authenti-Check Logon of the User Program.
TSF.ATK_KRK	The Authenti-Check Key Recovery Key (KRK) is derived from the users answers to the questions described in TSF.ATK_LOG, by hashing all of the answers using the SHA-256 hash function that conforms to the SHA-256 **[FIPS-SHA2]** standard to arrive at the Authenti-Check KRK. This function is implemented in the Authenti-Check Logon function of the User Program.
TSF.ADM_LOG	The TOE provides an EP Hard Disk Administrator logon function to the Administrator Program. The EP Hard Disk Administrator DSA private key and the Administrator Database Key are derived from the EP Hard Disk Administrator password. No EP Hard Disk Administrator functions should be available in the Administrator Program until the EP Hard Disk Administrator has successfully logged on. This function is implemented in the Administrator Logon function of the Administrator Program.
TSF.UAD_LOG	The TOE provides a logon function to the User Program for administrators to gain access to user data where the administrator must enter their user name. The Corporate Administrator has a fixed user name “Corporate Admin”. In this way, the administrator roles are identified during logon. The administrator ECDH private key is derived from their password, and this is used to decrypt the corresponding recovery block for the administrator role they authenticated as. No functions are available to the administrator in the User Program until the administrator has successfully logged on with the User Program Admin Logon function as described in TSF.ADM_LOG. The access is achieved by decrypting a recovery block component with the Elliptic Curve Key Recovery Key (KRK) as described in TSF.REC_BLK. The derivation of the ECKRK is described in TSF.EC_KRK. This function is implemented in the User Program Admin Logon function of the User Program.

TSF.CRY_TST	The TOE implements a suite of self-tests invoked at startup on some of the cryptographic constructs it uses. The self-tests are implemented using the standard test vectors as published in the associated standard where available. In some cases, a subset of the full set of test vectors from the associated standard (where applicable) is implemented to reduce time and space overheads of the test suite. This function is implemented at start up of all of the Programs: Administrator Program, User Program, One-Time Password Program and Recovery Program.
TSF.TDA_CSM	The TOE verifies checksums on the TSF data. All of the application functions implement checksums on the TSF data they read and write.
TSF.BIN_CSM	The TOE verifies checksums on its EPOS (Pre-Dos) executables, drivers and libraries at startup to resist software tampering. These checksums are implemented for the On-the-Fly Encryption function. Note: The Windows-level executables are not check-summed.
TSF.ACC_BAN	The TOE displays an EP Hard Disk Administrator configurable access banner at startup before user authentication. This function is implemented in the User Logon function.
TSF.REC_SEC	The access recovery procedure helps users to recover access to their data in the event that they lose their passwords. The recovery procedure in the User Program is available before logging on. The user must authenticate themselves to the administrator using a security officer-approved authentication method as described in security objective SO.USR_ATH before the administrator should proceed with the key recovery mechanism described in TSF.KEY_REC. This function is implemented in the Access Recovery function of the User Program and the administrator part is implemented in the Access Recovery function of the One-Time Password Program. Note: In practice the administrator could use the User Program to participate in the access recovery protocol. However, in the case of the administrator, this function has limited use, as the administrator already possesses the password allowing direct access. A scenario where this pattern of use would be logical would be where the administrator is suspicious about the integrity of the User Program and does not want to type his password into the machine, and yet desires to gain access to the user data. In this event, he could play the part of the user and the administrator in the access recovery procedure while using two machines (the user machine and another

trusted machine with the One-Time Password Program installed on it).

TSF.KEY_REC — After the user and administrator have completed the authentication described in TSF.REC_SEC, the administrator and the user exchange messages, and the cryptographic operations implemented by the One-Time Password Program — acting in coordination with the message sent and subsequent processing of the response on the User Program — provides secure key recovery. After access has been recovered, the user must choose a new password. The recovery block is overwritten with a freshly computed recovery block. The recovery mechanism is based on Elliptic Curve Diffie-Hellman, but the use of the standard cryptographic components to construct the recovery mechanism adheres to no standard. The ECDH key pair generation is described in TSF.DH_KP. The encryption and decryption of the recovery block component that is sent as the recovery request message is described in TSF.EC_KRK. The other recovery block component is the encrypted Disk Key that is encrypted using AES with the ECKRK. The derivation of the ECKRK is described in TSF.EC_KRK. The AES decryption of the encrypted Disk Key with the ECKRK is described in TSF.REC_BLK. This function is implemented in the Access Recovery function of the User Program and the administrator part is implemented in the Access Recovery function of the One-Time Password Program.

TSF.REC_BLK — The recovery block is composed of two parts that relate to Elliptic Curve-based key recovery. The first part is a key negotiation parameter derived from the public key and a random number chosen by the User Program, and the second part the encryption of the Disk Key with the ECKRK using AES in CBC mode. The AES decryption conforms to the **[AES]** and **[AES-MODES]** standards. The generation of the Elliptic Curve KRK is described in TSF.EC_KRK. The recovery block creation is implemented as part of the Initial Encryption function.

TSF.EC_KRK — The Elliptic Curve Key Recovery Key (ECKRK) is derived in two stages: first, a negotiated key is derived, and then the ECKRK is derived from the negotiated key and the user's user name. The two steps are described below. During encryption, the negotiated key is computed by the user who chooses a random negotiation parameter; during decryption, the negotiated key is derived from the encrypted negotiation value and the corresponding administrator's private key. This is a standard application of the ECDH key negotiation algorithm as described in **[ECDH]**. Once the negotiated key is computed, the ECKRK is derived from the

	negotiated key and the user's user name using KDF2 **[KDF2]**. For encryption, the ECKRK derivation is implemented as part of the Initial Encryption function; for decryption, the ECKRK derivation is implemented as part of the Access Recovery function.
TSF.PWD_STR	The TOE enforces minimum password length requirements on the user's selection of passwords. The minimum password length is configured by the administrator as described in TSF.ADM_CFG. The password length restrictions are implemented in the User Program. The password length restrictions are set in the User Program Setup wizard of the Administrator Program.
TSF.ADM_CFG	The minimum password length, access banner, and recovery methods to be used are set by the EP Hard Disk Administrator in the configuration options of the User Program install package. These settings can be later updated by the EP Hard Disk Administrator, who uses the Administrator Program to create and distribute signed configuration update messages that are read by installations of the User Program. The TOE ensures that only safe configuration options are selectable by the administrator. The list of user configuration defaults: EP Hard Disk Administrator ECDSA key, Corporate Administrator and Local Administrator ECDH public keys, logon banner message, password management parameters (time-outs, history settings, max incorrect attempts before lockout), user settings (whether the user can change their password, whether Single Sign-On feature is enabled, whether encryption is optimized for speed, whether the user can change their user name, whether the user can turn off encryption) and user messages (startup message, logon message, recovery procedure message) can only be set by the EP Hard Disk Administrator (and not by the Corporate Administrator, Local Administrator, or the user). The update configuration messages are signed by the administrator ECDSA key: the signature generation and subsequent verification by the user conforms to the FIPS DSA **[ECDSA]** standard. A configuration message can contain new ECDSA and ECDH public keys when the administrator changes his or her password to distribute the new keys to the user. This aspect of administrator configuration messages is described in TSF.PK_DST. The administrator configuration is implemented in the Administrator Program User Program Setup wizard.
TSF.PK_DST	The original ECDH public keys for the Corporate Administrator and one Local Administrator and the original ECDSA public key for the EP Hard Disk Administrator are contained in the User Program install package. There is one install package per domain of control for each corresponding Local Administrator. A

	configuration message can contain new ECDSA and ECDH public keys when the administrator changes their password to distribute the new keys to the user. Both the new ECDSA public key and the new ECDH public key (and any other configuration changes included in the message) are signed with the previous DSA public key. The configuration message includes all signed DSA public key update message components back to the message replacing original DSA public key installed in the User Program installation. This approach ensures that a user who misses some configuration update messages is guaranteed to be able to verify the chain of signatures on public keys and hence to verify the signature on the current configuration update message to be assured that the configuration message is authenticated. More information about the configuration messages and configuration update mechanism are given in TSF.ADM_CFG. The initial key distribution is implemented as part of the User Program Setup wizard, and the key updates are implemented as part of the Configuration Update functions of the Administrator Program.
TSF.USR_CFG	The user configuration options: user name and password can be configured by the user once he has successfully logged in to the User Program (or by an administrator if they having logged in to the User Program using the User Program Admin Logon function). Other configuration options are configurable only by the EP Hard Disk Administrator using the mechanisms described in TSF.ADM_CFG. The user configuration options to change user name and password may be unavailable depending on the value of the corresponding setting made by the EP Hard Disk Administrator. In addition, the user configuration values that are settable are restricted by the TOE to those that are safe. The User Configuration options are implemented as part of the User Program.
TSF.INI_ENC	The TOE first asks the user to provide a user name and password, and then encrypts the disk as described in TSF.DSK_ENC. The Initial Encryption is implemented as part of the Initial Encryption and Initial Encryption (Pre-installed) functions of the User Program. (The Pre-installed option is used if option 1 below is selected). As part of the User Program Setup process, there are three options available to the EP Hard Disk Administrator relating to the initial encryption process: 1. Administrator pre-install.

If the default user name and password is given for initial encryption, the system is considered to be in the pre-installed state. This option is intended for use by administrators to assist users. While the system is in pre-installed state, Authenti-Check configuration and the Access Recovery function are disabled until the user changes their password. The user is prompted at each application start to change their password until they choose a user name and password. After the user has changed their password, the Authenti-Check configuration screen is displayed to the user (if applicable). After configuration, the Authenti-Check Logon function is available. The Access Recovery function is also available after the user has changed their password from the default.

2. Encrypt only used space

   The EP Hard Disk Administrator has the option to encrypt only used space to speed up initial encryption. The factors the EP Hard Disk Administrator should consider in deciding whether or not to use this option are described in SO.USD_SPC.

3. Disable power-loss recovery

   The EP Hard Disk Administrator has the option to disable power-loss recovery during initial encryption to speed up the process. The factors the EP Hard Disk Administrator should consider in deciding whether or not to use this option are described in SO.PWR_LOS.

TSF.DSK_ENC — The TOE first authenticates the user and then makes encrypted partitions available by operating at the driver level so that user data is encrypted and decrypted on the fly as the operating system respectively writes and reads to the partition. The only TSF mediated function available before logon and successful authentication to the TOE is the recovery mechanism described in TSF.REC_SEC and TSF.KEY_REC. The Disk Key is used to encrypt the disk with AES in CBC mode, which conforms to the FIPS AES standard **[AES]**; CBC mode is specified in **[AES-MODES]**. Generation of the Disk Key is described in TSF.DSK_KEY. The Disk Key is stored encrypted under the Disk KEK; the Disk Key is encrypted with AES in CBC mode. The generation of the Disk KEK is described in TSF.DSK_KEK. The Disk decryption functionality is implemented as part of the On-the-Fly Encryption function of the User Program.

TSF.DSK_DEC	When the *allow user decrypt* attribute is set, the TOE allows authorized users to request decryption of partitions. The user will be considered authorized after having obtained access using a combination of authentication and recovery methods as documented in: TSF.USR_LOG, TSF.ATK_LOG, TSF.UAD_LOG, TSF.ADM_LOG and TSF.REC_SEC. Corporate and Local Administrators are allowed to request decryption of partitions whatever the setting of the *allow user decrypt* attribute. Disk Decryption is implemented as part of the Full Decryption function of the User Program.
TSF.DSK_KEY	The Disk Key is 256 bits and is generated using the cryptographic random number generator from version 0.9.6 of the openSSL library with MD5 replaced with SHA1 **[SHA1]**. The key size is 256 bits, one of the standard key sizes defined by the FIPS AES **[AES]** standard. The Disk Key is recovered by the key recovery method described in TSF.KEY_REC. The Disk Key generation is implemented as part of the Initial Encryption or Initial Encryption (Pre-installed) functions of the User Program.
TSF.DSK_KEK	The Disk Key Encryption Key (Disk KEK) is derived from the user password with the PBKDF2 key derivation function. The application of the PBKDF2 key derivation function conforms to the PKCS#5 standard **[PBKDF2]**. The Disk KEK derivation is implemented as part of the Initial Encryption or Initial Encryption (Pre-installed) functions of the User Program.
TSF.DB_KEY	The Administrator Database Key is generated using the cryptographic random number generator from version 0.9.6 of the openSSL library with MD5 replaced with SHA1 **[SHA1]**. The key size is 256 bits, one of the standard key sizes defined by the AES standard **[AES]**. The database key generation is implemented in the Administrator Program.
TSF.BAK_DBK	The TOE offers the EP Hard Disk Administrator the option of backing up the Administrator Database Key. The default location is the default removable storage device on the machine to encourage the administrator to store the key on removable media for physically secure storage. This function is implemented as part of the Administrator Program. Note: In this version of EP Hard Disk there is no function to make use of the backed up Administrator Database Key in event that the key is lost. This function may be added in a future version of EP Hard Disk.

TSF.DB_KEK	The Administrator Database Key Encryption Key (KEK) is derived from the EP Hard Disk Administrator password with the KDF2 key derivation function. The application of the KDF2 key derivation function conforms to the IEEE P1363a standard **[KDF2]**. The Administrator Database KEK derivation is implemented as part of the Administrator Program.
TSF.DB_ENC	The Administrator Database holds administrator private keys as a last line of defense against loss of availability in the event that an administrator leaves the company or forgets their password. The Administrator Database is encrypted with the Administrator Database Key using AES in CBC mode, which conforms to the FIPS AES standard **[AES]**. The generation of the AES Administrator Database Key is described in TSF.DB_KEY. The environmental security objective SO.BAK_DB gives details about precautions that the EP Hard Disk Administrator should take to ensure the continued availability of the Administrator Database. The Administrator Database Key is encrypted with the Administrator Database Key Encryption Key (KEK) using AES in CBC mode, which conforms to the AES standard **[AES]**. The derivation of the AES Administrator Database KEK is described in TSF.DB_KEK. The Administrator Database Key encrypted with the Administrator Database KEK is stored with the database. The Administrator Database encryption is implemented as part of the Administrator Program.
TSF.DSA_KP	The EP Hard Disk Administrator has an Elliptic Curve Digital Signature Algorithm (ECDSA) key pair: the private key is used by the administrator for signing update configuration messages; the public key is used by User Program installations to verify signatures on configuration update messages. The ECDSA private key is derived from the administrator password using the KDF2 key derivation function, which conforms to the IEEE P1363a standard **[KDF2]** standard. The ECDSA public key is computed from the private key as specified in the DSA standard **[ECDSA]** standard. The ECDSA key pair and the normal basis Koblitz Elliptic Curves used with it conform to the DSA standard **[ECDSA]** except in the respect that the private key is derived as described above. Old versions of the ECDSA private key are backed up in the encrypted Administrator Database described in TSF.DB_ENC. The ECDSA key pair generation is implemented as part of the Administrator Program.
TSF.DH_KP	Each of the Local Administrators and Corporate Administrator has an Elliptic Curve Diffie-Hellman (ECDH) key pair: the private key is used by the administrator for deriving the ECKRK that is sent to

the user in response to the recovery request messages in the EC-based access recovery procedure; the ECDH public key is used by the User Program to derive the ECKRK, which is used to encrypt the Disk Key in the recovery block and to create the negotiation component that will later be sent as the recovery request message. The ECDH private key is derived from the administrator password using the PBKDF2 key derivation function, which conforms to the PKCS#5 standard **[PBKDF2]**. The ECDH public key is computed from the private key as specified in the IEEE P1363 standard **[ECDH]**. The ECDH key pair and the normal basis Koblitz Elliptic Curves used were taken from the DSA standard **[ECDSA]**. The ECDH key pair generation is implemented as part of the Administrator Program.

6.2. TOE Assurance Measures

The following section first lists the assurance measures provided with the evaluation evidence and the product. The table tracing assurance measures to the assurance requirements laid out in Section 5.1.2 TOE Security Assurance Requirements is given in Section 7.4.3 Rationale AM to Security Assurance Requirements.

AM.ACM_CAP	The TOE is identified by a product name and a version number that are printed on the install media (where applicable), cited in the manuals (where applicable), displayed during the installation process, and are viewable on request in the application under the About menu item within the Administrator Program, the One-Time Password Program, and the User Program. The TOE identification is unique for each version of the TOE.
AM.USR_DOC	The User Program installation and start-up procedures are documented in the User Guide. The User Guide describes the steps necessary for secure installation, generation and start-up procedures. User guidance is given in the User Guide. The User Guide includes a description of the user accessible security functions provided by the TOE. The User Guide documents the security benefits of performing sensitive operations in a secure environment. The User Guide clearly presents all user responsibilities necessary for secure operation of the TOE, including assumptions regarding user behavior found in the statement of the TOE security environment. The User Guide is consistent with the Administrator Guide, this Security Target Document, and the Target of Evaluation submitted for evaluation. The User Guide describes all security requirements, security objectives for and assumptions about the IT environment that are relevant to the user.

AM.ADM_DOC The Administrator Program installation and start-up procedures are documented in the Administrator Guide. The Administrator Guide describes the steps necessary for secure installation, generation and start-up procedures for the Administrator Program. The Administrator Guide provides guidance addressed to administrator personnel about how to administer the TOE in a secure manner. The Administrator Guide describes the administrative functions and interfaces available to the administrator of the TOE. The Administrator Guide documents the security benefits of performing sensitive operations in a secure environment, particularly the assumption about secure backup and storage of the Administrator Database as described in non-IT security objective SO.BAK_DB. The Administrator Guide describes all assumptions regarding user behavior that are relevant to secure operation of the TOE. The Administrator Guide describes all security parameters under the control of the administrator and indicates secure values. The Administrator Guide describes each type of security relative event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF. The Administrator Guide is consistent with the User Guide, this Security Target Document and Target of Evaluation supplied for evaluation. The Administrator Guide describes all security requirements, security objectives for and assumptions about the IT environment that are relevant to the administrator.

AM.ADV_FSP The product description given in the Administrator and User Guides comprises the Informal Functional Specification for evaluation purposes. The Informal Functional Specification describes the TSF and its external interfaces. The Informal Functional Specification is consistent. The Informal Functional Specification describes the purpose and method of use of all external TSF interfaces, and provides details of the effects exceptions and error messages as applicable. The Informal Functional Specification completely represents the TSF.

AM.ADV_RCR Tracing from the TSFs to the Informal Functional Specification is given in section 7.4.3 Rationale AM to Security Assurance Requirements

The following table gives the tracing from Security Assurance Measures to Security Assurance Requirements.

Assurance Measure Requirement	Assurance Measure	Comments
ACM_CAP.1.1D ACM_CAP.1.1C	AM.ACM_CAP	The assurance measure meets all aspects of the assurance measure requirements.

ACM_CAP.1.2C		
ACM_CAP.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.1D ADO_IGS.1.1C	AM.USR_DOC AM.ADM_DOC	The assurance measures that are described in the AM.USR_DOC user documentation meet all aspects of the assurance measure requirements for the User Program. The assurance measures that are described in the AM.ADM_DOC administrator documentation meet all aspects of the assurance measure requirements for the Administrator Program and One-Time Password Program.
ADO_IGS.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.1D ADV_FSP.1.2C ADV_FSP.1.3C ADV_FSP.1.4C	AM.ADV_FSP	The assurance measure meets all aspects of the assurance measure requirements.
ADV_FSP.1.1E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_RCR.1.1D ADV_RCR.1.1C	AM.ADV_RCR	The assurance measure meets all aspects of the assurance measure requirements in showing a tracing from the TSFs to the Informal Functional Specification.
ADV_RCR.1.1E	Evaluation	To be evaluated by the evaluator.
AGD_ADM.1.1D AGD_ADM.1.2C AGD_ADM.1.3C AGD_ADM.1.4C AGD_ADM.1.5C AGD_ADM.1.6C AGD_ADM.1.7C AGD_ADM.1.8C	AM.ADM_DOC	The assurance measure meets all aspects of the assurance measure requirements.
AGD_ADM.1.1E	Evaluation	To be evaluated by the evaluator.
AGD_USR.1.1D AGD_USR.1.1C AGD_USR.1.2C AGD_USR.1.3C AGD_USR.1.4C AGD_USR.1.5C AGD_USR.1.6C	AM.USR_DOC	The assurance measure meets all aspects of the assurance measure requirements.
AGD_USR.1.1E	Evaluation	To be evaluated by the evaluator.
ATE_IND.1.1C	AM.ATE_IND	The assurance measure meets the assurance measure requirement.
ATE_IND.1.1E	Evaluation	To be evaluated by the evaluator.
ATE_IND.1.2E	Evaluation	To be evaluated by the evaluator.

Informal Functional Specification Rationale

6.3. Informal Functional Specification Rationale

The following table gives the tracing from Security Assurance Measures to Security Assurance Requirements.

Assurance Measure Requirement	Assurance Measure	Comments
ACM_CAP.1.1D ACM_CAP.1.1C ACM_CAP.1.2C	AM.ACM_CAP	The assurance measure meets all aspects of the assurance measure requirements.
ACM_CAP.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.1D ADO_IGS.1.1C	AM.USR_DOC AM.ADM_DOC	The assurance measures that are described in the AM.USR_DOC user documentation meet all aspects of the assurance measure requirements for the User Program. The assurance measures that are described in the AM.ADM_DOC administrator documentation meet all aspects of the assurance measure requirements for the Administrator Program and One-Time Password Program.
ADO_IGS.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.1D ADV_FSP.1.2C ADV_FSP.1.3C ADV_FSP.1.4C	AM.ADV_FSP	The assurance measure meets all aspects of the assurance measure requirements.
ADV_FSP.1.1E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_RCR.1.1D ADV_RCR.1.1C	AM.ADV_RCR	The assurance measure meets all aspects of the assurance measure requirements in showing a tracing from the TSFs to the Informal Functional Specification.
ADV_RCR.1.1E	Evaluation	To be evaluated by the evaluator.
AGD_ADM.1.1D AGD_ADM.1.2C AGD_ADM.1.3C AGD_ADM.1.4C AGD_ADM.1.5C AGD_ADM.1.6C AGD_ADM.1.7C AGD_ADM.1.8C	AM.ADM_DOC	The assurance measure meets all aspects of the assurance measure requirements.
AGD_ADM.1.1E	Evaluation.	To be evaluated by the evaluator.
AGD_USR.1.1D AGD_USR.1.1C AGD_USR.1.2C AGD_USR.1.3C AGD_USR.1.4C	AM.USR_DOC	The assurance measure meets all aspects of the assurance measure requirements.

AGD_USR.1.5C AGD_USR.1.6C		
AGD_USR.1.1E	Evaluation.	To be evaluated by the evaluator.
ATE_IND.1.1C	AM.ATE_IND	The assurance measure meets the assurance measure requirement.
ATE_IND.1.1E	Evaluation.	To be evaluated by the evaluator.
ATE_IND.1.2E	Evaluation	To be evaluated by the evaluator.

7. Rationale

7.1. Rationale Tracing Map

This section is not mandated by the CC, but is included to assist the reader in navigating and following the tracing in the following CC-required sections. The table in this section shows a high-level map of the tracing of the most significant functions of the TOE, focused around the main application functionality of Disk Encryption, Authenti-Check based recovery, Elliptic Curve-based recovery, the User Program Admin Logon function, and the Administrator Database encryption in that order. Other functionalities, non-direct threats, supporting security objectives, and TOE security functions are omitted for clarity.

The Security Functional Requirement tracing is omitted from this table for clarity as there are many SFRs, and their mapping is many-to-one from security objectives and many-to-one to TOE Security Functions, and they serve mainly to elaborate formally detailed requirements from the corresponding security objective and so are otherwise clearly grouped.

The full tracing organized as required by the CC are contained in the following sections.

Application Function	Threat / Assumption	Security Objective	TOE Security Function
Initial Encryption	T.DAT_SEC	SO.DAT_SEC	TSF.DSK_ENC TSF.DSK_KEY TSF.DSK_KEK
	T.USR_LOG	SO.USR_LOG	TSF.USR_LOG
On-the-Fly Encryption	T.DAT_SEC	SO.DAT_SEC	TSF.DSK_ENC TSF.DSK_KEY TSF.DSK_KEK
	T.USR_LOG	SO.USR_LOG	TSF.USR_LOG
User Program Admin Logon	T.UAD_LOG	SO.UAD_LOG	TSF.USR_LOG TSF.DSK_KEK
Authenti-Check based Recovery	T.PAS_LOS	SO.DAT_AVA SO.ATK_SEC	TSF.ATK_REC TSF.ATK_KRK TSF.DSK_KEY
	T.ATK_LOG	SO.ATK_LOG	TSF.ATK_LOG TSF.ATK_KRK

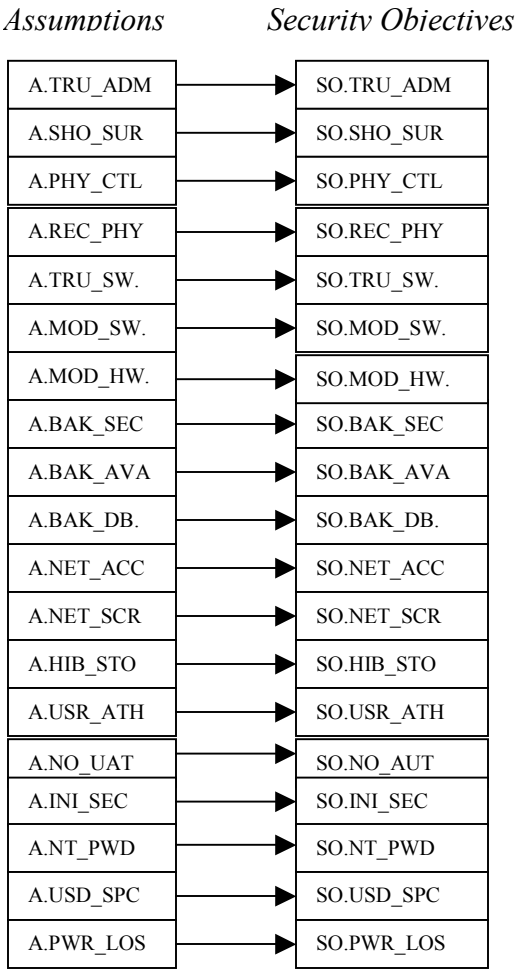
Elliptic Curve-based Recovery	T.PAS_LOS	SO.DAT_AVA SO.REC_SEC	TSF.REC_SEC TSF.KEY_REC TSF.REC_BLK TSF.DH_KP TSF.EC_KRK TSF.DSK_KEY
	A.USR_ATH	SO.USR_ATH	Not Included (Non-IT security objective)
Administrator Database Encryption	T.DB_SEC	SO.DB_ENC	TSF.DB_ENC TSF.DB_KEY TSF.DB_KEK TSF.BAK_DBK
	T.ADM_LOG	SO.ADM_LOG	TSF.ADM_LOG TSF.DB_KEK
Signed Administrator Configuration Update Messages	T.UPD_MOD	SO.UPD_ATH	TSF.ADM_CFG TSF.DSA_KP
	T.ADM_LOG	SO.ADM_LOG	TSF.ADM_LOG TSF.DSA_KP
Administrator Program	T.ADM_LOG	SO.ADM_LOG	TSF.ADM_LOG TSF.DH_KP

The diagrams below show the tracing from threats and assumptions to security objectives, from security objectives to TOE security functions, and from TOE security functions to TOE Summary Specifications.

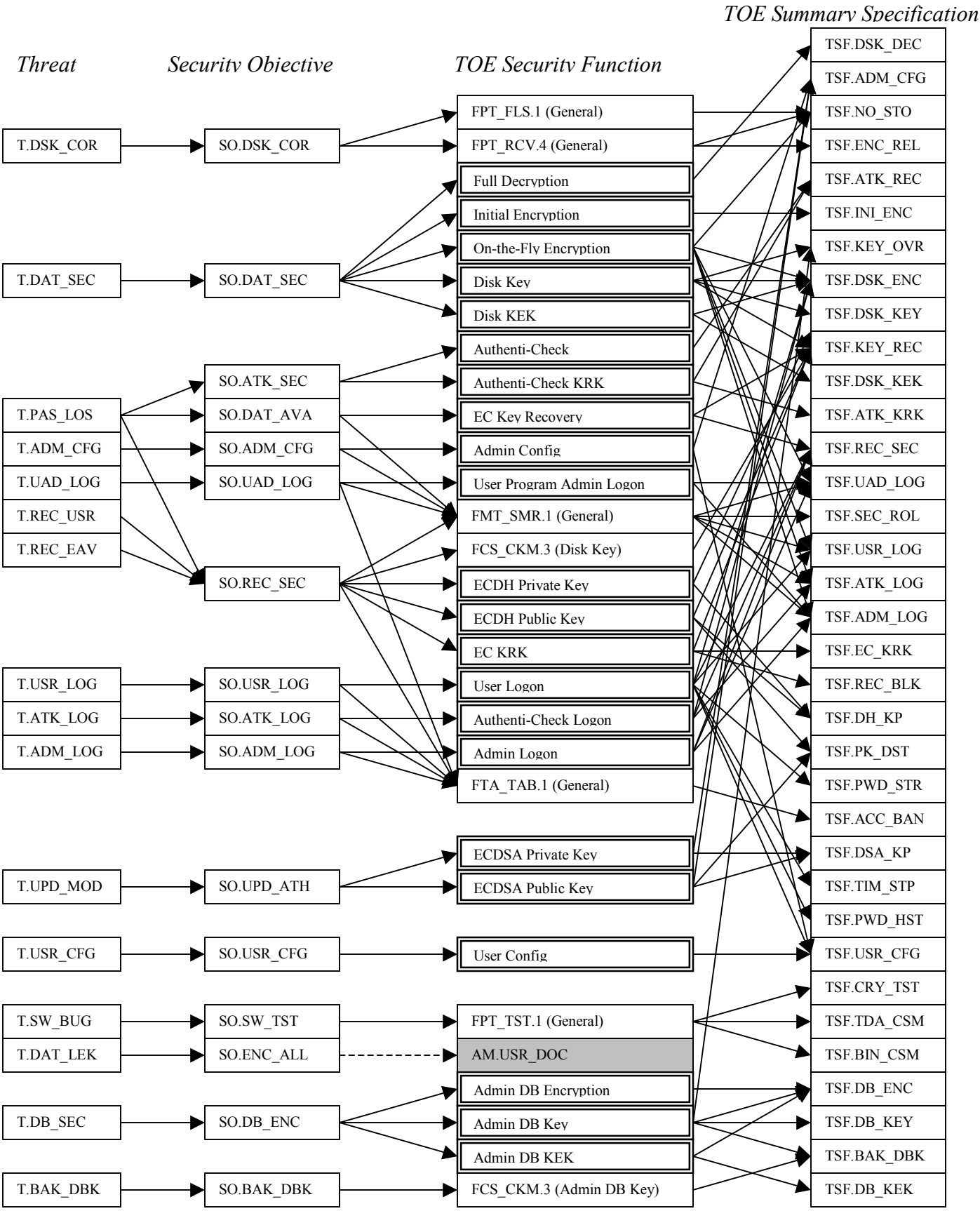
The diagrams serve mainly to allow the evaluator to more easily visually verify that:

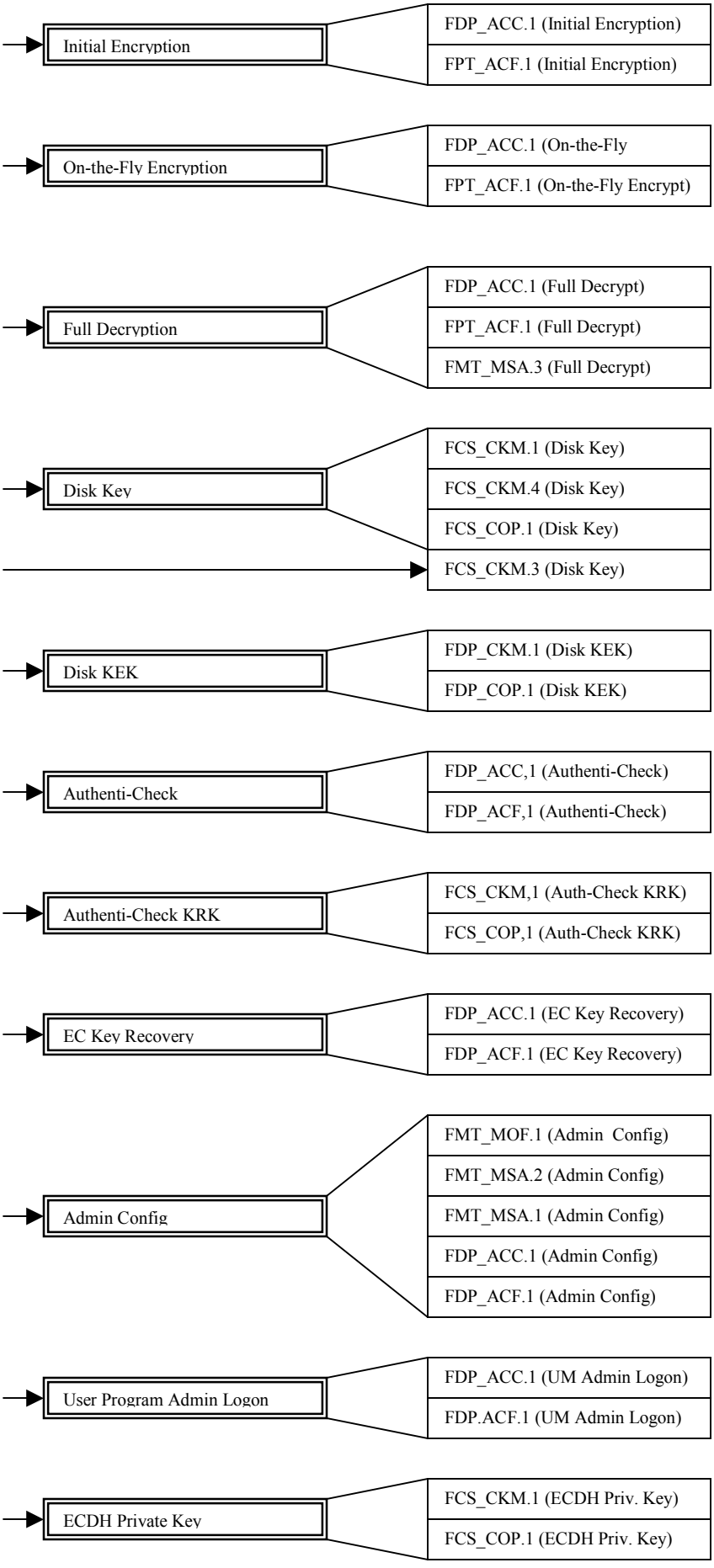
1. All assumptions and threats trace to one or more security objectives, and that all security objectives satisfy one or more threats or assumptions; and
2. All security objectives trace one or more TOE Security Functions, and that all TOE Security Functions satisfy one or more security objectives; and
3. All TOE Security Functions trace to one or more TOE Summary Specifications, and that all TOE Summary Specifications trace to one or more TOE Security Functions.

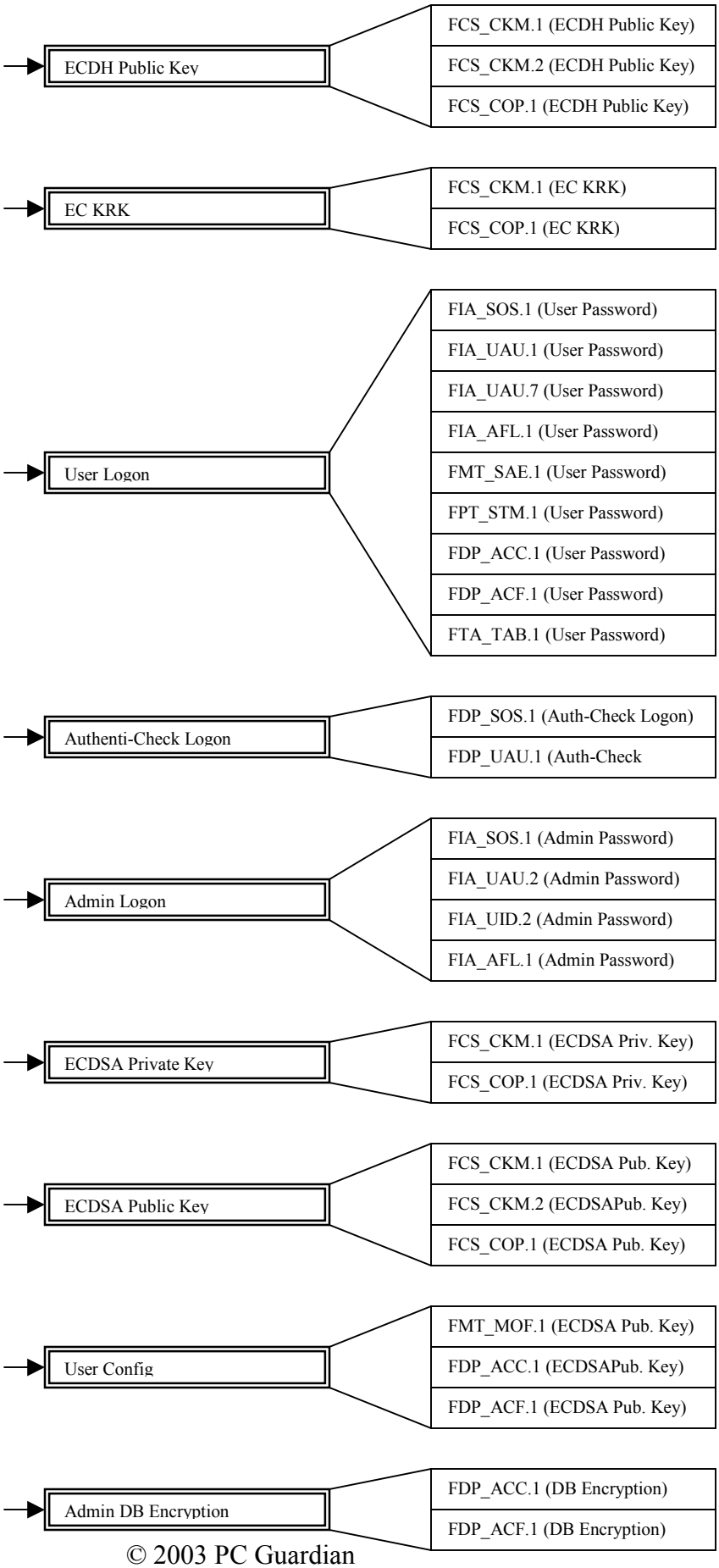
This tracing is not readily observable from the tables in the CC-required following sections due to the number of elements. The Security Objectives derived from Assumptions are not mapped to TOE Security Functions (or TOE Summary Specifications).

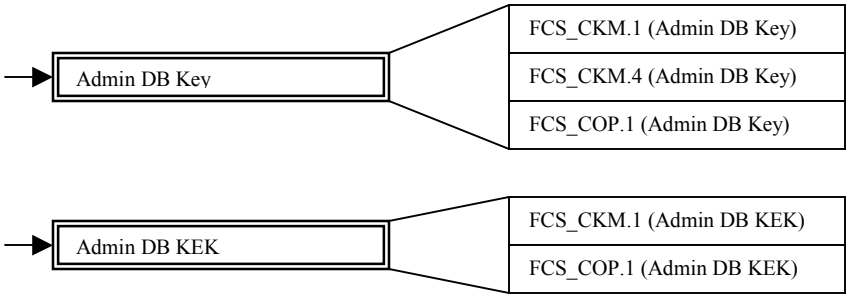


In the following diagram, boxes with a double border represent groups of TOE Security Functions that are further elaborated on the following page. The TOE Security Functions represented in groups are grouped as in Section 5. Where some TOE Security Functions have been separated from the rest of the grouping where it has tracing to TOE Summary Specifications unrelated to the rest of the group, the separated TOE Security Functions are shown diagrammatically associated with the main group in the elaboration of the group. For TOE Security Functions that are grouped in the diagram, the tracing is done only from TOE Security Function Group to TOE Summary Specification. Refer to the CC-required mapping later in this section for the full mapping from individual TOE Security Functions to TOE Summary Specifications. Boxes shown shaded gray are not implemented and further notes are given on them in the CC-mandated tracing in the following sections.




Initial Encryption
FDP_ACC.1 (Initial Encryption)
FPT_ACF.1 (Initial Encryption)
On-the-Fly Encryption
FDP_ACC.1 (On-the-Fly
FPT_ACF.1 (On-the-Fly Encrypt)
Full Decryption
FDP_ACC.1 (Full Decrypt)
FPT_ACF.1 (Full Decrypt)
FMT_MSA.3 (Full Decrypt)
Disk Key
FCS_CKM.1 (Disk Key)
FCS_CKM.4 (Disk Key)
FCS_COP.1 (Disk Key)
FCS_CKM.3 (Disk Key)
Disk KEK
FDP_CKM.1 (Disk KEK)
FDP_COP.1 (Disk KEK)
Authenti-Check
FDP_ACC,1 (Authenti-Check)
FDP_ACF,1 (Authenti-Check)
Authenti-Check KRK
FCS_CKM,1 (Auth-Check KRK)
FCS_COP,1 (Auth-Check KRK)
EC Key Recovery
FDP_ACC.1 (EC Key Recovery)
FDP_ACF.1 (EC Key Recovery)
Admin Config
FMT_MOF.1 (Admin Config)
FMT_MSA.2 (Admin Config)
FMT_MSA.1 (Admin Config)
FDP_ACC.1 (Admin Config)
FDP_ACF.1 (Admin Config)
User Program Admin Logon
FDP_ACC.1 (UM Admin Logon)
FDP.ACF.1 (UM Admin Logon)
ECDH Private Key
FCS_CKM.1 (ECDH Priv. Key)
FCS_COP.1 (ECDH Priv. Key)





7.2. Security Objectives Rationale

7.2.1. Security Objectives Rationale for Assumptions

Security Assumptions	Security Objective	Comments on rationale for tracing and coverage
A.TRU_ADM	SO.TRU_ADM	The security objective mirrors the assumption about the security value of having trusted administrators.
A.SHO_SUR	SO.SHO_SUR	The security objective discusses the objective of it not being possible for threat agents to observe users typing their passwords to counter the threat of this occurring described in the assumption. The security objective also discusses the implication that if closed circuit TV systems are employed and passwords hence viewable that the personnel operating the CCTV system should be added to the set of trusted personnel described under SO.TRU_ADM.
A.PHY_CTL	SO.PHY_CTL	The security objective mirrors the assumptions about the security value of ensuring the machine the User Program is installed on does not come under temporary and undetected physical control of a threat agent.
A.REC_PHY	SO.REC_PHY	The security objective mirrors the assumptions about the security imperative of ensuring the machine the One-Time Password Program is installed on does not come under physical control of a threat agent.
A.TRU_SW.	SO.TRU_SW.	The security objective mirrors the assumptions about the security value of running only trusted software approved by the security officer, and of using operating system features where available to restrict user ability to install software.
A.MOD_SW.	SO.MOD_SW.	The security objective describes and gives examples of counter-measures that can be taken to help detect software modification as identified

		in the assumption.
A.MOD_HW.	SO.MOD_HW	The security objective mirrors the assumption about risks of hardware modification. The assumption is that the threat agent has insufficient resources and expertise to carry out the hardware modification attack the assumption talks about. Note: There is no countermeasure proposed for the assumption.
A.BAK_SEC	SO.BAK_SEC	The security objective mirrors the assumption about taking appropriate measures to physically secure and/or encrypt backups of user data.
A.BAK_AVA	SO.BAK_AVA	The security objective mirrors the assumption about taking regular and complete backups of user data to ensure data availability in the face of equipment theft or destruction.
A.BAK_DB.	SO.BAK_DB.	The security objective refines the statement about the value of ensuring the continued availability of the Administrator Database described in the assumption.
A.NET_ACC	SO.NET_ACC	The security objective mirrors the statement about the security implications of connecting the protected machine to the network and enabling network services.
A.NET_SCR	SO.NET_SCR	The security objective mirrors the statement in the assumption about the security implications of using network logon scripts or other mechanisms involving automatic execution of remotely downloaded software.
A.HIB_STO	SO.HIB_STO	The security objective mirrors the statement in the assumption about the security value of disabling laptop hibernation features.
A.USR_ATH	SO.USR_ATH	The security objective mirrors the statement in the assumption about the importance of authenticating users with a security officer-approved authentication mechanism before the administrator proceeds with the access recovery procedure.
A.NO_UAT	SO.NO_UAT	The security objective mirrors the statement about the importance of not leaving the software unattended in a logged on state.
A.INI_SEC	SO.INI_SEC	The security objective mirrors the statement about the importance of taking physical security precautions with machines in the pre-installed state before the user has changed the password.
A.NT_PWD	SO.NT_PWD	The security objective advises administrators to

		evaluate whether the security risks identified in the assumption relating to use of the Windows password synchronization applies to their installation, and if so whether they are satisfied with the security afforded by their server configuration.
A.USD_SPC	SO.USD_SPC	The security objective advises administrators to only allow the *encrypt used space only* option for initial encryption where there is no sensitive information on the disk at the time of encryption.
A.PWR_LOS	SO.PWR_LOS	The security objective advises administrators not to use the *encrypt without power-loss protection* option unless there is reliable power source and no user data that is not backed-up on the disk.

7.2.2. Security Objectives Rationale for Threats

Security Threats	Security Objective	Comments on rationale for tracing and coverage
T.PAS_LOS	SO.DAT_AVA SO.REC_SEC SO.ATK_SEC	The high-level policy security objective of ensuring data availability is described in SO.DAT_AVA. The threat of password loss is countered by two technical alternative security objectives: SO.REC_SEC describes procedure to recover access allowing the user to regain access and chose a new password with the assistance of an administrator. SO.ATK_SEC describes an alternative recovery procedure to allow the user to regain access and chose a new password.
T.DSK_COR	SO.DSK_COR	The security objective describes how the objective of retaining a consistent state at all times aids the reliability of the product in the face of the threat of disk corruption due to mechanical failure or unclean operating system shutdown due to power failure.
T.DAT_SEC	SO.DAT_SEC	The security objective counters the aspect of the threat related to a threat agent recovering data from the machine by examining data stored on the disk. The threat is countered by ensuring that all user data stored in protected folders is encrypted on the disk.
T.USR_LOG	SO.USR_LOG	The security objective counters the threat of a threat agent attempting to abuse the logon security function by ensuring that only authorized users can successfully gain access to the TOE

		with the logon function.
T.UAD_LOG	SO.UAD_LOG	The security objective counters the threat of a threat agent attempting to abuse the User Program Admin Logon function by ensuring that only authorized administrators can successfully gain access to the TOE with the admin logon function.
T.ADM_LOG	SO.ADM_LOG	The security objective addresses the threat of a threat agent abusing the Administrator Program by providing a secure logon process where only authorized administrators can gain access to the Administrator Program.
T.REC_USR	SO.REC_SEC	The threat describes how one user may masquerade as another target user to the administrator in order to attempt to abuse the access recovery procedure to gain access to the target user's data. The security objective aspect that counters this threat is the objective that only the authorized user whose data is protected should be able to successfully use the access recovery procedure.
T.REC_EAV	SO.REC_SEC	The threat describes how a threat agent may try to use previous recovery request and response messages captured by eavesdropping on the user and administrator executing the recovery procedure to abuse the recovery procedure on a subsequently stolen machine. The security objective counters this threat by having the objective that possession of previous messages does not assist the threat agent in gaining access to the machine.
T.ATK_LOG	SO.ATK_LOG	The threat describes how a threat agent may try to abuse the Authenti-Check logon procedure. The security objective counters this threat by having the objective that the Authenti-Check logon procedure be secure.
T.UPD_MOD	SO.UPD_ATH	The threat describes how a threat agent may try to modify configuration update messages to weaken or otherwise compromise the security of the TOE; the security objective describes how this threat is countered by the administrator signing the configuration update messages.
T.ADM_CFG	SO.ADM_CFG	The security objective addresses the threat of the administrator unintentionally choosing insecure configuration values by preventing selection of insecure values.
T.USR_CFG	SO.USR_CFG	The security objective addresses the threat of the

		user unintentionally choosing insecure configuration values by preventing selection of insecure values. The default values, and restrictions should be configurable by the administrator to suit the environment and organizations policies.
T.SW_BUG	SO.SW_TST	The security objective addresses the threat of software bug by performing self-tests at startup on sensitive operations.
T.DAT_LEK	SO.ENC_ALL	The security objective addresses the threat of sensitive user data leaking by being accidentally written to unprotected partitions by encouraging the user to protect all partitions on the machine.
T.DB_SEC	SO.DB_ENC	The security objective addresses the threat of the sensitive administrator password and key related information stored in the Administrator Database being obtained by a threat agent by encrypting the database.
T.BAK_DBK	SO.BAK_DBK	The security objective addresses the threat of loss of availability of the Administrator Program functions and need for security in the backup of the Administrator Database key.

7.2.3. SO Rationale for Security Assurance Requirements

There is no table showing a mapping from Security Objectives to Security Assurance Requirements as the components do not map at a detailed level as the Security Assurance Requirements are more about documentation than individual requirements. However the security objectives collectively map to the assurance requirement ADV_FSP.1 in the sense that they are the objectives implemented by the TSS which implement the functional specification. The other assurance requirements are about documentation provided in the Administrator and User Guides and in this document.

EAL1 was chosen as appropriate to the security needs of customers the TOE is used by and will be marketed to.

As appropriate for selection of EAL1 for the expected uses of the TOE, some confidence in correct operation is required, but the threats to security are not viewed as serious. Independent assurance is required to support the contention that due care has been exercised with respect to the protection of personal or similar information.

7.3. Security Requirements Rationale

The rationale for Security Functional Requirements (SFRs) with tracing to TOE security objectives is given in the table below. The Security Functional Requirements are categorized by application functionality, because in many instances the same SFR component has been used multiple times to cover different application functionalities. To disambiguate which of the multiple uses of each Security Functional Requirement

component is referred to, and for clarity, the application function is included in (brackets) after the component short name. The application function break down used matches that used in Section 5.1.1 TOE Security Functional Requirements.

The table is ordering by security objective, with security objectives given the same ordering as their presentation in Section 4.1 Security Objectives for the TOE.

Security Objective	Security Functional Requirement	Comments on rationale for tracing and coverage
SO.DAT_AVA	FDP_ACC.1 (EC key recovery) FDP_ACF.1 (EC key recovery)	The requirements express the policy corresponding to high-level intent of the security objective of providing key recovery to recover from user forgetting their password: the key recovery procedure access to user data encrypted on a user disk should be granted when an administrator and the user collaborate to allow the user to re-gain access and choose a new password.
	FMT_SMR.1 (General)	The policy requires the definition of the security roles of user, EP Hard Disk Administrator, Corporate Administrator, and Local Administrator to express.
SO.DSK_COR	FPT_FLS.1 (General)	The requirement meets the security objective by stating more formally the failure events that must preserve a secure and consistent state.
	FPT_RCV.4 (General)	The requirement meets the security objective by stating more formally the requirements for recovery from failure to a secure and consistent state.
SO.DAT_SEC	FDP_ACC.1 (Initial Encryption) FDP_ACF.1 (Initial Encryption)	This pair of SFRs state describe some access control policies relating to the initial encryption of the partition.
	FDP_ACC.1 (On-the-Fly Encrypt) FDP_ACF.1 (On-the-Fly Encrypt)	This pair of SFRs state the high-level intent of the security objective as a policy requiring authentication from the user, a Local Administrator, or Corporate Administrator before access to user data is granted.
	FDP_ACC.1 (Full Decrypt) FDP_ACF.1 (Full Decrypt) FMT_MSA.3 (Full Decrypt)	This set of SFRs describe some access control policies relating to decrypting encrypted data for a partition.

	FCS_CKM.1 (Disk Key) FCS_CKM.4 (Disk Key) FCS_COP.1 (Disk Key)	The security objective is to ensure that a threat agent should not be able to recover user data by examining data stored on the disk. This group of SFRs give requirements for the aspect of the security objective related to encryption of user data on the disk and the handling of the Disk Key that is directly used to encrypt the disk. FCS_CKM.1 gives requirements for the generation of the Disk Key. FCS_CKM.4 gives requirements for key destruction method. FCS_COP.1 gives requirements for the data encryption operation for encrypting user data.
	FCS_CKM.1 (Disk KEK) FCS_COP.1 (Disk KEK)	This group of SFRs give detailed requirements for usage of the Disk KEK that is used to protect the Disk Key. This group of SFRs indirectly supports the security objective as it supports the security of handling of the Disk Key used in the above group of SFRs to directly meet the security objective. FCS_CKM.1 gives requirements for how the Disk KEK should be derived from the user password. FCS_COP.1 gives requirements for how the Disk KEK should be used to encrypt the Disk Key.

SO.USR_LOG	FIA_SOS.1 (User Password) FIA_UAU.1 (User Password) FIA_UAU.7 (User Password) FIA_AFL.1 (User Password) FMT_SAE.1 (User Password) FPT_STM.1 (User Password) FDP_ACC.1 (User Password) FDP_ACF.1 (User Password)	This group of SFRs meets the security objective of providing a secure logon function by giving detailed requirements about restrictions on the choice of password and handling of entry of the user password. FIA_SOS.1 gives requirements about restrictions on choice of user password. FIA_UAU.1 gives an exception to the general requirement that users be authenticated before access is granted to any TSF mediated function. The exception allows access to the access recovery procedure in the event that the user has forgotten their password. FIA_UAU.7 gives requirements about how the user password is displayed as it is typed. FIA_AFL.1 gives requirements about actions to be taken if the user enters their password incorrectly an administrator-configured number of times. FMT_SAE.1 gives the requirements for user password expiry and the restriction that only the EP Hard Disk Administrator can set the expiry timeout value. FPT_STM.1 gives the requirement that the implementation have reliable time-stamps in ordered to implement the password expiry requirement given in FMT_SAE.1 FDP_ACC.1 and FDP_ACF.1 give the requirements for a password history function where users are prevented from choosing the same password again.

	FTA_TAB.1 (General)	The requirement FTA_TAB.1 contributes to the security objective in that it requires information to be provided at logon about unauthorized use of the application. Note: Some of the password-related requirements involve user and administrator configurable parameters. See: SO.USR_CFG and SO.ADM_CFG respectively for requirements about application configuration relating to user logon and password entry.
SO.ADM_LOG	FIA_SOS.1 (Admin Password) FIA_UAU.2 (Admin Password) FIA_UID.2 (Admin Password) FIA_AFL.1 (Admin Password)	This group of SFRs meets the security objective of providing a secure logon function for the administrator for logon to the Admin Logon function of the User Program and for logon to the Administrator Program by giving detailed requirements about administrator authentication. FIA_SOS.1 gives requirements about minimum requirements for choice of administrator password that should be enforced by the TOE. FIA_UAU.1 gives an exception to the general policy that no TSF mediated functions of the User Program should be available to the administrator prior to administrator authentication. The exception is to allow the administrator to optionally use the access recovery procedure on behalf of the user before the administrator is authenticated. FIA_UAU.2 gives the requirement that the administrator must be logged before the protected functions are made available. FIA_AFL.1 requires that after the administrator has entered their password incorrectly a maximum number of times that the application should lock. (The administrator can try again by rebooting the machine.)
	FTA_TAB.1 (General)	The requirement FTA_TAB.1 contributes to the security objective in that it requires information to be provided at logon about unauthorized use of the application.

SO.UAD_LOG	FDP_ACC.1 (ED Admin Logon) FDP_ACF.1 (ED Admin Logon)	The requirements express the policy corresponding to the high-level intent of the security objective of providing Corporate and Local Administrators with access to user data that should only be available to authenticated administrators. The technical aspects of gaining access are described under SO.REC
	FMT_SMR.1 (General)	The policy requires the definition of the security roles of user, Corporate Administrator, and Local Administrator to express.
	FTA_TAB.1 (General)	The requirement FTA_TAB.1 contributes to the security objective in that it requires information to be provided at logon about unauthorized use of the application.
SO.REC_SEC	FCS_CKM.3 (Disk Key)	The overall set of SFRs grouped with SO.REC_SEC meet the security objective by providing requirements for the secure recovery mechanism used to implement the access control policy given in SO.DAT_AVA: to allow the user, with assistance from the administrator, to recover from forgetting their password. The SFR CKM.3 relates to the mechanisms used to provide recovery of the Disk Key and states that the key recovery mechanism adheres to no standard.
	FCS_CKM.1 (ECDH priv. key) FCS_COP.1 (ECDH priv. key)	This group of SFRs gives the requirements for the generation and handling of the ECDH private key used in the recovery mechanism. FCS_CKM.1 gives the requirements for the derivation of the ECDH private key from the administrator password. FCS_COP.1 gives the requirements for the cryptographic operations the ECDH private key is used by to effect the recovery mechanism.

	FCS_CKM.1 (ECDH public key) FCS_CKM.2 (ECDH public key) FCS_COP.1 (ECDH public key)	This group of SFRs gives the requirements for the generation and handling of the ECDH public key used in the recovery mechanism. FCS_CKM.1 gives the requirements for the derivation of the ECDH public key from the ECDH private key. FCS_CKM.2 gives the requirements for the distribution of the ECDH public key to send the original key and ECDH public key updates to the users. FCS_COP.1 gives the requirements for the use of the ECDH public key to encrypt the information in the recovery block.
	FCS_CKM.1 (ECKRK) FCS_COP.1 (ECKRK)	This group of SFRs gives the requirements for the generation and handling of the Elliptic Curve Key Recovery Key (ECKRK) used to decrypt the recovery block. FCS_CKM.1 gives the requirements for the derivation of the ECKRK from the administrator private key and the user name of the user being recovered. FCS_COP.1 gives the requirements for the cryptographic operations the ECKRK is used by to decrypt the recovery block.
	FMT_SMR.1 (General)	The policy requires the definition of the security roles of user, Corporate Administrator, and Local Administrator to express.
	FTA_TAB.1 (General)	The requirement FTA_TAB.1 contributes to the security objective in that it requires information to be provided at logon about unauthorized use of the application.

SO.ATK_LOG	FIA_SOS.1 (Authenti-Check) FIA_UAU.1 (Authenti-Check)	This group of SFRs meets the security objective of providing an alternate question and answer based secure logon function to enable recovery of access in the event that the user forgets their password. FIA_SOS.1 expresses a policy about enforcement of minimum length requirements for secrets. FIA_UAU.1 expresses a policy about the timing of use of the Authentic-Check logon function.
SO.ATK_KRK	FDP_ACC.1 (Authenti-Check) FDP_ACF.1 (Authenti-Check)	This group of SFRs give the requirements for authentication of the user via the Authenti-Check mechanism.
	FTA_TAB.1 (General)	The requirement FTA_TAB.1 contributes to the security objective in that it requires information to be provided at logon about unauthorized use of the application.
SO.ATK_SEC	FDP_ACC.1 (Authenti-Check) FDP_ACF.1 (Authenti-Check)	The overall set of SFRs grouped with SO.ATK_SEC meet the security objective of providing a secure Authenti-Check mechanism by providing requirements for this mechanism. This group of SFRs give the high-level intent policy requirements of the Authenti-Check mechanism.
	FCS_CKM.1 (Auth-Check KRK) FCS_COP.1 (Auth-Check KRK)	This group of SFRs meet the security objective of providing a secure Authenti-Check mechanism by providing requirements the generation and use of the Authenti-Check Key Recovery Key (KRK). FCS_CKM.1 gives the requirements for the derivation of the Authenti-Check KRK from the users answers to the Authenti-Check questions and answers they selected at installation. FCS_COP.1 gives the requirements for the cryptographic operations the Authenti-Check KRK is used by to decrypt the encrypted Disk Key stored with the recovery block.

SO.UPD_ATH	FCS_CKM.1 (ECDSA priv. key) FCS_COP.1 (ECDSA priv. key) FCS_CKM.3 (ECDSA priv.key)	This overall set of SFRs grouped with SO.UPD_ATH meet the security objective by providing requirements for the signed administrator configuration update message mechanism and requirements for the keys used. The high-level policy aspects of this mechanism are described under SO.ADM_CFG. FCS_CKM.1 gives the requirements for the derivation of the ECDSA private key from the EP Hard Disk Administrator password. FCS_CKM.3 gives the requirements for the backup of the ECDSA private key. FCS_COP.1 gives the requirements for the cryptographic operations the ECDSA private key is used by to create signatures on administrator configuration update messages.
	FCS_CKM.1 (ECDSA pub. key) FCS_CKM.2 (ECDSA pub. key) FCS_COP.1 (ECDSA pub. key)	This group of SFRs gives the requirements for the verification of signatures on administrator configuration update messages by the User Program on behalf of the user. FCS_CKM.1 gives the requirements for the derivation of the ECDSA public key from the ECDSA private key. FCS_CKM.2 gives the requirements for the distribution of the ECDSA public key to send the original key and ECDSA public key updates in the configuration update message mechanism. FCS_COP.1 gives the requirements for the cryptographic operations the ECDSA public key is used by to verify signatures on administrator configuration update messages in the User Program on behalf of the user.

SO.ADM_CFG	FMT_MOF.1 (Admin Config) FMT_MSA.2 (Admin Config) FMT_MSA.1 (Admin Config) FDP_ACC.1 (Admin Config) FDP_ACF.1 (Admin Config)	This group of SFRs expresses the high-level intent policy statement corresponding to the security objective of ensuring that only the administrator can set and modify administrator configuration values. By implication, this is also technically achieved by SO.UPD_ATH that describes how the administrator configuration update messages are signed. FMT_MOF.1 gives the requirements for the high-level policy that only the EP Hard Disk Administrator can modify the configuration parameters. FMT_MSA.2, FMT_MSA.1, FDP_ACC.1 and FDP_ACF give the requirements that the TOE ensure only secure values for configuration values are selectable.
	FMT_SMR.1 (General)	The policy requires the definition of the security roles of user, EP Hard Disk Administrator, Corporate Administrator, and Local Administrator to express.
SO.USR_CFG	FMT_MOF.1 (User Config) FDP_ACC.1 (User Config) FDP_ACF.1 (User Config)	This group of SFRs expresses the high-level intent policy statement corresponding to the security objective of ensuring that only the authorized user or an administrator can modify the user configuration settings. FMT_MOF.1 gives the requirements for the high-level policy that only the authorized user or an administrator can modify the user configuration settings. FDP_ACC.1, and FDP_ACF.1 give the requirements that the TOE ensure that the user is unable to select settings outside the range of allowed values configured by the EP Hard Disk Administrator or default ranges supplied with the TOE.
SO.SW_TST	FPT_TST.1 (General)	The SFR gives requirements for the security objective of having a test function to test software reliability.

SO.ENC_ALL	Not Included. See: AM.USR_DOC (Assurance Measure)	The objective of encouraging the user to encrypt all partitions is not implemented in the TOE. It is however documented in the User Guide, so is covered (by documentation only) by AM.USR_DOC.
SO.DB_ENC	FDP_ACC.1 (DB Encryption) FDP_ACF.1 (DB Encryption)	The SFRs give the high-level policy requirements for implementing the security objective of only allowing the EP Hard Disk Administrator after successful authentication to access the Administrator Database.
	FCS_CKM.1 (Admin DB Key) FCS_CKM.4 (Admin DB Key) FCS_COP.1 (Admin DB Key)	This group of SFRs meet the security objective of protecting the data stored in the Administrator Database. FCS_CKM.1 gives the requirements for generation of the Administrator Database Key. FCS_CKM.4 gives requirements for key destruction method. FCS_COP.1 gives the requirements for the encryption operations that will be used with the Administrator Database Key.
	FCS_CKM.1 (Admin DB KEK) FCS_COP.1 (Admin DB KEK)	This group of SFRs meet the security objective of protecting the Administrator Database Key. FCS_CKM.1 gives the requirements for the derivation of the Administrator Database KEK from the EP Hard Disk Administrator password. FCS_COP.1 gives the requirements for the operations that will be used with the Administrator Database KEK to encrypt the Administrator Database Key.
SO.BAK_DBK	FCS_CKM.3 (Admin DB Key)	This SFR meets the security objective of providing continued availability of the Administrator Database key in the event that the EP Hard Disk Administrator forgets his password or leaves the organization without revealing his password.

7.4. TOE Summary Specification Rationale

7.4.1. Rationale Introduction

In the following section, the SFR requirements are traced to the TSF functions that implement them, and notes are given providing a rationale for the coverage provided. The

tracing and rationale are organized as tables, and are grouped by application functionality.

7.4.2. Rationale by Application Function

The breakdown of the following validation tables matches the breakdown of SFRs used in Section 5.1.1TOE Security Functional Requirements.

7.4.2.1. General Application Functionality

SFR	TSF	Rationale
FPT_FLS.1.1	TSF.NO_STO	The TOE does not store sensitive information to disk at any time, as a result if the machine were to suffer hardware failure, or loss of power at any time this will have no adverse security implications; in particular when power is restored, or the failure recovered, the machine will be in a secure state.
FPT_RCV.4.1	TSF.NO_STO	The non-storage of sensitive information meets the security requirement.
	TSF.ENC_REL	The reliability functionality meets the requirements for consistency of state on recovery from failure.
FMT_SMR.1.1	TSF.SEC_ROL	Required roles are implemented.
FMT_SMR.1.2	TSF.USR_LOG TSF.ATK_LOG	The user is authenticated to the user role by the logon function and Authenti-Check Logon function, so the TSF is able to associate users with the user role. Note: For the access recovery procedure, the non-IT security objective documented in SO.USR_ATH that a security officer approved authentication method is used is relevant.
	TSF.ADM_LOG TSF.UAD_LOG	The Corporate Administrator and Local Administrator are authenticated to their respective roles by the Administrator Logon function and User Program Admin Logon function, so the TSF is able to associate administrators to their roles.
FPT_TST.1.1	TSF.CRY_TST	The cryptographic self-tests are the only aspects of the TOE subject to self-tests. This TSF implements the cryptographic self-tests.
FPT_TST.1.2	TSF.TDA_CSM	The TSF implements check-sums and modification detection codes on TSF data meeting the requirement. Note: There is no function the user can invoke explicitly, but TSF data integrity errors are reported if they occur.

FPT_TST.1.3	TSF.BIN_CSM	The TSF implements checksums on its EPOS (Pre-Dos) applications, drivers and libraries meeting the integrity verification requirements. Note: There is no function the user can invoke explicitly, but application integrity is tested at startup, and any errors are reported. Note: The windows level executables are not check-summed.
FTA_TAB.1.1	TSF.ACC_BAN	The TSF implements the required access banner.

7.4.2.2. Initial Encryption

This section, for clarity, groups the rationale tracing for the security requirements relating to initial encryption to associated TSFs.

SFR	TSF	Rationale
FDP_ACC.1	TSF.INI_ENC	The TSF implements the initial encryption of disk partitions and the pre-install option.
FPT_ACF.1		

7.4.2.3. On-the-Fly Encryption

This section, for clarity, groups the rationale tracing for the security requirements relating to on-the-fly encryption to associated TSFs.

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.NO._STO TSF.DSK_ENC TSF.USR_LOG TSF.ATK_LOG TSF.UAD_LOG TSF.ADM_LOG TSF.REC_SEC	The TSFs ensure that no sensitive data is written to disk in unencrypted form, that all user data stored on protected partitions is encrypted, and that all means to obtain access to the keys and hence user data are authenticated (TSF.USR_LOG user logon, TSF.ATK_LOG Authenti-Check logon, TSF.UAD_LOG User Program Admin Logon function for use by an administrator, and TSF.REC_SEC Access Recovery function of the One-Time Password Program, and it is assumed that the user is authenticated to the administrator using an approved authentication method described in non-IT security objective SO.USR_ATH).
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.4. Full Decryption

This section for clarity groups the rationale tracing for the security requirements relating to full decryption to associated TSFs.

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.DSK_DEC	The TSFs ensure that only a user authorized to read the data in unencrypted form and allowed to decrypt the partition is able to decrypt the partition. Authorization is provided as all means to obtain access to the keys and hence user data are authenticated (TSF.USR_LOG user logon, TSF.ATK_LOG Authenti-Check logon, and TSF.REC_SEC Access Recovery function of the One-Time Password Program, and it is assumed that the user is authenticated to the administrator using an approved authentication method described in non-IT security objective SO.USR_ATH). Administrator access is documented separately under FDP_ACF.1.3 below.
FDP_ACF.1.3	TSF.UAD_LOG TSF.DSK_DEC	The TSFs ensure that the users role is verified as being an Administrator, using TSF.UAD_LOG User Program Admin Logon function for use by an administrator, and this role is used in the DSK_DEC function to allow decryption by an administrator.

FDP_ACF.1.4	N/A	Empty requirement.
FMT_MSA.3.1 FMT_MSA.3.2	TSF.ADM_CFG	The TSF allows the administrator to change the value of the user decrypt attribute.

7.4.2.5. User Password

SFR	TSF	Rationale
FIA_SOS.1.1	TSF.PWD_STR	The TSF implements the required minimum password length enforcement.
FIA_UAU.1.1	TSF.REC_SEC	The access recovery procedure is available before the user is logged on.
FIA_UAU.1.2	TSF.DSK_ENC	Describes that the only TSF mediated function available before successful logon is the access recovery procedure described in TSF.KEY_REC.
FIA_UAU.7.1	TSF.USR_LOG	Describes how the application does not display passwords as the user types them, displaying asterisks instead.
FIA_AFL.1.1	TSF.USR_LOG	Describes the behavior when the user types their password incorrectly more than the EP Hard Disk Administrator defined maximum number of times.
FMT_SAE.1.1	TSF.USR_CFG	Describes that the TOE restricts a list of configuration changes to being made only by the administrator. This list includes the user password expiration time configuration option that meets this requirement.
FMT_SAE.1.2	TSF.USR_LOG	Describes the actions taken by the TOE when the user password expires that meet this requirement.
FPT_STM.1.1	TSF.TIM_STP	The TSF describes the TOE's source of time. As described no special services are used. The time stamp for this application is not a high assurance requirement as there is no immediate third party attack if the user bypasses the expiry limits by changing the time. The function is just to encourage the user to adopt good password change policies. A user hostile to their own security can otherwise weaken security for example by choosing poor passwords minimally meeting enforced requirements, or writing passwords down.
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.PWD_HST	The TSF describes how the TOE enforces the password history policy that meets these requirements.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.6. Disk Key

SFR	TSF	Rationale

FCS_CKM.1.1	TSF.DSK_KEY	The TSF describes the cryptographic key generation algorithm used for the Disk Key. The TSF meets the requirement.
FCS_CKM.3.1	TSF.KEY_REC	The TSF describes the key recovery method used for recovering Disk Keys. The TSF meets the requirement.
FCS_CKM.4.1	TSF.KEY_OVR	The TSF describes the TOEs handling of sensitive information. The TSF meets the requirement.
FCS_COP.1.1	TSF.DSK_ENC	The TSF describes the cryptographic operations the TOEs uses with the Disk Key. The TSF meets the requirement.

7.4.2.7. Disk KEK

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DSK_KEK	The TSF describes the cryptographic key generation algorithm used for the Disk KEK. The TSF meets the requirement.
FCS_COP.1.1	TSF.DSK_ENC	The TSF describes the cryptographic operations the TOEs uses with the Disk KEK. The TSF meets the requirement.

7.4.2.8. Authenti-Check Logon

SFR	TSF	Rationale
FIA_SOS.1.1	TSF.ATK_LOG	The TSF implements the required minimum Authenti-Check answer length enforcement.
FIA_UAU.1.1	TSF.REC_SEC	The access recovery procedure is available before the user is logged on.
FIA_UAU.1.2	TSF.DSK_ENC	Describes that the only TSF mediated function available before successful logon is the access recovery procedure described in TSF.KEY_REC.

7.4.2.9. Authenti-Check Key Recovery

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.ATK_REC	The TSF describes the Authenti-Check key recovery mechanism used to allow the user to regain access to their data. The TSF meets the requirements.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.10. Authenti-Check KRK

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.ATK_KRK	The TSF describes the cryptographic key generation algorithm used for the Disk KRK. The TSF meets the requirement.
FCS_COP.1.1	TSF.ATK_REC	The TSF describes the cryptographic operations the TOEs uses with the Authenti-Check KRK. The TSF meets the requirement.

7.4.2.11. Administrator Configuration

SFR	TSF	Rationale
FMT_MOF.1.1	TSF.ADM_CFG	The TSF describes the restriction that only the EP Hard Disk Administrator may modify the administrator-restricted configuration options, or by implication, the configuration update messages. The technical aspects of ensuring update messages have integrity protection is covered in TSF.DSA_KP. The TSF meets the requirement.
FMT_MSA.2.1	TSF.ADM_CFG	The TSF describes the restrictions the TOE places on configuration values to ensure safety and security. The TSF meets the requirement.
FMT_MSA.1.1 FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.USR_CFG	The TSF describes the restriction that only the EP Hard Disk Administrator can change the user configuration settings restrictions. The TSF meets the requirements.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.12. Administrator Database Encryption

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.DB_ENC	The SFRs describes the policy level intent of the database encryption function. The TSF meets the requirement.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.13. Administrator Password

SFR	TSF	Rationale
FIA_SOS.1.1	TSF.ADM_LOG	The TSF describes the restrictions on administrator choices of passwords. The TSF meets the requirement.
FIA_UAU.2.1	TSF.UAD_LOG TSF.REC_SEC	The TSFs describe the implementation features respectively that prevent the administrator from performing any TSF mediated functions on the User Program, and that prevent the administrator from performing any TSF mediated functions on the Administrator Program prior to logon.
FIA_UID.2.1	TSF.UAD_LOG TSF.REC_SEC	The TSFs describe the implementation features respectively that prevent the administrator from performing any other TSF mediated functions, and that prevent the administrator from performing any TSF mediated functions on the Administrator Program prior to logon.
FIA_AFL.1.1	TSF.UAD_LOG TSF.ADM_LOG	The TSFs describe the enforcement of restrictions on the number of unsuccessful logons that are allowed respectively on the Admin Logon function of the User Program, and on the Administrator Program.

7.4.2.14. Administrator Database Key

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DB_KEY	The TSF describes the cryptographic key generation algorithm used for the Administrator Database key. The TSF meets the requirement.
FCS_CKM.3.1	TSF.BAK_DBK	The TSF describes the method used for secure backup of the database key. The TSF meets the requirement.
FCS_CKM.4.1	TSF.KEY_OVR	The TSF describes the TOEs handling of sensitive information. The TSF meets the requirement.
FCS_COP.1.1	TSF.DB_ENC	The TSF describes the cryptographic operations the TOEs uses with the Administrator Database key. The TSF meets the requirement.

7.4.2.15. Administrator Database KEK

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DB_KEK	The TSF describes the cryptographic key generation algorithm used for the Administrator Database KEK. The TSF meets the requirement.
FCS_COP.1.1	TSF.DB_ENC	The TSF describes the cryptographic operations the TOEs uses with the Administrator Database KEK. The TSF meets the requirement.

7.4.2.16. Administrator ECDSA Private Key

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DSA_KP	The TSF describes the cryptographic key derivation algorithm used to derive the administrator ECDSA private key. The TSF meets the requirement.
FCS_CKM.3.1	TSF.DSA_KP	The TSF describes how old versions of the ECDSA private key are backed up in the encrypted Administrator Database.
FCS_COP.1.1	TSF.ADM_CFG	The TSF describes the cryptographic operations the TOEs uses with the administrator ECDSA private key. The TSF meets the requirement.

7.4.2.17. Administrator ECDSA Public Key

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DSA_KP	The TSF describes the cryptographic key generation algorithm used to derive the administrator ECDSA public key. The TSF meets the requirement.
FCS_CKM.2.1	TSF.PK_DST	The TSF describes the initial distribution of the DSA public key at installation time.
FCS_CKM.2.1	TSF.PK_DST	The TSF describes the public key distribution mechanism used to distribute updated ECDSA public keys corresponding to new administrator passwords.
FCS_COP.1.1	TSF.PK_DST TSF.ADM_CFG	The TSFs describes the cryptographic operations the TOEs uses the administrator ECDSA public key for. The User Program uses the ECDSA public key to verify the signatures in the configuration update messages it receives. The TSF meets the requirement.

7.4.2.18. Elliptic Curve Key Recovery

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.KEY_REC TSF.REC_SEC	The aspects of the Elliptic Curve recovery access control protocol referred to in FCS_ACC.1.1, FDP_ACF.1.1 and FDP_ACF.1.2 are described in the EC-based key recovery TSF. TSF.REC_SEC describes the recovery procedure, and TSF.KEY_REC describes the technical aspects of the EC-based recovery protocol.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.19. User Program Admin Logon

SFR	TSF	Rationale
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.UAD_LOG TSF.ADM_LOG	The aspects of the Elliptic Curve recovery access control protocol referred to in FCS_ACC.1.1 and FDP_ACF.1.1 are described in the User Program Admin Logon TSF. TSF.UAD_LOG describes how User Program Admin Logon is granted technically, and describes the Admin Logon function of the User Program. TSF.ADM_LOG implements rule (1) given in FDP_ACF.1.2 by preventing the administrator using the logon function until he or she has logged on.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.2.20. Administrator ECDH Private Key

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DH_KP	The TSF describes the cryptographic key derivation algorithm used to derive the administrator ECDSA private key. The TSF meets the requirement.
FCS_COP.1.1	TSF.KEY_REC	The TSF describes the cryptographic operations the TOE uses the administrator ECDH private key for. The TSF meets the requirement.

7.4.2.21. Administrator ECDH Public Key

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.DH_KP	The TSF describes the cryptographic key generation algorithm used to derive the administrator ECDH public key. The TSF meets the requirement.
FCS_CKM.2.1	TSF.PK_DST	The TSF describes the initial distribution of the ECDH public key at installation time.
FCS_CKM.2.1	TSF.PK_DST	The TSF describes the public key distribution mechanism used to distribute updated ECDH public keys corresponding to new administrator passwords.
FCS_COP.1.1	TSF.KEY_REC	The TSFs describes the cryptographic operations the TOEs uses the administrator ECDSA public key for. The User Program uses the ECDSA public key to verify the signatures in the configuration update messages it receives. The TSF meets the requirement.

7.4.2.22. Elliptic Curve KRK

SFR	TSF	Rationale
FCS_CKM.1.1	TSF.EC_KRK	The TSF describes the cryptographic key generation algorithm used to derive the ECKRK. The TSF meets the requirement.
FCS_COP.1.1	TSF.KEY_REC TSF.REC_BLK	The TSFs describes the operations the ECKRK is used for.

7.4.2.23. User Configuration

SFR	TSF	Rationale
FMT_MOF.1.1	TSF.USR_CFG	The TSF describes the restriction that only the authorized user, EP Hard Disk Administrator, Corporate Administrator, or Local Administrator may modify the configuration options. The TSF meets the requirement.
FDP_ACC.1.1 FDP_ACF.1.1 FDP_ACF.1.2	TSF.USR_CFG	The TSF describes the restrictions on roles that can modify the configuration on the User Program, and the restrictions on the configuration values that can be set. The TSF meets the requirement.
FDP_ACF.1.3	N/A	Empty requirement.
FDP_ACF.1.4	N/A	Empty requirement.

7.4.3. Rationale AM to Security Assurance Requirements

The following table gives the tracing from Security Assurance Measures to Security Assurance Requirements.

Assurance Measure Requirement	Assurance Measure	Comments
ACM_CAP.1.1D ACM_CAP.1.1C ACM_CAP.1.2C	AM.ACM_CAP	The assurance measure meets all aspects of the assurance measure requirements.
ACM_CAP.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.1D ADO_IGS.1.1C	AM.USR_DOC AM.ADM_DOC	The assurance measures that are described in the AM.USR_DOC user documentation meet all aspects of the assurance measure requirements for the User Program. The assurance measures that are described in the AM.ADM_DOC administrator documentation meet all aspects of the assurance measure requirements for the Administrator Program and One-Time Password Program.
ADO_IGS.1.1E	Evaluation	To be evaluated by the evaluator.
ADO_IGS.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.1D ADV_FSP.1.2C ADV_FSP.1.3C ADV_FSP.1.4C	AM.ADV_FSP	The assurance measure meets all aspects of the assurance measure requirements.
ADV_FSP.1.1E	Evaluation	To be evaluated by the evaluator.
ADV_FSP.1.2E	Evaluation	To be evaluated by the evaluator.
ADV_RCR.1.1D ADV_RCR.1.1C	AM.ADV_RCR	The assurance measure meets all aspects of the assurance measure requirements in showing a tracing from the TSFs to the Informal Functional Specification.
ADV_RCR.1.1E	Evaluation	To be evaluated by the evaluator.
AGD_ADM.1.1D AGD_ADM.1.2C AGD_ADM.1.3C AGD_ADM.1.4C AGD_ADM.1.5C AGD_ADM.1.6C AGD_ADM.1.7C AGD_ADM.1.8C	AM.ADM_DOC	The assurance measure meets all aspects of the assurance measure requirements.
AGD_ADM.1.1E	Evaluation	To be evaluated by the evaluator.
AGD_USR.1.1D AGD_USR.1.1C AGD_USR.1.2C AGD_USR.1.3C AGD_USR.1.4C AGD_USR.1.5C AGD_USR.1.6C	AM.USR_DOC	The assurance measure meets all aspects of the assurance measure requirements.
AGD_USR.1.1E	Evaluation	To be evaluated by the evaluator.

ATE_IND.1.1C	AM.ATE_IND	The assurance measure meets the assurance measure requirement.
ATE_IND.1.1E	Evaluation	To be evaluated by the evaluator.
ATE_IND.1.2E	Evaluation	To be evaluated by the evaluator.

7.5. Informal Functional Specification Rationale

7.5.1. Rationale Introduction

In the following section, the TSF functions are traced to the corresponding application functionality as described in the Informal Functional Specification, and notes are given providing a rationale for the coverage provided. The tracing and rationale are organized as a table and are grouped by TSF function. The Informal Functional Specification is at a higher level than the more detailed specification information provided by the TSF functions.

7.5.2. Rationale by TSF

The product description given in the Administrator and User Guides comprises the Informal Functional Specification for evaluation purposes. The TSFs are defined in Section 6.1 TOE Security Functions, and the TSFs are given in the same order as their presentation in that section.

TSF	Rationale
TSF.NO_STO	The principle of not storing sensitive information is a design consideration used in the implementation of the TOE, but the application of this principle is not further detailed in the Informal Functional Specification.
TSF.KEY_OVR	The requirement to overwrite changed or decommissioned keys is a design consideration used in the implementation of the TOE, but the application of this principle is not further detailed in the Informal Functional Specification.
TSF.ENC_REL	The requirement to protect against data loss is a design consideration used in the implementation of the TOE, but the approaches taken to ensure this objective are not further specified in the Informal Functional Specification.
TSF.SEC_ROL	The security roles are used in the Informal Functional Specification. The different administrator roles are described in the section on “User Program Setup Wizard” of the Administrator Guide.
TSF.USR_LOG	The use of the user login function is described in the Informal Functional Specification in the section on “EP Hard Disk Interface / Logging On” of the User Guide.

TSF.PWD_HST	The use of the password history function is described in the section on “User Program Setup Wizard / Password Management”.
TSF.TIM_STP	The source of time used in the TOE implementation is as specified in this TSF, but this use is not further specified in the Informal Functional Specification.
TSF.ATK_LOG	The use of the Authenti-Check Logon function is described in the section “Forgotten Passwords / Using Authenti-Check” of the User Guide.
TSF.ATK_REC	The use of the Authenti-Check Key Recovery function is as specified in this TSF, but this use is not further specified in the Informal Functional Specification.
TSF.ATK_KRK	The derivation and use of the Authenti-Check Key Recovery Key is as specified in this TSF, but this use is not further specified in the Informal Functional Specification.
TSF.ADM_LOG	The use of the EP Hard Disk Administrator Logon function is described in the section on “Installation and Setup / Logging on” in the Administrator Guide.
TSF.UAD_LOG	The use of the User Program Admin Logon is described in the section on “Installation and Setup / Logging on” in the Administrator Guide.
TSF.CRY_TST	The cryptographic library function self-test functions described in the TSF are implemented in the TOE but are not further detailed in the Informal Functional Specification as it has no user visible interface.
TSF.TDA_CSM	Checksums on TSF data are used in the TOE implementation, but their implementation is not further detailed in the Informal Functional Specification as this function has no user visible interface.
TSF.BIN_CSM	The TOE implementation uses checksums at startup as described in the TSF, but their implementation is not further detailed in the Informal Functional Specification as this function has no user visible interface.
TSF.ACC_BAN	The TOE presents access banners as described in the TSF. The messages displayed in the access banners are set as described in the section on “User Program Setup Wizard / User Messages” in the Administrator Guide.
TSF.REC_SEC	The use of the access recovery procedure is

	described in the section on “Forgotten Passwords / Using the One-Time Password Program” of the User Guide.
TSF.KEY_REC	The key recovery mechanism is as specified in this TSF but is not further described in the Informal Functional Specification as it is not a user visible aspect of the recovery process.
TSF.REC_BLK	The recovery block structures are as specified in this TSF but they are not further described in the Informal Functional Specification as they are not a user visible aspect of the recovery process.
TSF.EC_KRK	The derivation of the Elliptic Curve Key Recovery is as specified in this TSF, but is not further described in the Informal Functional Specification as it is not a user visible aspect of the recovery process.
TSF.PWD_STR	The minimum password length restrictions described in the TSF are implemented in the TOE as described in “Installation and Setup / Local Installation – Overview” in the User Guide.
TSF.ADM_CFG	The administrator configurable aspects of the User Program are described in the section on “User Program Setup Wizard” of the Administrator Guide.
TSF.PK_DST	The public keys are distributed in the install packages as described in the TSF, but this aspect of key distribution is not further detailed in the Informal Functional Specification as it is not a user visible aspect of the recovery process.
TSF.USR_CFG	The user configuration are described in overview in the section on “Installation and Setup” of the User Guide.
TSF.INI_ENC	The use of the Initial Encryption function and the options relating to this function are described in overview in section on “User Program Setup Wizard / Initial Encryption Settings” of the Administrator Guide.
TSF.DSK_ENC	The use of the Initial Encryption function is described in the section on “Installation and Setup / Pre-encrypted Drives – Overview”.
TSF.DSK_DEC	The disk decryption mechanism and policies surrounding its use are described in the section on “Initial Encryption and Decryption / Decrypting a Drive” in the User Guide.
TSF.DSK_KEY	The generation and use of the disk key is as specified in this TSF but this aspect of the TOE is

	not further described in the Informal Functional Specification as it is not a user visible function.
TSF.DSK_KEK	The derivation and use of the disk key encryption key is as specified in this TSF but this aspect of the TOE is not further described in the Informal Functional Specification as it is not a user visible function.
TSF.DB_KEY	The administrator database key is derived as specified in this TSF but this aspect of the TOE is not further described in the Informal Functional Specification as it is not a user visible function.
TSF.BAK_DBK	The administrator database key backup option is described in "Appendix A" of the Administrator Guide under the entry for "Symmetrical Key".
TSF.DB_KEK	The database KEK is derived as described in the TSF in the TOE implementation but this process is not further detailed in the Informal Functional Specification.
TSF.DB_ENC	The administrator database is encrypted as described in the TSF in the TOE implementation, but this process is not further detailed in the Informal Functional Specification.
TSF.DSA_KP	The derivation of the ECDSA key pairs and their use is as specified in this TSF, but this is not further described in the Informal Functional Specification as it is not a user visible aspect of the TOEs operation.
TSF.DH_KP	The derivation of the ECDH key pairs and their use is as specified in this TSF, but this is not further described in the Informal Functional Specification as it is not a user visible aspect of the TOEs operation.

8. Terminology

8.1. Cryptography Acronyms

AES Advanced Encryption Standard – see **[AES]**
CBC Cipher Block Chaining (mode) – see **[AES-MODES]**
DSA Digital Signature Standard – see **[ECDSA]**
DH Diffie-Hellman key negotiation – see **[IEEE-P1363]**
EC Elliptic Curve (Cryptography / Crypto-system)
ECDH Elliptic Curve analog of Diffie-Hellman – see **[ECDH]**
ECDSA Elliptic Curve analog of DSA – see **[ECDSA]**
EP Encryption Plus

FIPS	Federal Information Processing Standards (US)
IEEE	Institute of Electrical and Electronics Engineers
KDF	Key Derivation Function
KEK	Key Encryption Key
KRK	Key Recovery Key
KDF2	Key Derivation Function 2 – a key-derivation specified in **[KDF2]**
MD5	Message Digest 5 – see **[MD5]**
PBKDF2	Password-Based KDF – see **[PBKDF2]**
PKCS	Public Key Cryptography Standards – de facto set of standards published by RSA Data Security – for example see **[PKCS#5]**
RSA	Rivest-Shamir-Adleman public key algorithm – see **[RSA]**
SHA1	Secure Hash Algorithm 1 – see **[SHA1]**
SHA2	Secure Hash Algorithm – see **[SHA-256]**

8.2. Common Criteria Acronyms

This section reproduces the Common Criteria acronyms section. Not all of these acronyms are used in this document.

CC	Common Criteria
EAL	Evaluation Assurance Level
IT	Information Technology
PP	Protection Profile
SF	Security Function
SFP	Security Function Policy
SOF	Strength of Function
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSFI	TSF Interface
TSP	TOE Security Policy

8.3. Common Criteria Glossary

This section reproduces the Common Criteria terms. Not all of these terms are used in this document.

Assignment — The specification of an identified parameter in a component.

Assurance — Grounds for confidence that an entity meets its security objectives.

Attack potential — The perceived potential for success of an attack, should an attack be launched, expressed in terms of a threat agent's expertise, resources and motivation.

Augmentation — The addition of one or more assurance component(s) from Part 3 to an EAL or assurance package.

Authentication data — Information used to verify the claimed identity of a user.

Authorized user — A user who may, in accordance with the TSP, perform an operation.

Class — A grouping of families that share a common focus.

Component — The smallest selectable set of elements that may be included in a PP, an ST, or a package.

Connectivity — The property of the TOE that allows interaction with IT entities external to the TOE. This includes exchange of data by wire or by wireless means, over any distance in any environment or configuration.

Dependency — A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.

Element — An indivisible security requirement.

Evaluation — Assessment of a PP, an ST, or a TOE against defined criteria.

Evaluation Assurance Level (EAL) — A package consisting of assurance components from Part 3 that represents a point on the CC predefined assurance scale.

Evaluation authority – A body that implements the CC for a specific community by means of an evaluation scheme and thereby sets the standards and monitors the quality of evaluations conducted by bodies within that community.

Evaluation scheme -- The administrative and regulatory framework under which the CC is applied by an evaluation authority within a specific community.

Extension — The addition to an ST or PP of functional requirements not contained in Part 2 and/or assurance requirements not contained in Part 3 of the CC.

External IT entity — Any IT product or system, untrusted or trusted, outside of the TOE that interacts with the TOE.

Family — A grouping of components that share security objectives but may differ in emphasis or rigor.

Formal — Expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts.

Human user — Any person who interacts with the TOE.

Identity — A representation (e.g. a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.

Informal — Expressed in natural language.

Internal communication channel — A communication channel between separated parts of TOE.

Internal TOE transfer — Communicating data between separated parts of the TOE.

Inter-TSF transfers — Communicating data between the TOE and the security functions of other trusted IT products.

Iteration — The use of a component more than once with varying operations.

Object — An entity within the TSC that contains or receives information and upon which subjects perform operations.

Organizational security policies — One or more security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

Package — A reusable set of either functional or assurance components (e.g. an EAL), combined together to satisfy a set of identified security objectives.

Product — A package of IT software, firmware and/or hardware, providing functionality designed for use or incorporation within a multiplicity of systems.

Protection Profile (PP) — An implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.

Reference monitor — The concept of an abstract machine that enforces TOE access control policies.

Reference validation mechanism — An implementation of the reference monitor concept that possesses the following properties: it is tamperproof, always invoked, and simple enough to be subjected to thorough analysis and testing.

Refinement — The addition of details to a component.

Role — A predefined set of rules establishing the allowed interactions between a user and the TOE.

Secret — Information that must be known only to authorized users and/or the TSF

in order to enforce a specific SFP.

Security attribute — Information associated with subjects, users and/or objects that is used for the enforcement of the TSP.

Security Function (SF) — A part or parts of the TOE that have to be relied upon for enforcing a closely related subset of the rules from the TSP.

Security Function Policy (SFP) — The security policy enforced by an SF.

Security objective — A statement of intent to counter identified threats and/or satisfy identified organization security policies and assumptions.

Security Target (ST) — A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.

Selection — The specification of one or more items from a list in a component.

Semiformal — Expressed in a restricted syntax language with defined semantics.

Strength of Function (SOF) — A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behavior by directly attacking its underlying security mechanisms.

SOF-basic — A level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by threat agents possessing a low attack potential.

SOF-medium — A level of the TOE strength of function where analysis shows that the function provides adequate protection against straightforward or intentional breach of TOE security by threat agents possessing a moderate attack potential.

SOF-high — A level of the TOE strength of function where analysis shows that the function provides adequate protection against deliberately planned or organized breach of TOE security by threat agents possessing a high attack potential.

Subject — An entity within the TSC that causes operations to be performed.

System — A specific IT installation, with a particular purpose and operational environment.

Target of Evaluation (TOE) — An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.

TOE resource — Anything useable or consumable in the TOE.

TOE Security Functions (TSF) — A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TOE Security Functions Interface (TSFI) — A set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which TOE resources are accessed, mediated by the TSF, or information is obtained from the TSF.

TOE Security Policy (TSP) — A set of rules that regulate how assets are managed, protected and distributed within a TOE.

TOE security policy model — A structured representation of the security policy to be enforced by the TOE.

Transfers outside TSF control — Communicating data to entities not under control of the TSF.

Trusted channel — A means by which a TSF and a remote trusted IT product can communicate with necessary confidence to support the TSP.

Trusted path — A means by which a user and a TSF can communicate with necessary confidence to support the TSP.

TSF data — Data created by and for the TOE, that might affect the operation of the TOE.

TSF Scope of Control (TSC) — The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.

User — Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

User data — Data created by and for the user that does not affect the operation of the TSF.

9. References

[AES] Federal Information Processing Standards Publication 197 – Advanced Encryption Standard

[AES-MODES] National Institute of Standards – Special Publication 800-38A – Recommendation for Block Cipher Modes of Operation, 2001 Edition.

[DSS] Federal Information Processing Standards Publication 186-2 – Digital Signature Standard, 27 January 2000.

[ECDSA] Elliptic Curve Digital Signature Algorithm (ECDSA) from **[DSS]**.

[ECDH] Elliptic Curve Diffie-Hellman (ECDH) algorithm DL/ECKAS-DH1 (Discrete Log/Elliptic Curve Key Agreement Scheme Diffie-Hellman version 1) using derivation primitive ECSVDP-DH (Elliptic Curve Secret Value Derivation Primitive – Diffie Hellman version) from **[IEEE-P1363]**. This algorithm is used with KDF2 – see **[KDF2]**.

[IEEE-P1363] IEEE P1363 – Standard Specifications for Public Key Cryptography, Draft Version 13, 12 November 1999.

[IEEE-P1363a] IEEE P1363a / D9 (Draft Version 9) – Standard Specifications for Public Key Cryptography: Additional Techniques, 13 July 2001.

[KDF2] Key Derivation Function 2 (KDF2) algorithm from **[IEEE-P1363a]**.

[MD5] RFC1321 – The MD5 Message Digest Algorithm, R. Rivest, MIT Laboratory for Computer Science, April 1992.

[OpenSSL] OpenSSL cryptographic library http://www.openssl.org

[PBKDF2] Password-Based Key Derivation Function 2 (PBKDF2) algorithm from **[PKCS#5]**.

[PKCS#5] Public Key Cryptography Standard #5 v2.0: Password-Based Cryptography Standard, RSA Laboratories, 25 March 1999.

[RNG] OpenSSL version 0.9.6 random number generator with use of MD5 replaced with SHA1.

[RSA] IFEP-RSA (Integer Factorization Encryption Scheme) using IFEP-RSA (Integer Factorization Encryption Primitive) and IFDP-RSA (Integer Factorization Decryption Primitive) from IEEE-P1363 standard **[IEEE-P1363]**.

[SHA1] Federal Information Processing Standards Publication 180-1 – Secure Hash Standard, 17 April 1995.

[SHA-256] Draft Federal Information Processing Standards Publication 180-2 – Secure Hash Standard, 2001.