SafeGuard Easy Evaluation Documentation

Document Version:	1.04.00
Document Type:	Security Target / Sicherheitsvorgaben
Project Id:	
File Name:	D:\WorkShort\ST_SGE2000_EAL1_1_04_00.doc
Author(s):	Roland Reinl, Joachim Schneider
Office / Company:	Utimaco Safeware AG

Abstract: This document contains the Security Target for the Common Criteria certification of SafeGuard Easy for Windows 2000, Version 1.0, certification ID: BSI-DSZ-CC-0176.

Table Of Contents

1 Document Information

1.1 Owner / Master Location

Owner of this document is Thomas Reichert (TRE).

The location of the master copy is Developer Network Utimaco Unterfoehring at ST_SGE2000_EAL1_1_04_00.doc

1.2 Change History

Version	*Author*	*Date (finished)*	*Description*
1.00.00	RRE JOS	2001-04-17	first version
1.01.00	RRE	2001-06-08	Security objectives modified; new set of security functions; rationale updated
1.02.00	RRE	2001-08-24	extensive modifications to formulation of threats, security objective and assumptions; additional security requirements defined; more detailed rationales added to section 9.1
1.03.00	RRE	2001-10-09	modifications to description of “access” and “substantial access”, modified requirements for the environment, several editorial changes
1.04.00	RRE	2001-10-25	Modifications to dependencies of IT-environment SF, update of rationale accordingly, minor editorial changes and corrections

1.3 Distribution & Approval History

Version	*Distributed to / approved by*	*Date distributed*	*Date approved*
1.00.00	TRE, REN, THO, KAL	2001-04-17	2001-04-16, THO
1.01.00	TRE, REN, THO, KAL	2001-06-08	2001-06-11, THO
1.02.00	THO	2001-08-27	2001-09-04, THO
1.03.00	THO	2001-10-05	2001-10-19, THO
1.04.00	THO	2001-10-25	2001-10-26, THO

1.4 Assumptions made herein

2 ST Introduction

The chapter *ST Introduction* is divided into the following sections:

ST Identification – contains an identification of the TOE,

ST Overview – contains a short overview over the TOE's functions,

CC Conformance – contains the claims for the conformance of the Security Target to the CC.

2.1 ST Identification

This Security Target is the basis for the evaluation of the product *SafeGuard Easy for Windows 2000*.

The Target of Evaluation (TOE) is identified as:

SafeGuard Easy for Windows 2000, Version 1.0

An English and a German language version is included into the evaluation.

The User's Guide (using and administrating SGE) for each version is part of the TOE as printed document.

The TOE is called "SGE" in the following.

Members of the SafeGuard Easy product family for different operating system platforms (Windows 3.x, Windows 95, Windows 98, Windows NT) have already been evaluated and certified according to CC Version 2.0 by debisZert (debisZert: BSI-CC-0409) and according to ITSEC by the BSI (BSI-ITSEC-0012-1995).

2.2 ST Overview

SafeGuard Easy (SGE) is a software product to ensure secure access to data on Personal Computers (PCs). It works on a high security level but is easy to install, maintain and use.

This product under evaluation is designed for the Microsoft operating system Windows 2000. (Versions of SGE also are available for other PC operating systems, but are not part of this evaluation).

The security of SGE prevents unauthorised users from access to all data on the hard disk(s) of a PC operating under the named operating system.

Basically, the security provided by SGE bases upon the encryption of entire hard disk partitions. User authentication is done by PBA (Pre Boot Authentication) prior to booting the operating system. In this way, the access to data is restricted to authorised individuals only.

2.3 CC Conformance

The ST is structured according to the general rules listed in Part 1 of the Common Criteria [CC1].

The *TOE security assurance requirements* claim to be conformant to Part 3 of the Common Criteria [CC3],

The functional requirements for the TOE claim to be conformant to those in Part 2 of the Common Criteria [CC2].

The evaluation is based upon

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999
[CC2A]	Common Criteria for Information Technology Security Evaluation, Part 2, Annexes; Version 2.1, August 1999
[CC3]	[CC3]Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999

The individual assurance components for measuring the achieved assurance are those defined by

EAL1 (Evaluation Assurance Level 1)

in Part 3 of the Common Criteria [CC3].

The TOE does not claim to be conformant to any PP.

The assurance of a minimum strength of function (SOF) is not claimed, because this is not applicable for the evaluation assurance level EAL1.

The functional requirements are described in section 6.1.

3 TOE Description

The chapter *TOE Description* is divided into the following sections:

General Description – contains a basic description of the TOE,

TOE Components – contains a listing of the main components of the TOE,

TOE Hardware and Software Environment – contains a description of the technical IT environment, where the TOE is intended to be installed.

3.1 General Description

SafeGuard Easy (SGE) is a software product designed to protect user data on all types of Personal Computers (PCs) running Microsoft Windows 2000 as operating system. SafeGuard Easy is a software product installed on a PC to prevent unauthorised access to user data stored on hard disk partitions. In this context, user data means all files on hard disk partitions, i.e. data files, program files and even files of the operating system. The protection of the user data stored on hard disk partitions is realised by encryption. Encryption is done on sector level - not on file level. This provides the advantage of being independent from the behaviour of application programs and processing files difficult to handle, like temporary file areas or paging files of the operating system.

Encryption guarantees the confidentiality of data. However, it can not guarantee complete integrity of data, as e.g. sectors of the hard disk are not physically write protected. So, for example, the hard disk may be formatted, if it is possible to boot the system from a different booting device than the built-in hard disk. Usually, physical sector modifications on an encrypted hard disk will be detected, because (after decryption) they will at least generate unuseful random nonsense data.

User identification and authentication is done by PBA (Pre Boot Authentication) prior to booting the operating system. Only after a successful authentication, the user has access to the data on the hard disk partition. In this way, the access to data is restricted to the authorised individuals only. On a running system, after authentication, the encryption is completely transparent to the user, so that he is normally not aware of the security mechanisms behind. After shutting down the operating system and switching off the PC, the entire hard disk partitions are encrypted and therefore secured. Booting the PC from any device circumventing PBA results in a view to encrypted hard disk partitions.

Authentication bases upon user names and secret passwords. The cryptographic key necessary to encrypt the user data stored on the hard disk is encrypted with the password of each user and is secured in this way.

SGE provides additional features, which are not part of this evaluation. As an option, data stored on floppy disks and other removable devices (e.g. MO drives, ZIP drives) may also be protected by encryption. The function SAL (Secure Auto Logon) offers to store the operating system user name and password for a user under SafeGuard Easy and let SGE perform the user logon at the operating system automatically. A challenge-response authentication feature is built-in to allow a user to get access to the system even, if e.g. he forgot his password. Challenge-response can only be performed with the help of a second user with access to the system, and who has his password ready and operates a special challenge-response generation program delivered together with SGE.

SafeGuard Easy is installed from CD-ROM. The installation program together with the administration program installs the system kernel of SGE on the hard disk, adds some drivers to the operating system, changes the master boot record, and initially encrypts the hard disk partitions. After having installed SGE and completed the hard disk encryption, the PC is protected.

3.2 TOE Components

The Target of Evaluation (TOE) consists of

(i) the SafeGuard Easy program CD-ROM containing the installable program code and the installation program for SafeGuard Easy for Windows 2000 Version 1.0, English and German version,
where only the following parts of the installed programs implement the security functionality of the TOE:

(a) the system kernel of SGE,

(b) the master boot record of SGE,

(c) the drivers needed for encrypting and decrypting user data,

(d) the installation program and the administration program,

(ii) the User's Guide for using and administrating SGE, called
"SafeGuard Easy Windows 2000 – Mobile/Desktop Security – Version 1.0, User's Manual, Utimaco Safeware AG, August 2000" (English Version)
or
"SafeGuard Easy Windows 2000 – Mobile/Desktop Security – Version 1.0, Handbuch, Utimaco Safeware AG, August 2000" (German Version)

(iii) the User's Guide Enhancement for secure operation, called
"SafeGuard Easy Windows 2000 Version 1.0 – Manual for certification compliant operation – Utimaco Safeware AG, October 2001" (English Version)
or
"SafeGuard Easy Windows 2000 Version 1.0 – Handbuch für den zertifizierungskonformen Betrieb – Utimaco Safeware AG, October 2001" (German Version)

3.3 TOE Hardware and Software Environment

3.3.1 Hardware Requirements

The TOE runs on personal computer systems with following minimum requirements:

- microprocessor Intel Pentium (or successor type like Pentium II) or compatible device, with 32-bit internal operation, suitable for Windows 2000
- minimum system RAM of 32 MB,
- hard disk with a minimum of 4 MB free storage,
- CD-ROM drive for installation.

The TOE supports furthermore following hardware devices:

- up to four hard disks:
  hard disks may be accessed via IDE, Advanced IDE or SCSI controller,

Because of its security measures, SGE is especially suitable for the protection of user data on mobile computers.

3.3.2 Software Requirements

Operating System

The version of SGE under this evaluation is provided for the following operating system:

Microsoft Windows 2000 (Professional and Server)
(International versions for support of Western character sets)

SGE works with all available file systems under Windows 2000: FAT, FAT32, NTFS4, and NTFS5 (EFS).

Application Software Requirements

The TOE is working together with all application software, which is released for the mentioned operating system platform. However, application software, which is not using the respective Application Programming Interface of the OS platform for disk access, but circumventing some layers of the disk access system, may read encrypted sectors from the disk and therefore may not recognise the file structure on the disk correctly. Such software may also write plain text data directly onto a protected device. Then these data are not protected by the TOE against unauthorised disclosure.

In practice, such software has not been known to the vendor, except for special hard disk repair and copy functions. Using such software for hard disk repair and copy functions, while the TOE is installed, is not advised, as this also may - in extreme consequence - damage the TOE installation.

3.3.3 Connectivity Aspects

SGE works on any PC which meets hardware and software requirements, not regarding, if the PC is stand alone or if it is connected over a data line to any other computer system.

Data connection may include:

- Connection to a LAN (Local Area Network) or a WAN (Wide Area Network) by Ethernet, Arcnet or others
- Remote access connection to another computer system via serial line (serial cable, modem, USB connection).

In these cases it must be observed, that the security from SGE extends only to the local disk drives, and that there is no encryption of virtual drives in network environments.

Security may be inactive, when the secured PC is operated while connected to another computer system and parts of the PC's hard disk(s) are accessible to other users or programs (via shared partitions/drives/volumes, directories or files) within this connection. In this case, any user having access to those shares has access to the plain text data stored in it.

For these reasons, the threats defined for the TOE restrict denial of access for unauthorised users to the state, where the PC is not in operational state and the unauthorised individual tries to access data by anyhow setting the PC into operation or removing the hard disk from the PC and examining the device separately.

Also attention has to be paid to the fact, that, when the PC -with the TOE installed on it- is operated in connection to any other computer system, it might be possible for unauthorised individuals to manipulate the TOE in a way, that its security functionality can be circumvented or deactivated (e.g. by installing “Trojan Horse”-type programs/scripts). Therefore no partition-/drive-/volume-, directory- or file-shares shall be defined on a PC secured by the TOE.

When the TOE is operated in a network with connection to the Internet, a correctly installed and maintained firewall system shall be established to prevent access to the protected PC’s hard disk(s) and memory by unauthorised individuals from outside.

SGE is not intended to be used on servers in a network (however it will work there).

4 Security Environment

The chapter *Security Environment* is divided into the following sections:

Introduction – contains a definition of subjects, objects and operations

Secure Usage Assumptions – contains the assumptions made for the operation of the TOE,

Threats – contains a description of the threats averted by the TOE and the environment,

Organisational Security Policies – containing a description on the distinction of users in the TOE.

4.1 Introduction

4.1.1 Subjects, Objects and Operations

To simplify the definition of assumptions and of threats countered by the TOE, a definition of subjects and objects is preceded.

Subjects

Subjects relevant for considering the security of the TOE are:

<S.USER> Authorised user, i.e. all persons knowing a correct combination of user name and password or knowing the correct password for the standard user (if defined) for the current installation.
The following security attributes are assigned to that subject: User name (valid), password (valid).

<S.UNAU> Unauthorised individual (each persons knowing none of the passwords or user name/password combinations; these persons may also use an installation of SGE in another domain but they do not know any password for the domain considered here).
The following security attributes are assigned to that subject: User name (invalid), password (invalid).

Note on the distinction of subjects:

Basically SGE is able to distinguish between different users by checking user names and passwords. However, as the main task of SGE is to prevent any unauthorised individual from access to hard disk data, there is no further need in the scope of the evaluated functions of the TOE to distinguish between different user roles.

During installation, two users are predefined by the TOE: "system" and "user". One of these users ("system") is privileged by the fact, that this user account can not be deleted in order to avoid loss of authorised access to the TOE by accident. The person installing the TOE is first person able to define additional users and to determine the passwords of all users.

The intended method of using the TOE in this evaluation is to put all users on an equal footing: So, it is assumed that each authorised user is allowed to perform all management functions of the TOE.

The security functions of the TOE are only enforced by distinguishing between “authorised individuals (users)” <S.USER> and “unauthorised individuals” <S.UNAU>, as used for the definition of threats in section 4.3.

The user management of SGE offers the possibility of defining different rights on TOE management functions for different users. However, restricting the management rights for any user will not increase the security of the PC data and of the TOE itself within the scope of the evaluated TOE security functionality.

Objects

Objects relevant for considering the security of the TOE are mainly data objects (abbreviated with D.):

<D.USER> Plain text user data contained in partitions of the local hard disk drive(s) controlled by the TOE; plain text user data encloses data files, program files, operating system files and file system information on a hard disk.
The following security attribute is assigned to <D:USER>:
UserDataEncryptionType, with the only possible values {encrypted, decrypted}

Note: At installation time the encryption algorithm as well as the key can be selected for <D.USER>. After installation is completed both cannot be changed anymore. The attribute UserDataEncryptionType has the value “encrypted” after the TOE has been correctly installed and the initial hard disk encryption has been completely executed. The attribute will be set to “decrypted” by decrypting the hard disk during deinstallation of the TOE.

<D.PASSW> User passwords.

The following security attribute is assigned to <D:PASSW>:
User Name

Note:
Passwords are not stored within TSF data and are therefore listed separately.

<D.TSF> TOE software and its TSF data.
The following security attribute is assigned to <D.TSF>:
TSFDataEncryptionType, with the only possible value {encrypted}

Note:
For <D.TSF> neither encryption algorithm nor key nor encryption type can be selected or changed by any means at any time.

Access Operations

The TOE policy is to prevent unauthorised users from access to information by using encryption methods. Information, called "plain text" in the further document, is hidden by being

encrypted into “cipher text”. The following access operation is defined to specify the threats and the security policy of the TOE:

<ACC.SUB> Substantial Access,
means, that an individual – authorised or unauthorised – is accessing, i.e. reading, writing or modifying plain text user data on a hard disk (<D.USER>).
Substantial access does not include protection against writing physical sectors (cipher text) on the hard disk, but sectors modified this way will usually be decrypted to unusefull random nonsense data.

Furthermore it is defined, that substantial access by unauthorised individuals is only averted, when the PC is not in operational state, i.e. after the PC has been switched off by the user and must be booted by processing a master boot record (MBR) on any data media.

The PC is in operational state, after an authorised user has performed login by PBA, until the operating system has been shut down and the PC has been physically switched off. It is important to mention, that the PC remains in operational state, when the user invokes any screen/keyboard locking function or any suspend or hibernation mode provided by the operating system or by the PC's BIOS. During operational state, the plain text user data on the hard disk is always accessible, because the TOE's encryption/decryption functionality is active or can be used without any further authentication.

4.2 Secure Usage Assumptions

In this section several assumptions (or requirements to use SGE) are described. The first sections describe the hardware and software environment, where SGE is designed to fulfil its security functions. The next sections describe the physical, personnel, organisational and connectivity aspects which have to be regarded to operate SGE in a way that that the TOE’s security can be guaranteed.

4.2.1 Intended Usage Assumptions

SafeGuard Easy is a software product installed on a PC to prevent unauthorised access to user data stored in hard disk partitions. Only authorised individuals may use the computer (start/boot the operating system from an encrypted device, especially from the hard disk).

The following assumption is taken to guarantee the TOE’s security:

<A.INST> Installation Options
It is assumed, that the TOE is properly installed and configured regarding the required settings for the security attributes. These settings are listed in detail in the corresponding requirement <R.INST> in section 6.4 of this document.

4.3 Threats

4.3.1 Threats to Security

The following threats should be averted by the TOE:

<T.ACCESS> An unauthorised individual <S.UNAU> attempts to perform substantial access <ACC.SUB> to any plain text data stored on hard disk partitions defined as encrypted by the TOE <D.USER>. This attack is expected to be performed when the PC is not in operational state.

<T.MANAGE> An unauthorised individual <S.UNAU> attempts to perform TOE management operations (changing the protection status of the TOE or modifying other TSF data <D.TSF>). This attack is expected to be performed when the PC is not in operational state.

The following threats should be averted by the environment:

<T.PASSW> An unauthorised individual <S.UNAU> gets the password <D.PASSW> of an authorised individual <S.USER> (any user knowing any valid user name/password combination of the current installation). This includes password recording using hardware devices or software tools. In the case of password disclosure, an unauthorised person becomes an authorised person. As a consequence, there is no longer protection against <T.ACCESS> and <T.MANAGE>.

<T.INTRUD> An intruder <S.UNAU> succeeds in placing non-trusted software on the PC's hard disk designed to attack (disclose or modify) the TOE software or its TSF data <D.TSF>. The attacker's program will be executed by the authorised user (Trojan horse), unnoticed (virus), or accidentally (both). With such an attack, the attacker attempts (i) to disclose cryptographic keys or passwords in order to break or circumvent the certain security functions of the TOE, or (ii) to modify software of the TOE to cause the TOE's security functions or measures to fail or to operate against the security policy. In either cases, the attacker attempts to succeed in performing <T.ACCESS> or <T.MANAGE>.

<T.DIRECT> Non-trusted software, which does not use the respective Application Programming Interface of the OS platform for disk access, but directly accesses the hard disk by circumventing layers of the disk access system, is placed on the PC's hard disk or executed while the computer is operated. In this case, the threat <T.ACCESS> is no longer averted.

4.4 Organisational Security Policies

The Security Objectives of the TOE (chapter 5) are only derived from the identified threats (section 4.3) together with assumptions (section 4.2). There are no additional organisational security policies defined.

5 Security Objectives

The chapter *Security Objectives* is divided into the following sections:

TOE Security Objectives – contains the security objectives for the TOE,

Security Objectives for Environment – contains the security objectives for the environment.

The *Security Objectives* can be directly traced back to the *Threats* defined in section 4.3. Nevertheless, the description of the *Security Objectives* contains additional information to indicate how the security problem (*Threat*) is addressed by the TOE. This is to provide a clear link to understand the *TOE Functional Requirements* and the *IT Environment* and *Non-IT Environment Requirements*.

5.1 TOE Security Objectives

The TOE is designed to prevent unauthorised users from access to data and programs on PC hard disk partitions.

The *TOE's Security Objectives* are as follows:

<O.ACCESS> Unauthorised individuals <S.UNAU> shall not be able to perform any substantial access to user plain text data stored on hard disk partitions defined as encrypted by the TOE <D.USER>. This attempt is expected to be performed when the PC is not in operational state.
Solution:
These user data are protected by a TSF which encrypts the user data whenever being written onto the hard disk (imported into the TSC). Authorised individuals are identified by checking their respective password before the operating system is loaded. The latter function provides the cryptographic key necessary to access (decrypt and encrypt) the data stored on the protected hard disk partitions.

<O.MANAGE> Unauthorised individuals <S.UNAU> shall not be able to perform TOE management operations (changing the protection status of the TOE or modifying other TSF data) <D.TSF>. This attempt is expected to be performed when the PC is not in operational state.

Solution:
Access to TOE management operations is controlled by a TSF. Authorised individuals are identified by checking their user name and their respective password. To perform management operations TSF data must be accessed. These data are encrypted and can be accessed only if the correct password has been input.

5.2 Security Objectives for Environment

The *Security Objectives for Environment* are as follows:

<OE.INST> The TOE shall be properly installed. Details for the secure installation options are given in the requirement <R.INST> in section 6.4 of this document.

<OE.PASSW> Unauthorised individuals <S.UNAU> shall not get the password <D.PASSW> of an authorised individual <S.USER> (any user knowing any password of the current installation).

Solution:
The users are instructed to keep their password secret and not to write down their password, neither manually nor electronically.
The PC and its environment shall be protected against installing any devices, which enable capturing user password inputs on the keyboard.

<OE.DIRECT> Software which does not use the respective Application Programming Interface of the OS platform for disk access shall not be placed on the PC's hard disk or executed while the computer is operated.

Solution:
The users are instructed not to install or use utility programs like partition managers or disk copy programs while the TOE is installed and active.

<OE.INTRUD> An intruder <S.UNAU> shall not succeed in placing non-trusted software on the PC's hard disk designed to attack (disclose or modify) the TOE software or its TSF data <D.TSF>. The attacker's program might be executed by the authorised user (Trojan horse), unnoticed (virus), or accidentally (both).

Solution:
There are two different ways to place non-trusted software on the PC:
(i) by booting a switched off the PC from an internal or external device: this shall be averted by taking appropriate measures against booting from other devices than the built-in harddisk.
(ii) by accessing the running PC remotely via a network or modem line: this shall be averted by taking appropriate measures to secure the PC when connected to a network.

6 IT Security Requirements

The chapter *IT Security Requirements* is divided into the following sections:

TOE Security Functional Requirements - describes the functional requirements for the TOE on basis of CC functional components.

TOE Assurance Requirements – describes the requirements to assure that the TOE implements the functional requirements

Security Requirements for the IT Environment – describes the requirements defined for the IT environment.

Security Requirements for the Non-IT Environment – describes the requirements defined for the Non-TI environment.

6.1 TOE Security Functional Requirements

The *TOE Security Functional Requirements* are described using components taken from Part 2 of the Common Criteria.

The listed dependencies for each functional requirement do not include all options (where options are available on the Common Criteria using the “or” clause), but lists the dependencies, which are implemented as security functional requirements in the TOE.

6.1.1 Cryptographic Support (FCS)

6.1.1.1 Cryptographic key generation (FCS_CKM.1)

FCS_CKM.1.1
The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Random Key Generator* (defined in SafeGuard Easy for Windows 2000: Informal Functional Specification and Correspondence Demonstration, Utimaco, 2001) and specified cryptographic key sizes *64 bits (56 bits within used by DES) and 128 bits (used by IDEA)* that meet the following: *no defined standards*.

Hierarchical to: no other components.

Dependencies: FCS_COP.1, FCS_CKM.4, FMT_MSA.2

6.1.1.2 Cryptographic key destruction (FCS_CKM.4)

FCS_CKM.4.1
The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with standard pattern* that meets the following: *no defined standards*.

Hierarchical to: no other components.

Dependencies: FCS_CKM.1, FMT_MSA.2

6.1.1.3 Cryptographic operation (FCS_COP.1)(1)

FCS_COP.1.1
The TSF shall perform *symmetric data encryption and decryption of user data on the hard disk* in accordance with a specified cryptographic algorithm *DES* and cryptographic key sizes *56 bits* that meet the following: *DES standard with CBC encryption as published in ISO 8372, ISO/IEC 10116 standards*.

Hierarchical to: no other components.

Dependencies: FCS_CKM.1, FCS_CKM.4, FMT_MSA.2

6.1.1.4 Cryptographic operation (FCS_COP.1)(2)

FCS_COP.1.1
The TSF shall perform *symmetric data encryption and decryption of user data on the hard disk* in accordance with a specified cryptographic algorithm *IDEA* and cryptographic key sizes *128 bits* that meet the following: *IDEA standard as published by ASCOM Inc..*

Hierarchical to: no other components.

Dependencies: FCS_CKM.1, FCS_CKM.4, FMT_MSA.2

6.1.1.5 Cryptographic operation (FCS_COP.1)(3)

FCS_COP.1.1
The TSF shall perform *symmetric data encryption and decryption of TSF data* in accordance with a specified cryptographic algorithm *IDEA* and cryptographic key sizes *128 bits* that meet the following: *IDEA standard as published by ASCOM Inc..*

Hierarchical to: no other components.

Dependencies: FCS_CKM.1, FCS_CKM.4, FMT_MSA.2

6.1.2 User Data Protection (FDP)

6.1.2.1 Subset access control (FDP_ACC.1)(1) (Data Protection)

FDP_ACC.1.1
The TSF shall enforce the *data protection SFP on substantial access (read, write, modify) to all user data on hard disk(s) <ACC.SUB> to authorised users <S.USER> and unauthorised users <S.UNAU>* based on *user name and password*.

Hierarchical to: no other components.

Dependencies: FDP_ACF.1

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

FDP_ACF.1.1
The TSF shall enforce the *data protection SFP* to *plain text user data on local hard disk(s) <D.USER>* based on *UserDataEncryptionType*.

FDP_ACF.1.2
The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:
If an authorised user <S.USER> presents the valid user name and the valid password and if the attribute UserDataEncryptionType for <D.USER> has the value "encrypted", then substantial access to the user data on hard disk(s) is granted.
For an unauthorised user <S.UNAU>, who is not able to present a valid user name and a valid password and if the attribute UserDataEncryptionType for <D.USER> has the value "encrypted", then substantial access to the user data on the hard disk(s) is denied.

FDP_ACF.1.3
The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no rules*.

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

Hierarchical to: no other components.

Dependencies: FDP_ACC.1, FMT_MSA.3

6.1.2.3 Export of user data without security attributes (FDP_ETC.1)

FDP_ETC.1.1
The TSF shall enforce the *data protection SFP* when exporting user data, controlled under the SFP(s), outside of the TSC.

FDP_ETC.1.2
The TSF shall export the user data without the user data's associated security attributes.

Hierarchical to: no other components.

Dependencies: FDP_ACC.1

6.1.2.4 Import of user data without security attributes (FDP_ITC.1)

FDP_ITC.1.1
The TSF shall enforce *data protection SFP* when importing user data, controlled under the SFP, from outside of the TSC.

FDP_ITC.1.2
The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.

FDP_ITC.1.3
The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: *no additional import control rules*.

Hierarchical to: no other components.

Dependencies: FDP_ACC.1, FMT_MSA.3

6.1.3 Identification and Authentication (FIA)

6.1.3.1 User identification before any action (FIA_UID.2)

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

Hierarchical to: FIA_UID.1.

Dependencies: no dependencies

6.1.3.2 User authentication before any action (FIA_UAU.2)

FIA_UAU.2.1
The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.
Refinement: after a user is successfully authenticated then – and only then – the correct hard disk encryption key for the substantial access to the user data is provided (with the help of the user's password).

Hierarchical to: FIA_UAU.1.

Dependencies: FIA_UID.1

6.1.4 Security management (FMT)

6.1.4.1 Security roles (FMT_SMR.1)

FMT_SMR.1.1
The TSF shall maintain the role *authorised user*.

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

Hierarchical to: no other components.

Dependencies: FIA_UID.1

6.1.4.2 Management of security functions behaviour (FMT_MOF.1)

FMT_MOF.1.1
The TSF shall restrict the ability to *disable all security functions (uninstall the TOE)* to *authorised users*.

Hierarchical to: no other components.

Dependencies: FMT_SMR.1

6.1.4.3 Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1
The TSF shall enforce the *data protection SFP* to restrict the ability to *modify* the security attributes *UserDataEncryptionType, user name and password* to *authorised users*.

Hierarchical to: no other components.

Dependencies: FDP_ACC.1, FMT_SMR.1

6.1.4.4 Secure security attributes (FMT_MSA.2)

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

Hierarchical to: no other components.

Dependencies: ADV_SPM.1, FDP_ACC.1, FMT_MSA.1, FMT_SMR.1

Note 1:
The requirement ADV_SPM.1 (informal security policy model) will not be explicitly stated for this TOE. The reason is, that the TOE's security policy is rather simple. Its details are already stated in the sections *General Description* (3.1) and *Subjects, Objects and O* (4.1.1) in this document.

Note 2:
Since FDP_ACC.1 and FDP_IFC.1 are alternative dependencies to FMT_MSA.2.1 and FDP_ACC.1 has been selected, FDP_IFC.1 is not applicable.

6.1.4.5 Static attribute initialisation (FMT_MSA.3)(1)

FMT_MSA.3.1(1)
The TSF shall enforce the *data protection SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
Refinement: The TSF enforces the value "encrypted" for the attribute UserDataEncryptionType for <D.USER>, which is the most restrictive value for that attribute. This is guaranteed with the initial hard disk encryption after installation of the TOE.

FMT_MSA.3.2(1)
The TSF shall allow *no roles* to specify alternative initial values to override the default values when an object or information is created.

Hierarchical to: no other components.

Dependencies: FMT_MSA.1, FMT_SMR.1

6.1.4.6 Management of TSF data (FMT_MTD.1)

FMT_MTD.1.1
The TSF shall restrict the ability to *modify* the *TSF data (user names, passwords, hard disk encryption state, hard disk encryption key, boot option)* to *authorised user*.

Hierarchical to: no other components.

Dependencies: FMT_SMR.1

6.2 TOE Assurance Requirements

The *TOE security assurance requirements* are identical with those defined by the *Evaluation Assurance Level 1 (EAL1)*. These are:

Configuration management (Class ACM)
Version numbers (Component ACM_CAP.1)

Delivery and operation (Class ADO)
Installation, generation, and start-up (Component ADO_IGS.1)

Development (Class ADV)
Informal correspondence demonstration (Component ADV_RCR.1)
Informal functional specification (Component ADV_FSP.1)
Informal TOE security policy model (Component ADV_SPM.1)

Guidance documents (Class AGD)
Administrator guidance (Component AGD_ADM.1)
User guidance (Component AGD_USR.1)

Tests (Class ATE)
Independent testing – conformance (Component ATE_IND.1)

The *assurance requirements* are to give evidence that the security functions of the TOE work correctly.

6.3 Security Requirements for the IT Environment

There are the following *Security Requirements for the IT Environment*:

6.3.1 User Data Protection (FDP)

6.3.1.1 Subset access control (FDP_ACC.1)(2) (Data Protection In Network)

FDP_ACC.1.1
The security functions of the environment [1] shall enforce the *data protection in network SFP on substantial access (read, write, modify) to all user data on hard disk(s)<ACC.SUB> to authorised users <S.USER> and unauthorised users <S.UNAU>*.

Hierarchical to: no other components.

Dependencies: FDP_ACF.1

Note:

[1] See CEM 1.0, ASE_REQ.1-9, Para 409

This is a requirement to the (network-) operating system hosting the TOE, intended to prevent (together with <R.NOSHAR>) any subject from using a network connection to install a program/script as defined in <T.INTRUD> (4.3.1), suitable to tamper the TOE's security shield.

In an easy understandable form the requirement looks as:

It is required, that the (network-) operating system protects a network PC against any access to any Data stored on it, when this PC does not have any shared partitions/volumes/drives, directories or files.

6.3.1.2 Security attribute based access control (FDP_ACF.1)(2)

FDP_ACF.1.1
The security functions of the environment [1] shall enforce the *data protection in network SFP* to *plain text user data on local hard disk(s) <D.USER>* based on *no specific security attribute*.

FDP_ACF.1.2
The security functions of the environment [1] shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:
No user (neither <S.USER> nor <S.UNAU>) has access to the plain text data on local hard disk(s) <D.USER> over a network connection.

FDP_ACF.1.3
The security functions of the environment [1] shall explicitly authorise access of subjects to objects based on the following additional rules: *no rules*.

FDP_ACF.1.4
The security functions of the environment [1] shall explicitly deny access of subjects to objects based on the following rules: *no rules*.

Hierarchical to: no other components.

Dependencies: FDP_ACC.1, FMT_MSA.3

6.3.1.3 Dependencies of FDP_ACF.1 (2)

Note:

FDP_ACF.1 (2) contains the dependency FMT_MSA.3(2), which in turn requires FMT_SMR.1(2) and FMT_MSA.1(2).
Fulfilment of all those requirements would actually mean to model the requirements to an access control system of a generic (network-)operating system in terms of CC-Elements. This would be beyond the scope of this Security Target.
In addition this could involve the danger to cover up the only core requirement really needed, which is as simple as stated in 6.3.1.1 , last sentence.

6.4 Security Requirements for the Non-IT Environment

There are the following *Security Requirements for the Non-IT Environment*:

<R.INST> The TOE shall be installed directly by the installing user, not using a configuration file. The TOE shall be configured observing the following options during installation and first administration:

- Standard installation (with PBA).
- Selection of an appropriate encryption algorithm (DES with 16 rounds or IDEA).
- Minimum password length set to 6 characters.
- The user shall select the entire hard disk encryption, not the encryption of the operating system areas or single hard disk partitions only.

These settings must not be changed. Some of these options ensure that the security functions of the TOE are active after having installed the product on a PC.
These settings are default settings when defining the TOE security parameters with its administration program during first installation.

<R.BOOTP> Appropriate measures have to be taken to prevent unauthorised users from booting the PC by other devices than the internal hard disk. This is to prevent unauthorised users from placing non-trusted software on the PC to disclose passwords or manipulate security attributes.

<R.NOSHAR> No partitions/drives/volumes, directories or files on the local harddisk of the PC secured by the TOE shall be shared with other users, when the PC is connected to a network.

<R.CAPTKEY> The PC, where the TOE is installed, and the environment, where the PC is operated by any authorised user has to be secured against devices, which are capable of recording the password entered by an authorised user. Such devices may be keyboard grabbers in the cable between keyboard and PC, which are able to record the keystrokes as well as video cameras capturing the user during password entry.

<R.DIRECT> The users of the TOE are instructed not to install or run application software, which does not use the respective Application Programming Interface of the OS platform for disk access.

<R.PASSW> The users of the TOE are instructed to keep their passwords secret and not to write down their passwords, neither manually nor electronically.

7 TOE Summary Specification

The chapter *TOE Summary Specification* is organised as follows:

TOE Security Functions – specifies and describes the security functions of the TOE,

Assurance Measures – lists the items and documents provided to fulfil the assurance requirements.

7.1 TOE Security Functions

7.1.1 Overview

When the computer is started (before the operating system is booted) the user is prompted to input his user name and password (logon). If the user has successfully been authenticated by the function *Pre Boot Authentication (PBA)* <*SF1*> other functions of the TOE are invoked: The function *Protection of Data on Hard Disk Partitions* <*SF2*> ensures that all user data on the hard disk are encrypted. Even if the hard disk is removed, an attacker cannot perform any substantial access to these data. All these security functions use TSF data which are encrypted when the computer is not in operational state. After successful authentication the PBA provides all TSF data needed to the other security functions.

When the installation and administration program is started the user is prompted to input his user name and password (logon). Only authorised users can perform management operations. The function *Installation and Secure Administration* <*SF3*> ensures that only the authorised persons can (i) deinstall SGE (disable the security functions) or (ii) change TSF data.

Now the IT security functions are defined in an informal style to a level of detail necessary for understanding their intent.

A claim for the minimum strength of security functions is not applicable for EAL1 and is therefore omitted here.

7.1.2 Pre Boot Authentication (PBA) <SF1>

Pre Boot Authentication is a mechanism in SGE to check the user's authenticity before the operating system on a PC is booted from a hard disk.

With PBA installed, the system prompts for a valid user name and a password after starting the PC; when a standard user is defined, only a password is requested and the user name of the standard user is set by default (entering a different user name is possible after pressing a special function key). The password is entered via keyboard, when a login mask is displayed on the screen. Only a correct combination of user name and password – or the correct password for the defined standard user (if any) – enables to boot the operating system. In case of an incorrect password entered, the PBA module waits for some time until the next password entry is possible. This time increases for each incorrect entry.

PBA includes a mechanism, which calculates the hard disk encryption key for the boot partition and the remaining partitions from the password entered (for the use of the encryption key see <SF2>). The encryption key(s) cannot be calculated without knowledge of a valid password.

Some rules for password selection, like password minimum length and password expiration, may be defined during installation or later as a management operation. All users are enabled to change their user password during PBA logon.

This function implements the SFRs *User identification before any action (FIA_UID.2), User authentication before any action (FIA_UAU.2)*, *Security roles (FMT_SMR.1)*, *Management of security attributes (FMT_MSA.1)* and *Management of TSF data (FMT_MTD.1)*.

7.1.3 Protection of Data on Hard Disk Partitions <SF2>

The partitions of the installed hard disk(s) (up to four disks) of a PC <D.USER> can be held encrypted by SGE. The keys necessary to encrypt these data are provided by the security function <SF1> only if the user has been authenticated successfully. Hence, the security function <SF2> ensures that data provided by authorised users are protected when being stored on the hard disk.

The used encryption algorithm can be selected from the range of available system algorithms (DES or IDEA). The key for the encryption can be defined during installation or alternatively selected by a random key generator.

Note: SGE provides additional algorithms (Blowfish, Stealth, XOR etc.), but these are not part of the evaluation and may not be selected during installation of the evaluated versions due to <M.INST>.

All write and read accesses to the encrypted hard disk partitions are maintained by one of the encryption handlers (INT 13h handler or 32-bit disk access device driver) depending on the state of the system. On a write access, the data is encrypted; on a read access, the data is decrypted.

Booting a PC, where SGE is installed, from a floppy disk results in a state where information can't be retained from the hard disk(s) as a result of the encryption. When booting such a PC from hard disk, the control is handed from one part of SGE to another. First, the PBA module checks the authenticity of the user and calculates the encryption key for the boot partition. Next, an INT 13h handler is installed to decrypt the hard disk data during the boot phase. This handler remains active as long as hard disk access is performed during BIOS INT 13h (DOS e.g.). If a 32-bit operating system is booted, a device driver is automatically loaded, which is taking over hard disk on-line encryption and decryption during the 32-bit session.

Encryption state changes of any partition are defined by using the SGE administration program. As a result of such a change initial encryption or complete decryption of the affected partitions is automatically invoked by SGE. If the initial encryption or complete decryption is interrupted, it continues automatically after booting the PC again. This encryption task is performed by a background process, which is started after PBA, but before the operating system user logon.

This function implements the SFRs *Cryptographic operation (FCS_COP.1)(1)*, *Cryptographic operation (FCS_COP.1)(2)*, *Subset access control (FDP_ACC.1)(1) (Data Protection)*, *Security attribute based access control (FDP_ACF.1)(1)*, *Export of user data without security attributes (FDP_ETC.1)*, *Import of user data without security attributes (FDP_ITC.1)* and *Secure security attributes (FMT_MSA.2)*.

7.1.4 Installation and Secure Administration <SF3>

During system installation, the SGE kernel is placed on the harddisk and the system is rebooted with active kernel. Then the administration program is started to define the management data of the TOE. This includes the generation of the system key, which is used for encryption of the security relevant TSF data, like hard disk encryption key, user rights for TOE management operations etc. The administration program also generates the key used for hard disk encryption.

The system management data include all installation and maintenance parameters of the TOE: users, passwords, encrypted partitions, encryption keys, PBA definition and the rights of the user(s) to modify these parameters.

A special function of system management is the deinstallation of the TOE. During deinstallation the system kernel including the security data and keys is removed from the hard disk and the used disk space is overwritten.

SGE comprises an administration program to perform these functions. The administration program checks the identity and authenticity of the user calling it by asking for his user name and the password. This is done before the user is allowed to perform any action described below. It is possible that not only the user logged in at the moment is able to call the administration program but also every authorised user is able to do so.

The administration program is an application for Windows 2000 and allows to

1. change user names and passwords,
2. set password rules and PBA options,
3. define status, encryption algorithms and keys for hard disk encryption,
4. define encrypted and non-encrypted partitions on the hard disks,
5. perform deinstallation of the system kernel of SGE.

The following functions of the administration program refer to functions of SGE, which are not included into the scope of the TOE, their usage is optional:

- set management operation rights,
- define status, encryption algorithms and keys for floppy disk encryption and device encryption (as far as available),
- set MBR-related options,
- perform recovery operations like backup of the system kernel of SGE.

Each operation can be performed by any user, since it is assumed, that each user has identical rights for all management operations for the TOE.

The TOE management data is stored in the system kernel on the hard disk. To prevent unauthorised users from access or modification of the TOE management data, it is protected by an encryption mechanism. Only the access with a correct password will enable the system to decrypt the TOE management data and set the parameters into effect.

This function implements the SFRs *Cryptographic key generation (FCS_CKM.1)*, *Cryptographic key destruction (FCS_CKM.4)*, *Cryptographic operation (FCS_COP.1)(3)*, *Security roles (FMT_SMR.1)*, *Management of security functions behaviour (FMT_MOF.1)*, *Management of security attributes (FMT_MSA.1)*, *Secure security attributes (FMT_MSA.2)*, *Static attribute initialisation (FMT_MSA.3)* and *Management of TSF data (FMT_MTD.1)*.

7.1.5 Further Functions of SafeGuard Easy (informative only)

SGE supports some more functions for the convenience of secure operation and administration of PCs. **The following functions** are included into the product, but **are not part of the evaluated functions of the TOE**.

Floppy Disk Encryption

Besides hard disk encryption, SGE is able to encrypt floppy disks. The encryption algorithms available are the same as for the hard disk encryption. For the floppy disk encryption, for each user can be determined, if he is allowed to switch on and switch off the encryption.

Device Encryption

SGE is also enable to encrypt specific removable devices, like MOs, ZIPs and others. The encryption algorithms available are the same as for the hard disk encryption. For the device encryption, for each user can be determined, if he is allowed to switch on and switch off the encryption.

Challenge-Response Login

During PBA a Challenge Response Logon is possible using a challenge-response procedure. For this function, the PBA module generates a random challenge string, which can be transmitted by the user to a different user with access to that PC. With the Response Generation Program, the remote user creates a response string out of the challenge, his password and a function code. Then the user enters this response string at the PBA and is then enabled to perform the functions, which the remote user has enabled to him. The response code is only valid for a single login.

Secure Auto Logon

The product allows the pass through of the user name and the password from SGE to Windows 2000. This mechanism is called "SafeGuard Auto Logon". The user has to logon only once and his identity and authorisation is passed to the Windows 2000 logon. For this purpose his user name and password are stored within the TSF data for SGE. This option can be switched on and off with a command line tool.

MBR Protection

SGE operates with a modified master boot record (MBR) and is therefore interested in preventing other applications from manipulating the MBR after installation of SGE. Different kinds of reactions on a MBR modification request can be selected in the administration program.

Kernel Backup, Restore and Repair

All TSF maintenance data like user data, encrypted encryption keys and others are stored in the SGE kernel on the hard disk. Destroying or overwriting the kernel data may result in an inoperable system (keys can no longer be provided for hard disk decryption). For this case, a backup of the kernel data can be made on a floppy disk (or another removable media) and later restored in case of kernel damage.
The system has also a component, which is able to repair the kernel (within some limits) when errors occur. This is done by a particular program which can be executed by every user. The program only fixes a special part of the system kernel trying to restore an initial state of the kernel but grants no access to the data stored on the PC.

7.2 Assurance Measures

Appropriate documentation will be provided to satisfy the Security Assurance Requirements described in section 6.1.4.6.

The TOE itself does not provide any measure or mechanism to satisfy the assurance requirements. Assurance is guaranteed by the development process and by the users observing the corresponding directions.

The following table associates the measures and the documents describing them with the assurance requirements of CC EAL1:

EAL1 Requirement	Assurance Measure	Describing Document
ACM_CAP.1	The developers use a configuration management system, which includes the usage of properly assigned version numbers	Configuration Management Documentation, Utimaco, 2001
ADO_IGS.1	The installation, generation and start-up of the TOE is described in a separate document	SafeGuard Easy for Windows 2000: Installation, Generation and Start-Up, Utimaco, 2001
ADV_FSP.1	The informal functional specification is specified in a separate document	SafeGuard Easy for Windows 2000: Informal Functional Specification and Correspondence Demonstration, Utimaco, 2001
ADV_RCR.1	The correspondence demonstration is explained in a separate document together with the functional specification	SafeGuard Easy for Windows 2000: Informal Functional Specification and Correspondence Demonstration, Utimaco, 2001
ADV_SPM.1	The requirement ADV_SPM.1 (informal security policy model) will not be explicitly stated for this TOE. The reason is, that the TOE's security policy is rather simple. Its details are stated in the sections General Description (3.1) and Subjects, Objects and O (4.1.1) in this document.	SafeGuard Easy for Windows 2000: Security Target / Sicherheitsvorgaben, Utimaco, 2001, (this document)
AGD_ADM.1	Administrator and User Guidance are integrated into a common User's Manual. For each language version (German and English) a User's Manual is provided and delivered with the TOE.	User's Manual "SafeGuard Easy Windows 2000 – Mobile/Desktop Security – Version 1.0, User's Manual, Utimaco Safeware AG, August 2000" together with "SafeGuard Easy Windows 2000 Version 1.0 – Manual for certification compliant operation – Utimaco Safeware AG, Oktober 2001" (English Version)

		or "SafeGuard Easy Windows 2000 – Mobile/Desktop Security – Version 1.0, Handbuch, Utimaco Safeware AG, August 2000" together with "SafeGuard Easy Windows 2000 Version 1.0 – Handbuch für den zertifizierungskonformen Betrieb – Utimaco Safeware AG, Oktober 2001" (German Version)
AGD_USR.1	Administrator and User Guidance are integrated into a common User's Manual. For each language version (German and English) a User's Manual is provided and delivered with the TOE.	see <AGD_ADM.1>
ATE_IND.1	Independent Testing is carried out by the evaluation facility.	-

8 PP Claims

This Security Target does not make any claim that the TOE conforms with the requirements of a *Protection Profile*. As a result, sections "*PP Reference*", "*PP Refinement*" and "*PP Additions*" are omitted.

9 Rationale

The chapter *Rationale* is divided into the following sections:

Security Objectives Rationale – describing the relations between threats and security objectives,

Security Requirements Rationale – describing the relations between security objectives and security requirements rsp. security assurance requirement,

Dependency Rationale – describing the support of dependencies among the requirements.

TOE Summary Specification Rationale – describing the relations between the security requirements and the TOE's security functions and between the assurance requirements and the assurance measures,

PP Claims Rationale – describing the relations to a claimed Protection Profile.

The purpose of the ST rationale is to demonstrate that a complete, coherent and internally consistent set of security objectives, security requirements, IT security functions and assurance measures have been proposed to satisfy the identified security problem.

9.1 Security Objectives Rationale

It shall be demonstrated that the *Security Objectives* (chapter 5) are appropriate referring to the aspects of the *Security Environment* (chapter 4).

The stated *Security Objectives* (chapter 5) address all of the identified *Assumptions* (section 0) and *Threats* (section 4.3).

The following table shows that each security objective addresses at least one threat or assumption:

Objective	**Threats, Assumptions**
<O.ACCESS>	<T.ACCESS>
<O.MANAGE>	<T.MANAGE>
<OE.INTRUD>	<T.INTRUD>
<OE.DIRECT>	<T.DIRECT>
<OE.PASSW>	<T.PASSW>
<OE.INST>	<A.INST>

The table shows, that there is a bi-directional mapping between *Security Objectives* and *Threats/Assumptions*. Each *Security Objective* is assigned exactly to one *Threat* or to one *Assumption*. Each *Threat* and each *Assumption* is assigned exactly to one *Security Objective*.

Detailed Explanation / Justification:

Each *Security Objective* is derived from the corresponding *Threat* or *Assumption.* A comparison of the wording of the *Security Objectives* with that of the corresponding *Threats* or *Assumptions* shows, that each objective is exactly defined to avert the threat or assure the assumption.

<O.ACCESS> prevents unauthorised individuals <S.UNAU> from substantial access to user data stored on the hard disk partitions <D.USER> after the PC has been switched off. This security objective exactly counters threat <T.ACCESS>.

<O.MANAGE> prevents unauthorised individuals <S.UNAU> from performing management operations (i.e. modifying TSF data <D.TSF>) after the PC has been switched off. This security objective exactly counters threat <T.MANAGE>.

<OE.DIRECT> claims, that software, which does not use the respective Application Programming Interface of the OS platform, is not installed or executed while the TOE is installed on the PC. This security objective exactly counters the threat <T.DIRECT>.

<OE.INTRUD> claims, that any intruder <S.UNAU> will not succeed in placing non-trusted software on the hard disk of the PC to attack TSF data <D.TSF>. This security objective exactly counters the threat <T.DIRECT>.

<OE.PASSW> claims, that no unauthorised individual <S.UNAU> will be able to get knowledge of a password of any authorised individual <S.USER>. This security objective exactly counters the threat <T.PASSW>.

<OE.INST> claims, that the TOE is installed and configured with proper installation options and security attributes. This security objective exactly covers the assumption <A.INST>.

9.2 Security Requirements Rationale

It shall be demonstrated that the set of *Security Requirements* (TOE and environment, chapter 6) is suitable to meet and traceable to the *Security Objectives* (chapter 5).

9.2.1 Security Functional Requirements

The *TOE Functional Requirements* (section 6.1) can be mapped to the *TOE Security Objectives* (section 5.1) as follows:

SFR	<O.ACCESS>	<O.MANAGE>
FCS_CKM.1	X	X
FCS_CKM.4	X	X
FCS_COP.1(1)	X	
FCS_COP.1(2)	X	
FCS_COP.1(3)		X
FDP_ACC.1(1)	X	
FDP_ETC.1	X	

SFR	<O.ACCESS>	<O.MANAGE>
FDP_ITC.1	X	
FDP_ACF.1(1)	X	
FIA_UID.2	X	X
FIA_UAU.2	X	X
FMT_SMR.1	X	X
FMT_MOF.1		X
FMT_MSA.1		X
FMT_MSA.2		X
FMT_MSA.3(1)	X	
FMT_MTD.1		X

The table shows, that each TOE Security Objective is implemented by more than one SFR. The table also shows, that each SFR addresses at least one TOE Security Objective.

Detailed Explanation / Justification:

The goal to protect user data on the hard disk <O.ACCESS> is primarily addressed by the functional requirement *Subset access control (FDP_ACC.1)(1) (Data Protection)*. Access control is implemented by the means of cryptographic operations *Cryptographic operation (FCS_COP.1)(1)* or *Cryptographic operation (FCS_COP.1)(2)*. The SFRs *Export of user data without security attributes (FDP_ETC.1)* and *Import of user data without security attributes (FDP_ITC.1)* ensure that all data written onto the hard disk is encrypted (and all data read from hard disk is decrypted) with the named cryptographic operations. This is supported by *Static attribute initialisation (FMT_MSA.3)*, which forces all data on hard disk to be encrypted. The cryptographic operations are supplied with keys generated by *Cryptographic key generation (FCS_CKM.1)* and destroyed by *Cryptographic key destruction (FCS_CKM.4)*.

All authorised users are assigned the role *authorised user*, which is defined by the SFR *Security roles (FMT_SMR.1)*. Access is restricted to authorised users by SFR *Security attribute based access control (FDP_ACF.1)(1)*. Authorised users are determined with the help of the SFRs *User identification before any action (FIA_UID.2)* and *User authentication before any action (FIA_UAU.2)*.

The goal to protect the TOE management operations <O.MANAGE> (changing the protection status of the TOE or modifying other TSF data) is addressed by the SFR *Management of security functions behaviour (FMT_MOF.1)*. This function relies on the role *authorised user*, which is defined by the SFR *Security roles (FMT_SMR.1)*. Authorised users are determined with the help of the SFRs *User identification before any action (FIA_UID.2)* and *User authentication before any action (FIA_UAU.2)*.

TSF data includes security attributes, which are supported and maintained by the SFRs *Management of security attributes (FMT_MSA.1)* and *Secure security attributes (FMT_MSA.2)*. TSF data is secured by the SFR *Management of TSF data (FMT_MTD.1)*. To secure TSF data against unauthorised access, the cryptographic operation *Cryptographic operation (FCS_COP.1)(3)* is used, which is supplied by a key generated by *Cryptographic*

key generation (FCS_CKM.1) and destroyed by *Cryptographic key destruction (FCS_CKM.4)*.

9.2.2 Security Requirements for the Environment

The *Security Requirements for the IT Environment* (section 6.3) and the *Security Requirements for the Non-IT Environment* (section 6.4) response to the security problem defined in form of *Security Objectives for Environment* (section 5.2) as follows.

Objective	Requirement
<OE.INST>	<R.INST>
<OE.INTRUD>	FDP_ACC.1(2) and FDP_ACF.1(2) and FMT_MSA.3(2)[2] and <R.NOSHAR> and <R.BOOTP>
<OE.DIRECT>	<R.DIRECT>
<OE.PASSW>	<R.PASSW> and <R.CAPTKEY>

The table shows, that each Security objective for the environment is covered by a single requirement or a combination of requirements for IT environment rsp. for Non-IT requirement.

The table also shows, that each requirement is necessary to match the security objectives for the IT environment.

Detailed Explanation / Justification:

To install the TOE properly (<OE.INST>), the measures listed in the requirement <R.INST> have to be regarded.

To prevent the intruding of non-trusted software (<OE.INTRUD>), which may disclose passwords or modify security attributes of the TOE, two requirements are defined: Unauthorised users shall not be able to circumvent the TOE's authentication functionality by booting from a different device than the internal hard disk. This is formulated in the requirement <R.BOOTP>. Additionally, unauthorised users shall not have access to the PC via network or modem lines, when the PC is in operational state and is connected to other IT systems. This is formulated in the requirements FDP_ACC.1(2), FDP_ACF.1(2) and <R.NOSHAR>, where it is assumed, that the environment assures, that no user has access via network connections to the local hard disk(s) of the secured PC.

To prevent from the usage of software, which does not use the disk access API of the OS and therefore circumvents the access control function of the TOE (<OE.DIRECT>), the users are instructed not to install such software on the secured PC (<R.DIRECT>).

Disclosure of the password can be performed by three different possibilities:

[2] see note in para 6.3.1.3 this document

- An authorised user may write down his password or may tell his password to an unauthorised individual. This shall be avoided by following the instruction to keep the password secret as defined in <R.PASSW>.
- An electronic device is inserted by an attacker somewhere between the keyboard and the PC keyboard processor. This device is capable of recording the keystrokes and can be removed later and its memory can be read out. To avoid this, the PC must be secured in a way, that the insertion of such a device is not possible or that it can be easily detected by the user. This requirement is defined in <R.CAPTKEY>.
- The room, where the PC is operated and the user is entering his password, is supervised by a video camera and the image is recorded. In this way, an unauthorised individual could get information about the user's password. To avoid this, the PC environment must be checked, that such a supervision is not present or can not record the user's keyboard entries. This requirement is also defined in < R.CAPTKEY>.

Both requirements together avoid the disclosure of the user's password <OE.PASSW>.

9.2.3 Assurance Requirements and Strength of Security Functions

The *TOE Assurance Requirements* (section 6.1.4.6) cover all aspects to ensure that the security functions provided by the TOE are actually able to respond to the security problems defined in form of *TOE Security Objectives* (section 5.1). The assurance requirements are exactly those defined for the Evaluation Assurance Level 1. So, there is no need to further demonstrate that these requirements are useful and suitable.

A claimed rating of the minimum strength of security functions is not applicable for the Evaluation Assurance Level 1.

9.3 Dependency Rationale

9.3.1 Functional Requirements Dependencies

The following table show, all functional requirements dependencies required by the TOE and IT-environment.

[(1)– components of the TOE ; (2) – components of the IT-environment, para. numbers in parenthesis refer to the approriate paragraph in this document]

Component	Dependencies	Dependency fulfilled by
TOE security functional components		
FCS_CKM.1	FCS_COP.1	FCS_COP.1 (6.1.1.3, 6.1.1.4, 6.1.1.5)
	FCS_CKM.4	FCS_CKM.4 (6.1.1.2)
	FMT_MSA.2	FMT_MSA.2 (6.1.4.4)
FCS_CKM.4	FCS_CKM.1	FCS_CKM.1 (6.1.1.1)
	FMT_MSA.2	FMT_MSA.2 (6.1.4.4)

Component	Dependencies	Dependency fulfilled by
FCS_COP.1	FCS_CKM.1 FCS_CKM.4 FMT_MSA.2	FCS_CKM.1 (6.1.1.1) FCS_CKM.4 (6.1.1.2) FMT_MSA.2 (6.1.4.4)
FDP_ACC.1(1)	FDP_ACF.1	FDP_ACF.1(1) (6.1.2.2)
FDP_ACF.1(1)	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1(1) (6.1.2.1) FMT_MSA.3 (1) (6.1.4.5)
FDP_ETC.1	FDP_ACF.1	FDP_ACF.1(1) (6.1.2.2)
FDP_ITC.1	FDP_ACF.1 FMT_MSA.3	FDP_ACF.1(1) (6.1.2.2) FMT_MSA.3(1) (6.1.4.5)
FIA_UID.2	none	---
FIA_UAU.2	FIA_UID.1	FIA_UID.2 (6.1.3.1)
FMT_SMR.1	FIA_UID.1	FIA_UID.2 (6.1.3.1)
FMT_MOF.1.	FMT_SMR.1	FMT_SMR.1 (6.1.4.1)
FMT_MSA.1	FMT_SMR.1 FDP_ACC.1	FMT_SMR.1 (6.1.4.1) FDP_ACC.1(1) (6.1.2.1)
FMT_MSA.2	FDP_ACC.1 FMT_MSA.1 FMT_SMR.1 ADV_SPM.1	FDP_ACC.1(1) (6.1.2.1) FMT_MSA.1 (6.1.4.3) FMT_SMR.1 (6.1.4.1) see note 1 in para. 6.1.4.4 this document
FMT_MSA.3(1)	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 (6.1.4.3) FMT_SMR.1 (6.1.4.1)
FMT_MTD.1	FMT_SMR.1	FMT_SMR.1 (6.1.4.1)
IT-environment security functional components		
FDP_ACC.1(2)	FDP_ACF.1	FDP_ACF.1(2) (6.3.1.2)
FDP_ACF.1(2)	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1(2) (6.3.1.1) see note in para 6.3.1.3 this document

9.3.2 Assurance Requirements Dependencies

The following table shows, that all assurance requirements dependencies are fulfilled.

Assurance Req.	Dependencies	Dependency fulfilled by
ACM_CAP.1	none	---
ADV_FSP.1	ADV_RCR.1	ADV_RCR.1
ADV_RCR.1	none	---

ADO_IGS.1	AGD_ADM.1	AGD_ADM.1
AGD_ADM.1	ADV_FSP.1	ADV_FSP.1
AGD_USR.1	ADV_FSP.1	ADV_FSP.1
ATE_IND.1	ADV_FSP.1 (#2) AGD_ADM.1 (#5) AGD_USR.1 (#6)	ADV_FSP.1 AGD_ADM.1 AGD_USR.1

9.4 TOE Summary Specification Rationale

9.4.1 Satisfaction of Functional Requirements

The TOE Summary Specification Rationale shows, that the set of *TOE Security Functions* (as described in section 7.1) is working together to satisfy the *TOE Functional Requirements* (section 6.1). The following table describes the references between the *TOE Functional Requirements* and the *TOE Security Functions*:

SFR	<SF1>	<SF2>	<SF3>
FCS_CKM.1			X
FCS_CKM.4			X
FCS_COP.1(1)		X	
FCS_COP.1(2)		X	
FCS_COP.1(3)			X
FDP_ACC.1(1)		X	
FDP_ACF.1(1)		X	
FDP_ETC.1		X	
FDP_ITC.1		X	
FIA_UID.2	X		
FIA_UAU.2	X		
FMT_SMR.1	X		X
FMT_MOF.1			X
FMT_MSA.1	X		X
FMT_MSA.2	X	X	X
FMT_MSA.3(1)			X
FMT_MTD.1	X		X

Detailed explanation:

FCS_CKM.1
The generation of the hard disk encryption key for protection of the user data and the generation of the system key for protection of the TSF data is provided by a protpietary digital random number generator integrated in the administration program and invoked during system installation and first administration (security function <SF3>)

FCS_CKM.4
The overwriting of the (encrypted) system key and the (encrypted) hard disk encryption key is done during deinstallation of the TOE by the administration program (security function <SF3>).

FCS_COP.1(1) and FCS_COP.1(2)
The cryptographic operations for encrypting and decrypting user data are included according to DES rsp. IDEA standard into the driver components and transparently encrypt and decrypt user data on the hard disk as a part of <SF2>.

FCS_COP.1(3)
The cryptographic operation for securing TSF data is included into the administration program and encrypts the TSF data in the system kernel on the hard disk; this is part of <SF3>.

FDP_ACC.1
Access control to user data on hard disk is simply reached by encrypting the user data. Only a user with successful identification and authentication can gain access to the key, which allows decryption and hence access to the user data. This is part of <SF2>.

FDP_ACF.1
The security attribute *UserDataEncryptionType* has always the value “encrypted”, when the TOE is properly installed. Encryption and decryption of user data is performed within <SF2> of the TOE.

FDP_ETC.1 and FDP_ITC.1
The export and import of user data is only possible, when the encryption/decryption mechanism of <SF2> is active and the user has access to his data. Export of user data does not maintain any security attributes with the data. Import of user data does not maintain security attributes with the data and does unconditionally encrypt the data after import. This is part of <SF2>.

FIA_UID.2 and FIA_UAU.2
During Pre Boot Authentication (<SF1>), the user has to enter a valid user name and the corresponding password. Otherwise the hard disk encryption key cannot be gained and the operating system can not be booted. The identification and authentication must be done prior to any other system operation.

FMT_SMR.1
As stated within the definition of subject <S.USER>, there is only one role maintained by the TOE, which is the *authorised user*. Each user logging in at PBA (<SF1>) is assigned to this role. This role has all rights for TOE management functions (<SF3>), especially the right of deinstallation.

FMT_MOF.1
The right for deinstallation of the TOE is restricted to the *authorised user*, because only this role has access to the management operations (<SF3>), which comprise the deinstallation function.

FMT_MSA.1
The security attribute *UserDataEncryptionType* can only be modified by *authorised user*

when running the administration program and deinstalling the TOE. This is part of <SF3>.
The security attributes *user name* and *password* can be modified by *authorised user* when running the administration program (with the restriction, that the user name for “system” can not be modified). This is part of <SF3>. The security attribute *password* can additionally be modified after user login at PBA, but only by the affected user himself (hence by an *authorised user*). This is part of <SF1>.FMT_MSA.2
The security attribute *UserDataEncryptionType* has always the value “encrypted”. This is guaranteed by <SF2>, which can not be disabled, except by complete hard disk decryption and deinstallation of the TOE. As long as the TOE is installed, <SF2> is active and enforces encryption.
The values for the security attribute *user name* can not be classified as secure vs. insecure, so this attribute is not affected by this SFR.
The secure values for the security attribute *password* are defined by internal checking algorithms on weak passwords and on minimum password length. These checks are included in every function, where passwords can be modified, i.e. in <SF1> and <SF3> (see explanation of FMT_MSA.1 above).

FMT_MSA.3
The security attribute *UserDataEncryptionType* is initialised with the value “encrypted” during the initial hard disk encryption after the installation of the TOE. This is part of <SF3>.
The security attribute *user name* has an initial value for the two predefined users during standard installation of the TOE. For all further users there is no initial value for the attribute *user name*, as it has always to be specified by the user creating it. This is part of <SF3>
The security attribute *password* has no initial value and has always to be specified by the authorised user (for the predefined users during installation of the TOE). This is part of <SF3>.

FMT_MTD.1
The user management including changing of passwords is part of the administration program and therefore included in <SF3>. The management of the encrypted partitions and the management of the boot options is also is part of the administration program and therefore included in <SF3>. Only an *authorised user* had access to the administration program and can modify user accounts and passwords.
A user’s password can also be modified after a successful login to PBA (<SF1>).

9.4.2 Mutual Support of Security Functions

The security functions <SF1>, <SF2> and <SF3> mutually support each other.

They support the only security role *authorised user*. Only authorised users can pass *Pre Boot Authentication (PBA) <SF1>*, which is the only instance to decrypt the hard disk key. This is required to operate *Protection of Data on Hard Disk Partitions <SF2>* correctly and give the user access to the user data on hard disk.

The security function *Installation and Secure Administration <SF3>* supports all other security functions. Administrative operations can be performed again by the role *authorised user* only, which is guaranteed by *Pre Boot Authentication (PBA) <SF1>* and by the administration program login.

As shown in the section before, there is a complete and sufficient mapping between the Security Functions and the Security Functional Requirements. Therefore no additional functionality is required to meet the Security Functional Requirements.

9.4.3 TOE Assurance Requirements

The *TOE Assurance Requirements* (section 6.2) cover all aspects to ensure that the security functions provided by the TOE are actually able to response to the security problems defined in form of *TOE Security Objectives* (section 5.1). The assurance requirements are exactly those defined for the Evaluation Assurance Level 1. The documentation provided by the sponsor as listed in the table in section 6.2 describes, that the assurance requirements are properly fulfilled.

The TOE itself does not provide any measure or mechanism to satisfy the assurance requirements.

9.5 PP Claims Rationale

This Security Target does not make any claim that the TOE conforms with the requirements of a *Protection Profile*. As a result the chapter *PP Claims Rationale* is omitted.

10 Glossary

CC	Common Criteria
DES	Data Encryption Standard
EFS	Extended File System (Microsoft Windows NT/2000 File System)
FAT	File Access Table (Microsoft DOS/Windows File System)
IDEA	International Data Encryption Algorithm
LAN	Local Area Network
NTFS	New Technology File System (Microsoft Windows NT/2000 File System)
OS	Operating System
PBA	Pre Boot Authentication
PP	Protection Profile
SGE	SafeGuard Easy
TOE	Target of Evaluation
WAN	Wide Area Network

11 References

[CC1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999

[CC2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999

[CC2A] Common Criteria for Information Technology Security Evaluation, Part 2, Annexes; Version 2.1, August 1999

[CC3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999