Becrypt Trusted Client v2.3

Security Target

EAL2

Version 1.1

November 2009

Document History

Version	Date	Author	Description
1.0	20-Oct-09	Ben Bromhead	Final Version
1.1	1-Nov-09	Ben Bromhead	Updated typographic errors

Table of Contents

List of Tables

1 Document introduction

1 This section provides preliminary information and documenting conventions which are used to present the Security Target (ST) to the reader, as well as other information which aims at assisting the reader in understanding the ST and the TOE it describes.

1.1 TOE overview

2 Trusted Client provides a low cost highly secure mobile access to networks and data, drastically reducing the risk of data loss and insecure access. Trusted Client provides ***data protection*** functionality for user data and TSF-data at rest and for transferring user data between the trusted environment and an authorised IP address.

3 Delivered through a CD-ROM and installed on a USB token, Trusted Client allows remote workers to access their network from any Internet enabled PC by simply inserting the USB token containing the Trusted Client and booting up the machine. A secure isolated environment is created on the Host PC: all data including the operating system is encrypted; critical applications and data can be accessed in confidence. Once removed from the Host PC no trace of Trusted Client remains.

4 The following are the key characteristics of Becrypt's Trusted Client:

a) Trusted Client is a bootable trusted environment that resides on a USB flash drive.

b) Trusted Client devices are encrypted to protect data stored on the USB flash drive.

c) Self-tests ensure the integrity of the operating system.

d) The Trusted Client is password protected to prevent unauthorised access.

e) When inserted into an unmanaged computer and booted up, Trusted Client launches a self-contained environment, entirely separate from its temporary host's operating system or hard drive, and leaves nothing behind when the session ends.

f) The trusted environment typically provides a simple user interface, a web browser, and optional extra functionality, including thin clients, email access, and stand-alone applications.

g) When used with a third party VPN, such as Juniper or AEP, Trusted Client provides secure thin client functionality that may be used to remotely access enterprise applications.

h) Configurable security features prevent accidental or deliberate misuse of the Trusted Client device by the authorised user.

i) All user data saved to a Trusted Client device is automatically encrypted.

j) Trusted Client devices may optionally be centrally managed, providing an audit trail, and allowing the remote decommissioning of devices.

1.2 Document conventions

5 Part 2 of the Common Criteria defines an approved set of operations that may be applied to the statement of security functional requirements. Following are the approved operations and the document conventions as used within this ST to depict their application:

a) **Assignment.** The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows [**assignment**].

b) **Selection.** The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows [***selection***].

c) **Refinement.** The refinement operation allows the addition of extra detail to a requirement. Refinements are indicated using bolded text, for **additions**, and strike-through, for ~~deletions~~.

d) **Iteration.** The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing a letter at the end of the component identifier as follows FDP_IFF.1a and FDP_IFF.1b.

e) **Application note.** An informal explanation by the author of the ST to highlight and explain an unusual or otherwise exceptional wording either in the requirements for an artefact of the ST or in the statement of a specific artefact in the ST.

1.3 Terminology

6 The essential terminology used in this ST is described in Table 1.

Table 1 – Terminology

Term	Description
Access Software	Software running on the Host PC or on a PC connected to the Host PC used for accessing the TOE.
Common Access Card (CAC)	The U.S. DoD smart card for authentication of personnel.
EEPROM	Electrically Erasable Programmable Read-Only Memory.
Execution Domain	A logically separate environment in which applications can be executed independently of other applications being executed in different execution domains.
Host Device	The USB Device on which the TOE is installed and resides on, and using which the TOE is connected to the Host PC.
Host PC	The PC to which the TOE is connected and whose peripheral devices (keyboard, screen, etc.) the TOE shares and from which the TOE is powered.
Shareability	The property of a data storage or other device being usable by more than one end user either simultaneously or one at the time.
USB flash drive	A USB-port powered mass memory device that uses Flash memory technology.
User Data	Files stored on the TOE by the end user, including application data and security parameters associated to the end user.

1.4 References

[1] Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model, version 3.1 Revision 1, September 2006, CCMB-2006-09-001.

[2] Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, version 3.1 Revision 2, September 2007, CCMB-2007-09-002.

[3] Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, version 3.1 Revision 2, September 2007, CCMB-2007-09-003.

1.5 Document organisation

7 This document is organised into the following sections:

a) Section 1 provides introductory and preliminary explanations and document conventions to assist readers in understanding this ST.

b) The assurance families required for fulfilling assurance class ASE (ST Evaluation) at EAL2, excluding the rationales, are covered as follows:

i) ASE_CCL.1 (Conformance claims) in Section 3.

ii) ASE_ECD.1 (Extended components definition) In Section 6.

iii) ASE_INT.1 (ST introduction) in Section 2.

iv) ASE_OBJ.2 (Security objectives) in Section 5.

v) ASE_REQ.2 (Derived security requirements) in Section 7.

vi) ASE_SPD.1 (Security problem definition) in Section 4.

vii) ASE_TSS.1 (TOE summary specification) in Section 8.

c) The rationales as all presented centrally in Section 9.

2 ST introduction

8 This section identifies the ST and describes the TOE in a narrative manner.

2.1 ST reference

9 The ST reference that uniquely identifies the ST is a combination of the TOE title and document version, the values of which are stated in Table 2.

Table 2 – ST reference information

ST Title	Becrypt Trusted Client v2.3 EAL2 Security Target
ST Version	1.1 (4-Nov-09)

2.2 TOE reference

10 The TOE reference is used to uniquely reference the TOE is a combination of product version, TOE Evaluation Assurance Level (EAL), and CC version identification, the values of which are stated in Table 3.

Table 3 – TOE identification information

TOE Version	Becrypt Trusted Client v2.3
EAL	EAL2
CC Version Identification	Common Criteria for Information Technology (IT) Security Evaluation, Version 3.1 as stated in [1], [2], and [3].

2.3 TOE overview

2.3.1 Usage and major security features of the TOE

11 Becrypt Trusted Client provides end users with a method for gaining remote secure access to their organisation's network from any computer in any location. The main advantages are related to business continuity and accessing organisational resources remotely in a secure manner even in the absence of a trustworthy computing environment within the remote end user location.

12 The Trusted Client is a bootable trusted environment that resides on a USB flash drive called the Host Device. The user data on the FLASH memory of the device is encrypted to ensure confidentiality of the user data. The device is password protected to ensure that only legitimate and authenticated users are able to gain access to the data and functions on the device. This applies only when no "non-secure" partition has been configured during device setup.

13 The Host Device is to be inserted into the Host PC prior to the booting up of the PC. Upon booting up of the Host PC, the Trusted Client launches a self-contained trusted execution environment for applications. This environment is separate from the operating system or hard drive of the Host PC, and leaves no traces to the hard disk of the host PC when the session ends.

14 In this self-contained environment, the TOE executes a Linux based light weight operating system. Inserting the Host Device into a computer that is already booted up causes no action as the device is not recognised by the operating system of the host PC.

15 Once the TOE is booted up, it provides a trusted environment for executing applications. These applications, however, are not part of the TOE. Typical applications include stand-alone applications, a simple user interface, a web browser, and thin clients for enterprise services such as email or VPN.

16 The Becrypt Trusted Client implements the security features and characteristics summarised in Table 4.

Table 4 – Trusted Client security features and characteristics

Feature	Characteristics
Trusted environment and isolation	Trusted Client is a bootable trusted environment that can be run on any unmanaged, untrusted platform bootable from USB drive. The Trusted Client consists of a lightweight Operating System, a web browser, and optional additional components (such as an SSL VPN, thin client applications, an email client), all of which are written to and reside on the Host Device. The drive provides a Linux based mini operating system that is isolated from the hard disk of the Host PC and thus protects the sensitive user data from exposure to the potentially hostile Host PC. The operating system is part of the TOE but the host device, the web browser and the thin clients as well as other applications residing on the host device are not part of the TOE.
Portability	Data managed by the TOE when residing on the host device is protected by AES encryption and by user authentication. As such, the host device of the Trusted Client may safely be removed from trusted premises and transported by any authorised user. If the device is lost or stolen, the data stored thereon remains protected. Unauthorised users who may gain physical possession of the host device shall not be able to access the TOE without breaking the encryption or authentication scheme.
Encryption and authentication	All sensitive data on the Trusted Client, including the operating system files, is protected by encryption and strong authentication. Only if the user enters a correct username and password will the Trusted Client begin decryption and launch the trusted environment. Any attempt to boot from the device will result in Trusted Client requesting the user to authenticate. Access to the data on the device and creation of a trusted environment only occurs upon successful authentication. If the user inserts the host device of the Trusted Client into a booted machine, the operating system of that machine will assume that the device is unformatted (because it is encrypted and does not appear as a USB drive) and prompts the user to format it. All data residing on the device will be lost if the formatting takes place. The authentication policy may be strengthened to enforce strong passwords or dual-factor authentication. Alternatively, the user may be forced to use the strong password generator included in the Trusted Client. Neither the dual-factor authentication nor the device-generated passwords are part of the TOE.

Feature	Characteristics
Dual factor authentication	Optionally, the Trusted Client supports additional authentication using the US Department of Defence issued Common Access Card. This allows a dual factor authentication and further ensures that password guessing attacks shall not succeed in circumventing the end user authentication. Dual factor authentication is not part of the TOE and, hence, excluded from the scope of evaluation.
Trusted configuration	Configuration files of the TOE stored on the device are encrypted and cannot be altered by unauthorised users.
Device recovery	Forgotten passwords can be recovered through an administrator-assisted challenge-response mechanism by which access may be regained. The mechanism uses specific recovery data generated during the initiation of the TOE. The original password shall never be exposed during the recovery process.
Clone protection	The Trusted Client device automatically performs a self-test during boot-up. If it fails the clone test, the device erases the data thus preventing its execution.
Management	If so configured, the Trusted Client devices automatically contact the management server on boot-up (provided that a connection can be established) so that the boot-up details may be viewed by the administrator. If the device has been marked for revocation, it is sent a decommission message which causes itself to immediately erases itself. These management features are not part of the TOE and are, hence, excluded from the scope of evaluation.

2.3.2 TOE Type

17 The TOE is not of any type defined in CC Part 1.

18 The TOE is categorised as a Data Protection product.

19 The TOE is a software product installed on a USB Token for secure storage of files and for booting up a host PC into a trusted execution domain and executing applications in a secure manner even if inserted into a potentially hostile host PC. Access to the secure domain created by the TOE is only granted to authentic and authorised end users.

2.3.3 Hardware, software and firmware required by the TOE

20 The TOE is software installed on a USB token constituting the Host Device. The Host Device itself is not part of the TOE and requires a Host PC for power and connectivity. The TOE communicates with the operating system of the Host PC through a USB connection of the Host Device connecting the Host Device into the Host PC. Both USB 1.1 and USB 2.0 are supported.

21 The Host PC must run BIOS that controls the USB interface. The platforms supported are any general x86 PC platforms with either Intel or AMD processors that allow booting from USB devices.

22 The TOE also requires the presence of thin clients for the critical applications. The thin clients that are executed on the trusted domain created by the TOE are typically developed by third parties and as such, are not part of the TOE.

23 Additionally, some administrative features of the TOE are only available in the presence of the central management software.

2.4 TOE description

2.4.1 Physical scope of the TOE

24 The TOE is a software product housed in the Host Device. The TOE interacts with the host PC and with the hardware of the Host Device. Application level software executables residing in the Host Device (such as thin clients, email clients and VPN software) are not part of the TOE.

25 In addition to the Host Device hosting the TOE, the TOE also requires presence of a Host PC which can be any PC running a supported operating system. Neither the Host PC nor the operating system of the Host PC nor any application level software of the host PC is part of the TOE.

26 The Trusted Client implements support for a Common Access Card (CAC) based authentication. However, both the authentication function and the CAC itself are outside the scope of the TOE. Any hardware infrastructure needed for accessing the CAC in the Host PC is also outside the scope of the TOE.

27 The TOE implements support for deploying administrative software, however, this software is not part of the TOE. Neither is any network connectivity required for the administrative software to communicate with the TOE.

2.4.2 Logical scope of the TOE

28 The TOE provides the essential security functionality of the Becrypt Trusted Client.

29 A high level view of the TOE is given in Figure 1. The TOE is connected to the Host PC over the USB interface and connectivity hardware provided by the Host Device.

30 The initial connection of the Host PC and the TOE is with the boot-loader which performs the TOE boot-up sequence and provides the initial user authentication and, upon successful user authentication, sets the authentication state. If the authentication state is set, i.e. the authentication was successful; the AES key persistently stored on the FLASH memory of the Host Device is unlocked and made accessible to the cryptographic module.

31 Once the AES key is accessible, the OS files persistently stored on the FLASH memory of the Host Device are decrypted and installed on the TOE where they constitute the OS. The decryption of the OS files ensures that their integrity remains of good quality and the OS can be trusted to not allow flow of information to the hard disk of the local PC. Furthermore, once the OS requires access to the user data, as long as the authentication state is set, the cryptographic module has access to the AES key and the user data stored on the FLASH memory of the Host Device is decrypted prior to being passed to the OS of the TOE and encrypted once forwarded to the FLASH memory for persistent storage.

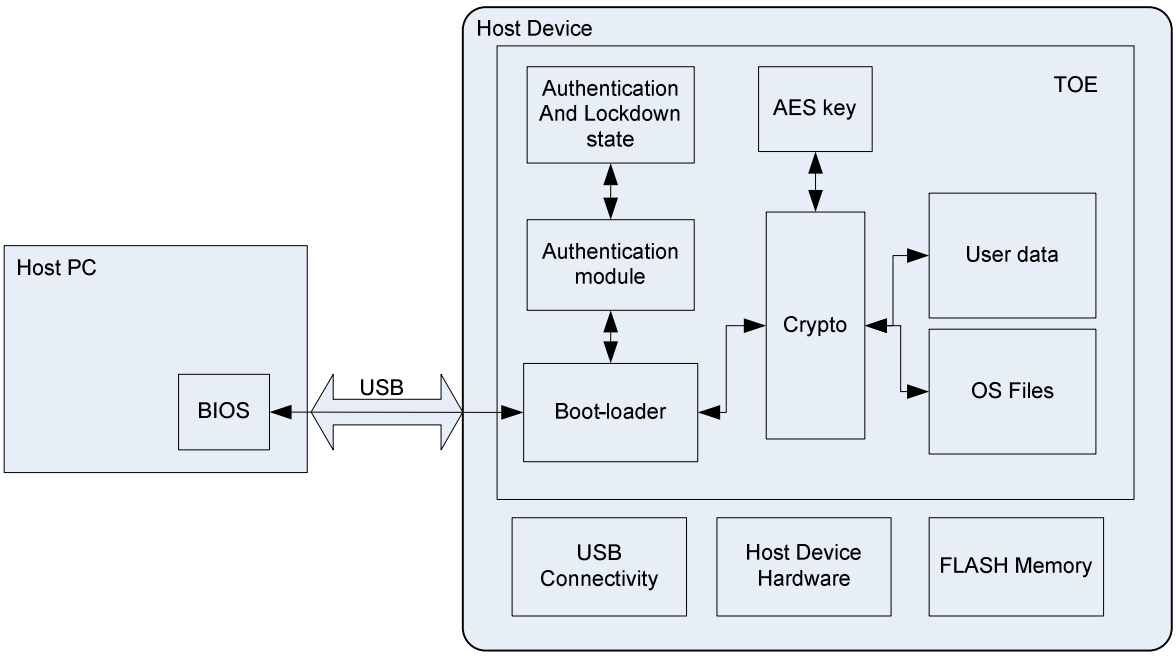

Figure 1 – TOE system architecture

32 Fundamentally, the TOE is the embedded operating system in which thin clients execute and which provides the fundamental support services to authenticate users and encrypt and decrypt the data on the FLASH memory. It is a multi-user, single-thread operating system focused on the very core execution services to applications.

33 The human user of the TOE can establish a session with the TOE by booting the Host PC from the Host Device. The session is initialised by authenticating the human user using a username-password combination. Upon successful authentication, the critical OS executables and configuration files are decrypted and the light weight OS residing on the TOE booted up.

34 The TOE maintains a counter of consecutive, unsuccessful authentication attempts. If the value exceeds a threshold defined in the TOE configuration, the TOE enters a Lockdown state. The Lockdown state is a response to a suspected password guessing attack in which all access requests are denied. The device can be recovered from the Lockdown state through the device recovery function available to the users with administrative privileges.

35 Each time the TOE initialises, it ensures that no sensitive information or data remains in the OS structures from the previous session and executes the clone control checks to ensure that it is a legitimate TOE. If the tests fail, the TOE determines it is an illegitimate clone of a legitimate TOE. Upon this determination, the TOE triggers erasure of critical files to ensure it cannot be successfully booted.

36 Once the TOE is successfully initialised, the light weight applications can be used for accessing the files on the token and back-end services (if a network connection is present). The files cannot be copied to the hard disk of the Host PC and if the network connection is present, connections can only be established to pre-approved IP addresses. The light weight applications are not part of the TOE, though.

37 The human user can terminate the session by issuing a shutdown command – controlled session termination. The session also terminates upon loss of power to the TOE, when it is removed from the Host PC or when the Host PC looses power. This type of termination is called uncontrolled session termination.

38 Upon controlled session termination, all decrypted OS and configuration files, as well as any sensitive OS data of temporary nature, are cleared to ensure that only upon

successful authentication can the TOE be rebooted and that no files stored on the FLASH memory are accessible to the host PC.

39 If the power is lost unexpectedly, no TOE functions can be executed upon uncontrolled session termination. Upon being re-powered, the duration of the power-down cannot be determined and there is no assurance that the human user of the TOE remains authentic. Therefore, the boot-up procedures of the TOE includes mechanisms for clearing any sensitive data remaining in the TOE data structures, even if they may have been sufficiently cleared upon controlled session termination.

40 The TOE can be managed externally. In this case, a secure session is established between the TOE and a management system residing in the Host PC. The TOE is capable of identifying a legitimate management system and restricting access to the management of the TOE only to the management systems whose authenticity has been successfully verified.

3 Conformance claims

41 The following conformance claims are made for the TOE and ST:

a) **CCv3.1 Rev.2 conformant**. The ST and the TOE are Common Criteria conformant to Common Criteria version 3.1 Revision 2 defined in [1], [2] and [3].

b) **Part 2 conformant**. The ST is Common Criteria Part 2 conformant.

c) **Part 3 conformant.** The ST is Common Criteria Part 3 conformant.

d) **Package conformant.** The ST is package conformant to the package Evaluation Assurance Level EAL2 as defined in [3].

e) **Protection Profile conformance**. The ST does claims conformance to the following Protection Profiles: **None**.

4 Security problem definition

42 The TOE is concerned with the protection of the assets enumerated in Table 5.

Table 5 – Assets protected by the TOE

Identifier	Asset statement
AST.USER_DATA	Confidentiality of the user data stored on the device.
AST.OS_FILES	Integrity and authenticity of the operating system files (both executables and configuration files) persistently stored on the TOE.
AST.BOOT	Integrity and authenticity of the boot sequence of the TOE.

43 The subjects, some of which constitute threat agents as highlighted in the description of threats, are stated in Table 6

Table 6 – Subjects relevant to the TOE

Identifier	Subject definition
S.UNKNOWN_SW	Any operating system or application software running on the host PC or on the TOE not known to have a legitimate right to initialise or administer the TOE or to gain access to the user data persistently stored on the TOE.
S.MGMT_SW	Legitimate management software for administering the TOE.
S.HUMAN_USER	The legitimate human user of the TOE.
S.UNKNOWN_USER	A human user of the TOE other than S.HUMAN_USER.

4.1 Threats

44 Threats enumerated in Table 7 are relevant to the TOE.

Table 7 – Threat statements

Identifier	Threat statement
T.CRYPTO	S.UNKNOWN_SW succeeds in violating AST.USER_DATA or AST.OS_FILES by successfully cryptoanalysing the TOE. The cryptoanalysis may result in the disclosure of the cryptographic key used for encrypting the data persistently stored on the TOE or in a discovery of a weakness in the cryptographic primitives or implementation thereof used for protecting the user data or OS files encrypted and persistently stored on the TOE.
T.DATA_LEAK	S.UNKNOWN_SW or S.UNKNOWN_USER compromises AST.USER_DATA by copying data to the host hard drive or external network hosts.
T.AUTH_FAIL	S.UNKNOWN_USER succeeds in violating AST.USER_DATA by successfully masquerading as S.HUMAN_USER by guessing the correct authentication data of S.HUMAN_USER. A successful guess of the authentication data would allow the attacker to successfully masquerade as a legitimate end user of the TOE. This would result on the loss of TOE's ability to differentiate between legitimate and illegitimate end users

Identifier	Threat statement
	and subsequent loss of the confidentiality of the user data stored on the TOE.
T.MANAGEMENT	S.UNKNOWN_SW may compromise AST.OS_FILES allowing an attacker to violate AST.USER_DATA.
T.BOOT	S.UNKNOWN_USER or S.UNKNOWN_SW compromises AST.BOOT allowing an attacker to violate AST.USER_DATA.

4.2 Organisational security policies

45 The following organisational security policies are relevant to the TOE: **None.**

4.3 Assumptions

46 Assumptions enumerated in Table 8 govern the operational environment of the TOE.

Table 8 – Assumption statements

Identifier	OSP statement
A.APPL	The end users of the TOE are aware of the security issues of uploading and executing applications on the TOE and follow the regulations on application management established by the deploying organisation so that only approved applications are installed on the TOE.

5 Security objectives

47 This section states the exact security objectives for the TOE so that the security problem definition is adequately and completely addressed. The security objectives are stated for both the TOE and for the operational environment of the TOE.

5.1 Security objectives for the TOE

48 Security objectives for the TOE are enumerated in Table 9.

Table 9 – Security objectives for the TOE

Identifier	Objective statement
O.CRYPTO	The cryptographic keys, the underlying cryptographic algorithms and other cryptographic primitives are sufficiently secure and cryptographic operations sufficiently protect the user data and security parameters stored on the TOE.
O.AUTH	The TOE can enforce password complexity requirements and maximum failed logon attempts. After an administrator set threshold on the number of failed logon attempts has been reached, the user can only login after administrator intervention or a reboot.
O.ISOLATION	The TOE is isolated from the Host PC and no user data can be copied from the TOE to the hard disk of the Host PC or communicated to an IP address other than an IP address explicitly authorised by the TOE administrator.
O.MANAGEMENT	Only legitimate management software is granted administrative access to the TOE.
O.BOOT	The boot sequence implements the necessary controls to ensure that the boot sequence is authentic.

5.2 Security objectives for the environment

5.2.1 Security objectives for the IT environment

49 Security objectives for the IT environment of the TOE are stated in Table 10.

Table 10 – Security objectives for the IT environment

Identifier	Objective statement
OE.APPL	Only approved applications are installed for execution on the TOE by the end users.

5.2.2 Security objectives for the non-IT environment

50 The following security objectives are stated for the non-IT environment of the TOE:
None.

6 Extended components definition

51 The following extended components are defined in this ST: **None**.

52 There are no extended components applicable to the TOE; hence none of the requirements for the Extended Components Definition (ASE_ECD) are applicable to this ST.

7 IT security requirements

7.1 Overview

53 This section defines the security requirements satisfied by the TOE. Each requirement has been extracted from version 3.1 of the Common Criteria, part 2 providing functional requirements and part 3 providing assurance requirements.

7.2 TOE security functional requirements

54 This section contains the security functional components from part 2 of the Common Criteria with the operations completed.

55 Standard Common Criteria text is in regular black font and the text inserted to perform an operation on the requirement is in accordance with the conventions specified in section 1 of this ST.

Table 11 – Summary of TOE security functional requirements

Identifier	Title
Cryptographic support	
FCS_CKM.1a	Cryptographic key generation (AES)
FCS_CKM.1b	Cryptographic key generation (HMAC)
FCS_CKM.4	Cryptographic key destruction
FCS_COP.1a	Cryptographic operation (Password hashing)
FCS_COP.1b	Cryptographic operation (User data protection)
FCS_COP.1c	Cryptographic operation (OS data protection)
FCS_COP.1d	Cryptographic operation (HMAC)
User data protection	
FDP_ACC.1a	Subset access control (Lockdown SFP)
FDP_ACF.1a	Security attribute based access control (Lockdown SFP)
FDP_ACC.1b	Subset access control (Data Access SFP)
FDP_ACF.1b	Security attribute based access control (Data Access SFP)
FDP_IFF.1	Subset information flow control (HD isolation SFP)
FDP_IFC.1	Simple security attributes (HD Isolation SFP)
FDP_RIP.1a	Subset residual information protection (OS data)
FDP_RIP.1b	Subset residual information protection (Cryptographic keys)
Identification and authentication	
FIA_AFL.1	Authentication failure handling

Identifier	Title
FIA_ATD.1	User attribute definition
FIA_UAU.1	Timing of authentication
FIA_UID.1	Timing of identification
Security management	
FMT_MSA.1a	Management of security attributes (Lockdown SFP)
FMT_MSA.1b	Management of security attributes(Data Access SFP)
FMT_MSA.1c	Management of security attributes (HD Isolation SFP)
FMT_MSA.2	Secure security attributes
FMT_MSA.3a	Static attribute initialization (Lockdown SFP)
FMT_MSA.3b	Static attribute initialization (Data access SFP)
FMT_MSA.3c	Static attribute initialization (HD Isolation SFP)
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security roles
Protection of the TSF	
FPT_FLS.1	Failure with preservation of secure state
FPT_TST.1a	TSF testing (Key generation)
FPT_TST.1b	TSF testing (Start-up)
FPT_TST.1c	TSF testing (Boot)
Trusted Paths/Channels	
FTP_ITC.1	Inter-TSF trusted channel

7.2.1 Cryptographic support

7.2.1.1 FCS_CKM.1a Cryptographic key generation (AES)

Hierarchical to:	No other components.
FCS_CKM.1a.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**AES**] and specified cryptographic key sizes [**256 bits**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 197, "Advanced Encryption Standard (AES)", 26 November 2001**].
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
Notes:	None.

7.2.1.2 FCS_CKM.1b Cryptographic key generation (HMAC)

Hierarchical to:	No other components.
FCS_CKM.1b.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**HMAC-SHA-256**] and specified cryptographic key sizes [**64 bits**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 198-1, "The Keyed-Hash Message Authentication Code (HMAC)", July 2008**].
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
Notes:	None.

7.2.1.3 FCS_CKM.4 Cryptographic key destruction

Hierarchical to:	No other components.
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [**Zeroization**] that meets the following: [**FIPS 140-2 Level 1**].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
Notes:	The crypto officer can zeroise the key. Alternatively, the key is destroyed upon device re-formatting prior to re-initialization.

7.2.1.4 FCS_COP.1a Cryptographic operation (Password hashing)

Hierarchical to:	No other components.
FCS_COP.1a.1	The TSF shall perform [**Password hashing**] in accordance with a specified cryptographic algorithm [**SHA-256**] and cryptographic key sizes [**N/A**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 180-3, "Secure Hash Standard", October 2008**].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
Notes:	None.

7.2.1.5 FCS_COP.1b Cryptographic operation (User data protection)

Hierarchical to:	No other components.
FCS_COP.1b.1	The TSF shall perform [**Encryption and decryption of user data**] in accordance with a specified cryptographic algorithm [**AES**] and cryptographic key sizes [**256 bits**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 197, "Advanced Encryption Standard (AES)", 26 November 2001**].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
Notes:	None.

7.2.1.6 FCS_COP.1c Cryptographic operation (OS data protection)

Hierarchical to:	No other components.
FCS_COP.1c.1	The TSF shall perform [a) **Decryption of executable OS files, and** b) **Encryption and decryption of OS configuration files**] in accordance with a specified cryptographic algorithm [**AES**] and cryptographic key sizes [**256 bits**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 197, "Advanced Encryption Standard (AES)", 26 November 2001**].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
Notes:	The executable OS files are decrypted prior to the boot-up upon successful end

	user authentication. The executable OS files do not change so there is no need to re-encrypt them. Rather, the encrypted OS files installed during the manufacturing stage of the TOE remain encrypted and decryption creates the decrypted executable files without removing the encrypted OS files. The OS configuration files are also decrypted prior to the boot-up upon successful end user authentication. Once the TOE configuration changes, the changes must be reflected in the configuration files. In such a case, the altered configuration files are encrypted and used to replace the earlier configuration files.

7.2.1.7 FCS_COP.1d Cryptographic operation (HMAC)

Hierarchical to:	No other components.
FCS_COP.1d.1	The TSF shall perform [**Verification of the executable cryptographic library code integrity**] in accordance with a specified cryptographic algorithm [**HMAC-SHA-256**] and cryptographic key sizes [**64 bits**] that meet the following: [**Federal Information Processing Standard (FIPS) Publication 198-1, "The Keyed-Hash Message Authentication Code (HMAC)", July 2008**].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
Notes:	None.

7.2.2 User data protection

7.2.2.1 FDP_ACC.1a Subset access control (Lockdown SFP)

Hierarchical to:	No other components.
FDP_ACC.1a.1	The TSF shall enforce the [**Lockdown SFP**] on [**a) Subjects:** **i. Any User.** **b) Objects:** **i. Lockdown State, and** **ii. TOE Object.** **c) Operations:** **i. Any access.**].
Dependencies:	FDP_ACF.1 Security attribute based access control
Notes:	None.

7.2.2.2 FDP_ACF.1a Security attribute based access control (Lockdown SFP)

Hierarchical to:	No other components.
FDP_ACF.1a.1	The TSF shall enforce the [**Lockdown SFP**] to objects based on the following: [**a) Any User:** **i) None.** **b) Lockdown State:** **i) Status.** **c) TOE Object:** **i) Identity.**].
FDP_ACF.1a.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**IF** **the Status of the Lockdown State has a value of SET** **THEN** **the following operations are allowed:** **a) Presentation of the TOE Object whose identity is "response code" for recovery from the Lockdown mode.**].
FDP_ACF.1a.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].
FDP_ACF.1a.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules [**None**].

Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
Notes:	None.

7.2.2.3 FDP_ACC.1b Subset access control (Data Access SFP)

Hierarchical to:	No other components.
FDP_ACC.1b.1	The TSF shall enforce the [**Data Access SFP**] on [**a) Subjects:** **i. TOE.** **b) Objects:** **i. User Data.** **c) Operations:** **i. Decrypt,** **ii. Decrypt and release to the Network Interface, and** **iii. Encrypt and store persistently.**].
Dependencies:	FDP_ACF.1 Security attribute based access control
Notes:	None.

7.2.2.4 FDP_ACF.1b Security attribute based access control (Data Access SFP)

Hierarchical to:	No other components.
FDP_ACF.1b.1	The TSF shall enforce the [**Data Access SFP**] to objects based on the following: [**a) TOE:** **i) Authentication State.** **b) User Data:** **i) Request Status, and** **ii) Destination IP Address.**].
FDP_ACF.1b.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**IF** **the value of TOE Authentication State is SET** **THEN** **the TOE may:** **a. Decrypt the User Data with a Request Status value of REQUESTED,** **b. Decrypt and release to the network interface that User Data whose (i) Request Status is of value REQUESTED and (ii)**

	Destination IP Address is of value of APPROVED, and **c. Encrypt and store persistently the User Data that has a Request Status value of REQUESTED.** **].**
FDP_ACF.1b.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].
FDP_ACF.1b.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules [**None**].
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
Notes:	None.

7.2.2.5 FDP_IFC.1 Subset information flow control

Hierarchical to:	No other components.
FDP_IFC.1.1	The TSF shall enforce the [**HD Isolation SFP**] on [**a) Subjects:** **i. Storage media, and** **ii. Communication endpoint.** **b) Information:** **i. User data.** c) **operations:** i. **Be stored at, and** ii. **Be forwarded to.**].
Dependencies:	FDP_IFF.1 Simple security attributes
Notes:	None.

7.2.2.6 FDP_IFF.1 Simple security attributes

Hierarchical to:	No other components.
FDP_IFF.1.1	The TSF shall enforce the [**HD Isolation SFP**] based on the following types of subject and information security attributes: [**a) Storage media:** **i. Locality.** **b) User data:** **i. None.**].
FDP_IFF.1.2	The TSF shall permit an information flow between a controlled subject and

	controlled information via a controlled operation if the following rules hold: [**a) User data may be stored at the Storage Media whose Locality attribute is of value TOE, and** b) **User data may NOT be stored at the Storage Media whose Locality attribute is NOT of value TOE.**].
FDP_IFF.1.3	The TSF shall enforce the [**None**].
FDP_IFF.1.4	The TSF shall explicitly authorise an information flow based on the following rules: [**None**].
FDP_IFF.1.5	The TSF shall explicitly deny an information flow based on the following rules: [**None**].
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
Notes:	The Locality of Storage Media refers to the physical location of the mass storage media. The locality of the Flash memory is TOE as the Flash memory is located within the TOE, and the locality of the HD of the host PC would be the host PC. The only allowed information flows are to the storage media residing at the TOE to ensure that no user data is stored on the HD of the host PC.

7.2.2.7 FDP_RIP.1a Subset residual information protection (OS data)

Hierarchical to:	No other components.
FDP_RIP.1a.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***Deallocation of the resource from***] the following objects: [**a) Authentication data,** **b) Decrypted OS executables,** **c) Decrypted OS configuration files, and** **d) End user session data.**].
Dependencies:	None.
Notes:	None.

7.2.2.8 FDP_RIP.1b Subset residual information protection (Cryptographic keys)

Hierarchical to:	No other components.
FDP_RIP.1b.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***Allocation of the resource to***] the following objects: [**AES Key**].
Dependencies:	None.
Notes:	None.

7.2.3 Identification and authentication

7.2.3.1 FIA_AFL.1 Authentication failure handling

Hierarchical to:	No other components.
FIA_AFL.1.1	The TSF shall detect when [***an administrator configurable positive integer within*** [**a value range one (1) to three (3)**]] unsuccessful authentication attempts occur related to [**user authentication**].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been [***met***], the TSF shall [**enter a lockdown mode**].
Dependencies:	FIA_UAU.1 Timing of authentication
Notes:	None.

7.2.3.2 FIA_ATD.1 User attribute definition

Hierarchical to:	No components.
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual **TOE** users: [**Role**].
Dependencies:	None.
Notes:	The Role attribute is for the human users of the TOE.

7.2.3.3 FIA_UAU.1 Timing of authentication

Hierarchical to:	No other components.
FIA_UAU.1.1	The TSF shall allow [**a) device reset,** **b) returning of error status,** **c) returning version information,** **d) showing status,** **e) device initiation, and** **f) entering a lockdown mode.**] 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
Notes:	None.

7.2.3.4 FIA_UID.1 Timing of identification

Hierarchical to:	No other components.

FIA_UID.1.1	The TSF shall allow [**Initialization of a trusted channel between itself and an entity claiming to be the Protect Management Software**] on behalf of the user to be performed before the user is identified.
FIA_UID.1.2	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
Dependencies:	None.
Notes:	None.

7.2.4 Security Management

7.2.4.1 FMT_MSA.1a Management of security attributes (Lockdown SFP)

Hierarchical to:	No other components.
FMT_MSA.1a.1	The TSF shall enforce the [**Lockdown SFP**] to restrict the ability to [***modify***] the security attributes [**Lockdown Status**] to [**the Becrypt Protect Management software**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	The Management software is allowed to modify the Lockdown Status to recover the TOE from the Lockdown mode.

7.2.4.2 FMT_MSA.1b Management of security attributes (Data Access SFP)

Hierarchical to:	No other components.
FMT_MSA.1b.1	The TSF shall enforce the [**Data Access SFP**] to restrict the ability to [***modify***] the security attributes [**a) Authentication Status, and** **b) Destination IP Address.**] to [**None**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	Authentication Status or the Destination IP Address are used internally by the TOE and are restricted from access by any end user.

7.2.4.3 FMT_MSA.1c Management of security attributes (HD Isolation SFP)

Hierarchical to:	No other components.
FMT_MSA.1c.1	The TSF shall enforce the [**HD Isolation SFP**] to restrict the ability to [***modify***] the security attributes [**Locality**] to [**None**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	None.

7.2.4.4 FMT_MSA.2 Secure security attributes

Hierarchical to:	No other components.
FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for [**a) Authentication status,** **b) Lockdown status, and** **c) Destination IP address.**].
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Notes:	None.

7.2.4.5 FMT_SMR.1 Security roles

Hierarchical to:	No other components.
FMT_SMR.1.1	The TSF shall maintain the roles [**a) End User, and** **b) Becrypt Protect Management Software.**].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
Dependencies:	FIA_UID.1 Timing of identification
Notes:	End User is a role for the human user. All other roles are roles for technical users.

7.2.4.6 FMT_SMF.1 Specification of Management Functions

Hierarchical to:	No other components.
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: [**a) Manage Authentication State,** **b) Manage Lockdown State, and** **c) Modify the set of legitimate IP Addresses.**].
Dependencies:	None.
Notes:	None.

7.2.4.7 FMT_MSA.3a Static attribute initialisation (Lockdown SFP)

Hierarchical to:	No other components.
FMT_MSA.3a.1	The TSF shall enforce the [**Lockdown SFP**] to provide [***permissive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3a.2	The TSF shall allow the [**None**] 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
Notes:	The permissive initial value is when the Lockdown State is not set.

7.2.4.8 FMT_MSA.3b Static attribute initialisation (Data Access SFP)

Hierarchical to:	No other components.
FMT_MSA.3b.1	The TSF shall enforce the [**Data Access SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3b.2	The TSF shall allow the [**None**] 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
Notes:	The restrictive value is that the Authentication state is not set. The restrictive value is that the Legitimacy of a destination IP address is not approved.

7.2.4.9 FMT_MSA.3c Static attribute initialisation (HD Isolation SFP)

Hierarchical to:	No other components.
FMT_MSA.3c.1	The TSF shall enforce the [**HD Isolation SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3c.2	The TSF shall allow the [**None**] 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
Notes:	The restrictive default values are when storage of the User data is only allowed in the Storage media whose Locality is TOE.

7.2.5 Protection of the TSF

7.2.5.1 FPT_FLS.1 Failure with preservation of secure state

Hierarchical to:	No other components.

FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: [**a) Power failure, and** **b) OS integrity failure.**].
Dependencies:	None.
Notes:	A start-up procedure clears the authentication data temporarily stored on the TOE as well as the authentication state each time the TOE is powered up. This ensures a secure start-up when power is lost during authentication. The start-up procedure includes a check of the OS executable and configuration file integrity. If an integrity failure occurs the TOE will not complete start-up.

7.2.5.2 FPT_TST.1a TSF Testing (Key generation)

Hierarchical to:	No other components.
FPT_TST.1a.1	The TSF shall run a suite of self tests [[**Prior to the generation of cryptographic keys**]] to demonstrate the correct operation of [[**The key generation function**]].
FPT_TST.1a.2	The TSF shall provide authorised users with the capability to verify the integrity of [[**The Random Number Generator**]].
FPT_TST.1a.3	The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.
Dependencies:	None.
Notes:	None.

7.2.5.3 FPT_TST.1b TSF Testing (Start-up)

Hierarchical to:	No other components.
FPT_TST.1b.1	The TSF shall run a suite of self tests [***During initial start-up***] to demonstrate the correct operation of [[**a) Cryptographic algorithms, and** **b) Cryptographic parameters.**]].
FPT_TST.1b.2	The TSF shall provide authorised users with the capability to verify the integrity of [[**a) Cryptographic algorithms, and** **b) Cryptographic parameters.**]].
FPT_TST.1b.3	The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.

Dependencies:	None.
Notes:	None.

7.2.5.4 FPT_TST.1c TSF Testing (Boot)

Hierarchical to:	No other components.
FPT_TST.1c.1	The TSF shall run a suite of self tests [***During initial start-up***] to demonstrate the correct operation of [[**OS files**]].
FPT_TST.1c.2	The TSF shall provide authorised users with the capability to verify the integrity of [[**OS files**]].
FPT_TST.1c.3	The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.
Dependencies:	None.
Notes:	Authorised users (i.e. those that know the correct password) can verify the integrity of the stored TSF executable code by rebooting the TOE. If the rebooting is successful, the executable code is intact. Otherwise, there has been an integrity violation.

7.2.6 Trusted Paths/Channels

7.2.6.1 FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to:	No other components.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit [***another trusted IT product***] to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for [**a) Executing Becrypt Protect Management functions on the TOE, and** **b) Triggering device recovery.**].
Dependencies:	None.
Notes:	None.

7.3 TOE security assurance requirements

60 The assurance package for the evaluation of the TOE is Evaluation Assurance Level 2 (EAL2) with no augmentations.

61 EAL2 assurance requirements provide confidence in the security functionality of the TOE by analysis using a functional and interface specification, guidance documentation and the high-level design of the TOE, to understand the security behaviour.

62 The analysis is supported by independent testing of the TOE security functions, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, strength of function analysis, and evidence of a developer search for obvious vulnerabilities.

63 EAL2 also provides assurance through a configuration list for the TOE, and evidence of secure delivery procedures.

64 Table 12 lists the TOE security assurance requirements for this evaluation. Complete details of all assurance components are located in part 3 of the Common Criteria.

Table 12 – Summary of TOE security assurance requirements

Assurance class	Assurance components
Development (ADV)	ADV_ARC.1
	ADV_FSP.2
	ADV_TDS.1
Guidance Documents (AGD)	AGD_OPE.1
	AGD_PRE.1
Life-Cycle Support (ALC)	ALC_CMC.2
	ALC_CMS.2
	ALC_DEL.1
Security Target Evaluation (ASE)	ASE_CCL.1
	ASE_ECD.1
	ASE_INT.1
	ASE_OBJ.2
	ASE_REQ.2
	ASE_SPD.1
	ASE_TSS.1
Tests (ATE)	ATE_COV.1
	ATE_FUN.1
	ATE_IND.2
Vulnerability Assessments (AVA)	AVA_VAN.2

8 TOE summary specification

8.1 Overview

66 This chapter provides the TOE summary specification, a high-level definition of the security functions of the TOE and a summary of how those security functions meet the requirements.

8.2 Security functions

67 The TOE security functions include the following:

a) **Isolation.** The TOE implements a light weight operating system that ensures that the user data files are isolated from the hard disk and the network interface of the host PC. This ensures that no files are stored on the hard disk of the Host PC and that communication of user data stored on the TOE is only allowed to explicitly approved IP addresses.

b) **Protection of the TOE.** The TOE provides the capability to assert its own integrity during the boot-up phase.

c) **Management.** The TOE provides the capability to differentiate between a legitimate management software and untrusted software. Management rights on the TOE are only granted to legitimate administrators and any management access must ensure that only legitimate TOE configurations result.

d) **Cryptography.** The TOE includes a dedicated cryptographic library for 16-bit and 32-bit architectures. The cryptographic functions and their implementations are sound and appropriately engineered to ensure sufficient protection of the resources.

e) **IAA (identification, authentication, access control).** All TOE users are identified and authenticated prior to granting access to protected resources. Each access request is individually evaluated and only those accesses that are explicitly allowed by the access control rules governing the TOE are permitted. The TOE can also detect a potential password guessing attack by maintaining a counter for recording the number of consecutive authentication failures, and if the number exceeds a defined threshold, enter into a lockdown state in which all user data access requests are denied until the device is recovered by an authorised administrator. The TOE implements a means to protect its security state from unauthorised modification or disclosure.

68 Each of the security functions listed above is discussed in more detail below. The relationship of the security functions and the SFRs for the TOE is illustrated in Table 13.

Table 13 – Security functions and SFRs

Security function	Applicable SFRs
Isolation	FDP_ACC.1b Subset access control (Data access SFP) FDP_ACF.1b Security attribute based access control (Data access SFP) FMT_MSA.1b Management of security attributes(Data access SFP) FMT_MSA.3b Static attribute initialization (Data access SFP) FDP_IFC.1 Subset information flow control FDP_IFF.1 Simple security attributes FMT_MSA.1c Management of security attributes (HD Isolation SFP)

Security function	Applicable SFRs
	FMT_MSA.3c Static attribute initialization (HD Isolation SFP)
Protection of the TOE	FDP_RIP.1a Subset residual information protection (OS Data) FPT_FLS.1 Failure with preservation of secure state FPT_TST.1b TSF testing (Start-up) FPT_TST.1c TSF testing (Boot)
Management	FMT_MSA.2 Secure security attributes FMT_SMF.1 Specification of Management Functions FMT_SMR.1 Security roles FTP_ITC.1 Inter-TSF trusted channel
Cryptography	FCS_CKM.1a Cryptographic key generation (AES) FCS_CKM.1b Cryptographic key generation (HMAC) FCS_CKM.4 Cryptographic key destruction FCS_COP.1a Cryptographic operation (Password hashing) FCS_COP.1b Cryptographic operation (User data protection) FCS_COP.1c Cryptographic operation (OS data protection) FCS_COP.1d Cryptographic operation (HMAC) FDP_RIP.1b Subset residual information protection (Cryptographic keys) FPT_TST.1a TSF testing (Key generation)
IAA	FDP_ACC.1a Subset access control (Lockdown SFP) FDP_ACF.1a Security attribute based access control (Lockdown SFP) FIA_AFL.1 Authentication failure handling FIA_ATD.1 User attribute definition FIA_UAU.1 Timing of authentication FIA_UID.1 Timing of identification FMT_MSA.1a Management of security attributes (Lockdown SFP) FMT_MSA.3a Static attribute initialization (Lockdown SFP)

8.2.1 Isolation

70 Isolation provides the capability for the TOE to restrict access to user data only to legitimate parties. This means that the user data is only decrypted if the end user of the TOE is successfully authenticated and never released to the disk drive of the Host PC. In addition, communications with the device is only allowed to those IP addresses explicitly approved by the administrator. The access restrictions implemented in the TOE therefore cover FDP_ACC.1b and FDP_ACF.1b.

71 The access control restrictions are enforced based on the security attributes defined in the above SFRs. The TOE also implements measures to ensure that only legitimate initial values are accepted for the security attributes so that the TOE is initialised and always boots up into a secure state. These restrictions implement measures to address the requirements coded in FMT_MSA.1b and FMT_MSA.3b.

72 The TOE also ensures that only legitimate information flows of user data occur, i.e. the user data can only be stored on the FLASH memory of the TOE and isolates the user data from the hard disk of the host PC. This feature enforces the information flow policies stated in FDP_IFC.1 and FDP_IFF.1. Additionally, the default values are restrictive meaning that a storage media must be explicitly identified as legitimate to allow user data to be stored at and that once the TOE is initialised, the configuration of the approved storage media cannot be changed. This covers FMT_MSA.1c and FMT_MSA.3c.

8.2.2 Protection of the TOE

73 The TOE implements a number of measures to protect itself from the integrity violations and to ensure that a well defined, secure state follows both from legitimate and expected TOE accesses as well as from anticipated failures.

74 The TOE implements measures to ensure that all sensitive data – including the authentication data and OS configuration files – is properly cleared immediately when it is no longer used. This is performed to ensure that an attacker gaining physical possession of the TOE may not deduce any sensitive data that could facilitate illegitimate access to the sensitive data. This covers FDP_RIP.1a.

75 The TOE also ensures that at each boot-up, a legitimate and authentic OS is booted. This prevents an insecure state resulting from a TOE modified when persistently stored on the USB token between the sessions. In case a failure occurs, such as the OS data is tampered with or has otherwise become corrupt, the TOE cancels the boot-up sequence to ensure that no unauthentic OS results. At the time of start-up booting of the TOE, measures are in place to check the integrity of the boot sequence and control the legitimacy of the TOE against illegal clones possibly produced by malicious parties between sessions. These measures jointly address FPT_FLS.1, FPT_TST.1b and FPT_TST.1c.

8.2.3 Management

76 In addition to the specific management functions of individual security functions, the TOE also implements generic measures for administration. Firstly, the TOE controls the values of manageable attributes to ensure that only secure values are accepted on security parameters and attributes. This covers FMT_MSA.2.

77 Second, the TOE can only be managed through well defined functions by users entering administrative roles. This covers FMT_SMF.1 and FMT_SMR.1. Finally, as coded in FTP_ITC.1, the TOE implements a means of establishing a trusted channel between itself and the administrative software outside the TOE for management of the TOE.

8.2.4 Cryptography

78 The TOE includes the Becrypt Cryptographic Library v 1.0, which is used for implementing the cryptographic functions of the TOE.

79 The library primitives are used for implementing key generation for AES and HMAC using SHA-256 (FCS_CKM.1a and FCS_CKM.1b) as well as for zeroizing the keys when they are to be destroyed (FCS_CKM.4).

80 The cryptographic functions include SHA-256 for password hashing (FCS_COP.1a), AES for user data protection (FCS_COP.1b) and for the protection of the OS files when stored on the TOE in between the sessions (FCS_COP.1c) as well as a HMAC using SHA-256 (FCS_COP.1d).

81 Additional assurance on cryptography is obtained through the measures the TOE implements to ensure that upon each generation of a key, the previous information content is properly wiped to ensure the non-correlation of the old and new cryptographic keys (FDP_RIP.1b) and that the self tests are conducted on the random number generator prior to the key generation to ensure the randomness of the resulting keys (FPT_TST.1a).

8.2.5 IAA

82 The identification, authentication and access control functionality provides assurance that access to sensitive data and resources is only granted to the parties whose authenticity is unambiguously established and who have an explicitly declared right to access the data (FIA_UID.1 and FIA_UAU.1).

83 This includes the identification and authentication routines to ensure that the user attributes used in the authentication are well defined, (FIA_ATD.1) and that the TOE maintains a counter of the number of consecutive, failed authentication attempts. If a defined number of unsuccessful attempts are detected, the TOE enters a lockdown state as a defence against password guessing or brute force attacks (FIA_AFL.1).

84 If a brute force or password guessing attack is detected, the TOE enters the lockdown state in which all the accesses are denied. The exact rules are explicitly coded in FDP_ACC.1a and FDP_ACF.1a. The TOE can be restored into an operational state only by the administrative software through the recovery procedure as coded inFMT_MSA.1a and FMT_MSA.3a.

9 Rationale

9.1 Conformance claim rationale

85 This ST does not claim conformance to any Protection Profile. Hence, there are no elements to be covered in the conformance claim rationale.

9.2 Security objectives rationale

86 Security objectives rationale is provided for the TOE and for the environment of the TOE.

9.2.1 Security objectives for the TOE

87 Table 14 provides a mapping of the TOE security objectives and threats and a justification for the mapping.

Table 14 – Mapping of TOE security objectives to threats

Threats	Objective	Justification
T.CRYPTO	O.CRYPTO	Cryptographic protection of the user data persistently stored on the TOE is a core security function of the TOE. The cryptographic primitives must be sufficiently secure and the TOE implementation must ensure that the underlying security of the cryptographic primitives is not reduced by the TOE design. Additionally, the implementation must be sufficient so that the software running on the Host PC could not successfully crypto analyse the TOE to deduce the user data without prior knowledge of the cryptographic key used for protecting that user data. Consequently, objective O.CRYPTO counters the threat T.CRYPTO. As O.CRYPTO is upheld, all cryptographic attacks against the user data and security parameters stored on the TOE exhaust the cryptoanalyst with an overwhelming probability. In practice, this means that even in the presence of an attacker the protected data on the TOE remains secure. **Application note**: At EAL2 it is sufficient to not assume a high attack potential and the ability of a threat agent to physically possess the TOE and subject it to advanced cryptanalysis. Instead, the threat scenarios of T.CRYPTO only cover software attacks by malicious software residing on the host PC while the TOE is being inserted into the USB slot.
T.AUTH_FAIL	O.AUTH	O.AUTH mitigates T.AUTH_FAIL by ensuring that passwords are of a sufficient complexity to make the effort required to guess the password unfeasible. Maximum failed logon attempts further increases the difficulty and time required to guess authentication data.

Threats	Objective	Justification
T.MANAGEMENT	O.MANAGEMENT	T.MANAGEMNT is mitigated by the TOE implementing mechanisms that maintain the integrity and confidentiality of AST.OS_FILES.
T.BOOT	O.BOOT O.AUTH	The TOE shall implement mechanisms that prevent the unauthorised modification of AST.BOOT
T.DATA_LEAK	O.ISOLATION	The TOE shall implement mechanisms that prevent AST.USER_DATA from being transferred to the host or non-trusted external network host.

9.2.2 Security objectives for the environment

88 Table 15 provides a mapping of the security objectives for the environment of the TOE to relevant threats and organisational security policies, as well as a justification for the mapping. There are no assumptions governing the usage and operation of the TOE, hence no assumptions are relevant to the mapping and justification.

Table 15 – Mapping of security objectives for the environment to threats, assumptions and OSPs.

Threat/OSP	Objective	Justification
A.APPL	OE.APPL	As the applications are outside the scope of the TOE, the TOE cannot enforce controls on their implementation. Neither does the TOE include a programmatic pre-assessment of the security of the applications and issuance of security statements as a means of asserting the trustworthiness of the applications. Consequently, the only way to ensure that the applications do not implement functions that might violate the security of the TOE is by TOE administrators to ensure security and trustworthiness of the applications for execution of the TOE and maintain an inventory of applications approved for execution on the TOE. Only applications explicitly declared trustworthy must at any time be installed on the TOE.

9.3 Security requirements rationale

9.3.1 SFR dependency rationale

89 Table 16 demonstrates the mutual supportiveness of the SFRs for the TOE by demonstrating how the SFR dependencies are fulfilled by the TOE, or justifying those dependencies not implemented.

90 The SARs relevant to the TOE constitute an evaluation assurance level EAL2 as defined in Common Criteria and include no extensions or augmentations. Therefore, as a complete evaluation assurance level, they are a mutually supportive set and require no further justification.

Table 16 – TOE SFR dependency demonstration

SFR	Dependency	Justification
FCS_CKM.1a	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1b and FCS_COP.1c by the TOE FCS_CKM.4 by the TOE The dependency to FCS_COP.1a is not applicable as FCS_COP.1a concerns with a cryptographically secure hash function SHA-256 which uses no keys.

SFR	Dependency	Justification
FCS_CKM.1b	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	FCS_COP.1d by the TOE FCS_CKM.4 by the TOE The dependency to FCS_COP.1a is not applicable as FCS_COP.1a concerns with a cryptographically secure hash function SHA-256 which uses no keys.
FCS_CKM.4	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]	FCS_CKM.1a and FCS_CKM.1b by the TOE
FCS_COP.1a	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	None of the dependencies is applicable as FCS_COP.1a concerns with a cryptographically secure hash function SHA-256 which uses no keys.
FCS_COP.1b	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1a by the TOE FCS_CKM.4 by the TOE.
FCS_COP.1c	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FDP_CKM.1a by the TOE FCS_CKM.4 by the TOE
FCS_COP.1d	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction	FCS_CKM.1b by the TOE FCS_CKM.4 by the TOE
FDP_ACC.1a	FDP_ACF.1 Security attribute based access control	FDP_ACF.1a by the TOE
FDP_ACC.1b	FDP_ACF.1 Security attribute based access control	FDP_ACF.1b by the TOE

SFR	Dependency	Justification
FDP_ACF.1a	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization	FDP_ACC.1a by the TOE FMT_MSA.3a by the TOE
FDP_ACF.1b	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization	FDP_ACC.1b by the TOE FMT_MSA.3b by the TOE
FDP_IFC.1	FDP_IFF.1 Simple security attributes	FDP_IFC.1 by the TOE
FDP_IFF.1	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialization	FDP_IFC.1 by the TOE FMT_MSA.3c by the TOE
FDP_RIP.1a	None.	None.
FDP_RIP.1b	None.	None.
FIA_AFL.1	FIA_UAU.1 Timing of authentication	FIA_UAU.1 by the TOE
FIA_ATD.1	None.	None.
FIA_UAU.1	FIA_UID.1 Timing of identification	FIA_UID.1 by the TOE
FIA_UID.1	None.	None.
FMT_MSA.1a	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1a by the TOE FMT_SMR.1 by the TOE FMT_SMF.1 by the TOE
FMT_MSA.1b	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1b by the TOE FMT_SMR.1 by the TOE FMT_SMF.1 by the TOE
FMT_MSA.1c	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_IFC.1 by the TOE FMT_SMR.1 by the TOE FMT_SMF.1 by the TOE
FMT_MSA.2	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FDP_ACC.1a, FDP_ACC.1b and FDP_IFC.1 by the TOE FMT_MSA.1a, FMT_MSA.1b and FMT_MSA.1c by the TOE FMT_SMR.1 by the TOE

SFR	Dependency	Justification
FMT_MSA.3a	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1a by the TOE FMT_SMR.1 by the TOE
FMT_MSA.3b	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1b by the TOE FMT_SMR.1 by the TOE
FMT_MSA.3c	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1c by the TOE FMT_SMR.1 by the TOE
FMT_SMR.1	FIA_UID.1 Timing of identification	FIA_UID.1 by the TOE
FMT_SMF.1	None.	None.
FPT_FLS.1	None.	None.
FPT.TST.1a	None.	None.
FPT_TST.1b	None.	None.
FPT_TST.1c	None.	None.
FTP_ITC.1	None.	None.

9.3.2 Tracing of SFR to security objectives

92 Table 17 provides the mapping of the TOE SFRs and the security objectives for the TOE.

Table 17 – Mapping TOE SFRs to objectives

Objective	SFRs	Demonstration
O.CRYPTO	FCS_CKM.1a FCS_CKM.1b FCS_CKM.4 FCS_COP.1a FCS_COP.1b FCS_COP.1c FCS_COP.1d FDP_RIP.1b FPT_TST.1a	Cryptographic protection of the TOE concerns with cryptographic protection of the sensitive data and all the functions concerned with the management of cryptographic keys. Key management concerns with the generation and destruction of the keys. Two types of keys are generated by the TOE: 256-bit AES keys and 64-bit integrity keys to be used with the HMAC construct based on SHA-256. AES keys are generated as stated in FCS_CKM.1a and integrity keys as stated in FCS_CKM.1b. All keys are destroyed as stated in FCS_CKM.4. Key generation is supported by the TOE ensuring that no residual information remains of the previously generated keys. Appropriate self tests are executed by the random number generator to ensure correct operation. This covers FDP_RIP.1b and FPT_TST.1a. The actual cryptographic operations concern with the hashing of passwords using a cryptographically secure hash function SHA-256 (FCS_COP.1a), encryption of sensitive user data and OS data when persistently stored on the TOE (FCS_COP.1b and FCS_COP.1c) and constructing integrity checksums using the HMAC constructed from SHA-256 (FCS_COP.1d). There are no additional cryptographic features implemented by the TOE, hence the SFR's fully address O.CRYPTO.
O.AUTH	FDP_ACC.1a FDP_ACF.1a FIA_AFL.1 FIA_ATD.1 FIA_UAU.1 FIA_UID.1 FMT_MSA.1a FMT_MSA.3a FPT_FLS.1	Preserving O.AUTH requires that the authentication rules are enforced so that only explicitly defined accesses to the TOE are available to the users prior to the identification or authentication of those users. This covers FIA_UAU.1 and FIA_UID.1. When the TOE loses power, there are no guarantees that the end user possessing the TOE is the legitimate end user but the TOE may have been lost or stolen during the power-off period. Therefore, whenever the TOE is powered up, the authentication state is cleared to ensure freshness of each session. This covers FPT_FLS.1. Additionally, the user attributes used for authentication must be defined (FIA_ATD.1) and in case of an authentication failure occurring, the necessary and explicitly stated action follows (FIA_AFL.1). The explicit action to take when an authentication failure occurs is to increase the counter keeping track of the number of consecutive, failed authentication attempts. If the counter value exceeds a threshold defined in the initialisation of the TOE, the TOE enters a lockdown state. This is a reaction to an assumed password guessing or brute force attack ongoing. In the lockdown state, no access is granted except TOE

Objective	SFRs	Demonstration
		Recovery requiring administrative access (FDP_ACC.1a, FDP_ACF.1a, FMT_MSA.1a and FMT_MSA.3a). Jointly, these SFRs address all aspects of O.AUTH and fully enforce O.AUTH.
O.MANAGEMENT	FMT_MSA.2 FMT_SMF.1 FMT_SMR.1 FTP_ITC.1	The management of the TOE concerns with ensuring that a secure channel between the TOE and the administrative software can be established. This covers FTP_ITC.1. Once the secure channel is established, the TOE ensures that all changes made in the security attribute values are such that the TOE remains in a secure state after those changes. This covers FMT_MSA.2. The TOE also maintains end user and administrative roles, and explicitly defines the administrative accesses available to the administrative users. This covers FMT_SMF.1 and FMT_SMR.1. Jointly, these SFRs cover the entire O.MANAGEMENT and fully enforce O.MANAGEMENT.
O.BOOT	FDP_RIP.1a FPT_TST.1b FPT_TST.1c FPT_FLS.1	A secure boot-up sequence (i.e. the one that results in an authentic TOE being booted up into a secure state) of the TOE requires implementation of a number of controls. Any residual data remaining from the previous session must be cleared to ensure that no inter-session data transfer may occur directly or indirectly. This is coded in FDP_RIP.1a. Sufficient self tests must be conducted to ensure that first the boot sequence is correct and then that the start-up of the TOE is correct. This is coded in FPT_TST.1b and FPT_TST.1.c. In case of a power failure occurring so that it interrupts a session, i.e. causes an uncontrolled termination of a session, the TOE must restore a secure state upon next power-on prior to the commencement of the new boot-up sequence. This covers FPT_FLS.1. Jointly, these SFRs ensure a secure boot-up sequence and a good integrity of the TOE once booted up, and fully enforces O.BOOT.
O.ISOLATION	FDP_ACC.1b FDP_ACF.1b FMT_MSA.1b FMT_MSA.3b FDP_IFF.1 FDP_IFC.1 FMT_MSA.1c FMT_MSA.3c	O.ISOLATION concerns with the prevention of data accesses that could expose user data to the hard disk of the host PC or allow communication of the user data to untrusted IP addresses. The access rules are coded in FDP_ACC.1b and FDP_ACF.1b. They also depend on the appropriate management of the security attributes on which the access control decisions are made (FMT_MSA.1b and FMT_MSA.3b). Additionally, there is the information flow control that complements the access controls by ensuring that only legitimate information flows may occur, i.e. user data can only be stored on an approved storage media. The rules for the flow controls are explicitly stated in FDP_IFF.1 and

Objective	SFRs	Demonstration
		FDP_IFC.1, and supported by FMT_MSA.1c and FMT_MSA.3c. Together, these access control rules and information flow controls fully enforce O.ISOLATION.

9.3.3 SAR justification

93 The set of SARs selected for the TOE constitute the entire evaluation assurance level EAL2 with no augmentations. As an EAL2 package, the set of SARs is an internally consistent and mutually supportive set of SARs.

94 When in use the TOE will be physically attached to an untrusted Host PC. When the TOE is not in use it will be in the physical possession of the end user. The relevant attack scenarios are logical attacks occurring through the external interfaces of the TOE by malicious software potentially residing in the Host PC.

95 The potentially malicious software running on a Host PC can only access the TOE through the USB interface. Attack scenarios concerning internal interfaces are not accessible as access to those interfaces would require physical probing of the TOE.

96 Attack scenarios concerned with physically probing the TOE with expert skill and resources are not relevant. Tamper evidence of the TOE casing is designed to provide basic level of assurance against undetected attempts to physically tamper with the TOE and to draw end user attention to the possible lack of integrity of the TOE.

97 Consequently, it is sufficient for the TOE to be engineered to demonstrate sufficient assurance against logical attacks by malicious software through externally visible interfaces as demonstrated by EAL2. As the EAL2 selected for the TOE provides a consistent baseline of assurance, the developer determines it is sufficient for the TOE and this evaluation.