blancco

Blancco Drive Eraser v6.9.1

Security Target v6.0

1 Document introduction

This document is the Common Criteria Security Target for Blancco Drive Eraser v6.9.1. It defines all the elements of a Common Criteria Security Target as defined in Common Criteria Version 3.1 Revision 5 Part 1, Part 2 and Part 3.

1.1 Revision history

Version	Date	Notes
6.0	22.05.2020	Final certification version

1.2 Abbreviations

BIOS	Basic Input/Output System
BMC	Blancco Management Console
CD	Compact Disk
CLI	Command Line Interface
CMOS	Complementary Metal Oxide Semiconductor
cPP	Collaborative Protection Profile
CPU	Central Processing Unit
DECT	Drive Eraser Configuration Tool
DHCP	Dynamic Host Configuration Protocol
EAL	Evaluation Assurance Level
EAL*n*	Evaluation Assurance Level n, $n \in \{1,2, \dots ,7\}$
eMMC	embedded Multi Media Card
FC	Fiber Channel
FW	Firmware
GB	Giga Byte
GUI	Graphical User Interface
HASP	Hardware Against Software Privacy
HDD	High Disk Drive
HTTPS	Hypertext Transfer Protocol Secure
HW	Hardware
ISO	ISO image formatted as per the ISO 9660 file system

NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NVM	Non-Volatile Memory
NVMe	NVM Express
OSP	Organisational Security Policy
PC	Personal Computer
PII	Personally Identifiable Information
PP	Protection Profile
PSU	Power Supply
PXE	Pre-Boot eXecution Environment
RAID	Redundant Array of Independent Disks
RAM	Random Access Memory
SAS	Serial Attached SCSI
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Interface
SFR	Security Functional Requirement
SSD	Solid State Drive
SSH	Secure Shell
SVGA	Super Video Graphics Array
SW	Software
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function
UCT	Universal Coordinated Time
USB	Universal Serial Port
VESA	Video Electronics Standards Association

1.3 References

[CC Part1] Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model. April 2017 Version 3.1 Revision 5 CCMB-2017-04-001.

[CC Part 2] Common Criteria for Information Technology Security Evaluation Part 2: Security functional components. April 2017 Version 3.1 Revision 5 CCMB-2017-04-002.

[CC Part 3] Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components. April 2017 Version 3.1 Revision 5 CCMB-2017-04-003.

1.4 Table of contents

2 Security Target Introduction

This section is the Security Target Introduction. ST Reference is given in Sect. 2.1. The TOE Reference is given in Sect. 2.2. The TOE Overview is given in Sect. 2.3. The TOE Description is given in Sect. 2.4.

2.1 ST reference

ST Title	Blancco Drive Eraser v6.9.1 Security Target
ST Version number	6.0
ST Date	22.05.2020

2.2 TOE reference

TOE Name	Blancco Drive Eraser
TOE Version	6.9.1

2.3 TOE Overview

This section provides the TOE Overview. An introduction to the TOE is given in Sect. 2.3.1. Usage and major security features of the TOE are given in Sect. 2.3.2. The TOE Type is stated in Sect. 2.3.3. Non-TOE hardware, software and firmware used by the TOE is identified in Sect. 2.3.4.

2.3.1 Introduction to Blancco Drive Eraser v6.9.1

Blancco Drive Eraser v6.9.1 (the TOE) is a software product for securely erasing entire hard disk drives or partitions thereof (collectively called drives) in accordance with recognized standards.

The TOE is delivered as an ISO file which can be stored on the media of choice and used for booting a PC to a state where the TOE is running in RAM and the drive(s) attached to the Host PC can be securely erased. The TOE can erase traditional Hard Disk Drives (HDD), Solid State Drives (SSD) and NVM Express (NVMe) drives. The TOE also erases hidden partitions and other 'hard to clean' parts of the drives.

Standards for secure erasure of drives can be based on overwriting, cryptography or voltage manipulation. HDDs can be erased using overwriting and cryptographic erasure methods. SSDs and NVMe drives can be erased using overwriting, cryptographic and voltage based erasure methods. SSDs typically include control software which prevents overwriting and cryptographic erasure methods from being efficient, but the TOE is capable of secure erasing SSD and NVMe disks using these methods even in the presence of the control software.

All disks contain firmware which is required to implement relevant erasure primitives in accordance with the standard the disk conforms with. Secure erasure of the content typically requires these primitives to be applied in a sequence in accordance with the selected erasure standard and the type of the disk. The TOE implements the erasure standards using the firmware primitives of the drive selected for erasure. Given that the primitives cannot be trusted to function precisely as required by the standard, the TOE verifies the erasure

outcome after each erasure round before reporting the success or failure of the erasure to the user.

Once stored on the media, the .iso image of the TOE can be configured using the Blancco Drive Eraser Configuration Tool (DECT). DECT is an external software tool and is not part of the TOE. This configuration takes place prior to the TOE being booted to the RAM of the Host PC and includes setting of the key parameters of the TOE. Some of those parameters may be changed by the user once the TOE becomes operational, but some cannot.

The TOE can be configured to behave in a certain manner via files embedded in the image (config.xml and preferences.xml) while the rest of the image consists of identical binaries. Such configurations define the type of licensing that the TOE will use (a.k.a. the TOE edition), the source of the licensing (a.k.a. the TOE license container) as well as other user preferences (e.g. default erasure standard to use, etc.). The TOE license container stores the licenses used for verifying whether a user has a right to access an operation. Some configurations may also be associated to a Blancco Management Console (BMC) which is a tool for operating the TOE and is not part of the TOE.

The different license containers of the TOE are summarised in Table 1. The different editions of the TOE are summarised in Table 2.

Blancco Drive Eraser solutions are also available as hardware appliances but that hardware appliance models are not included in the certification.

Table 1 TOE License Containers

Container	Characteristics
BMC	The TOE is used in associate with a Blancco Management Console (BMC) which holds the licenses of the user.
HASP	The TOE is used in association with a secure dongle (HASP) which holds the licenses of the user. The HASP may also be associated with a BMC instead of the TOE.
BIOS	The TOE is used without a BMC or HASP and access to TOE functions is granted on the validity of the TOE as verified against the BIOS clock, not using the licenses.

A given TOE edition will use a certain license type that is identified by a license number and different license numbers allow access to different functions. The numbering is only used for identification purposes, license numbers are not hierarchical but there are two types of licenses: asset licenses and erasure licenses. Asset licenses are required for all operations on the TOE and additional erasure licenses are required for performing erasure and reporting operations. Unless in the BIOS configuration, licenses are verified before any operation.

Table 2 TOE Editions

Edition	Characteristics
Server	A solution for erasing servers and storage systems with RAID (HDDs and/or SSDs). Blancco Drive Eraser (Server) licenses are required for erasing the connected HDDs/SSDs
PC	A solution for erasing (non-RAID) desktop and laptop computers that contain HDDs and/or SSDs. Blancco Drive Eraser (PC) licenses are required for erasing the connected HDDs/SSDs.

Enterprise	A solution for erasing servers and storage systems with RAID (HDDs and/or SSDs). Blancco Drive Eraser (Enterprise) licenses are required for erasing the connected HDDs/SSDs. The Enterprise edition is functionally identical to the Server edition but has different licensing options.

Throughout the operational life-cycle, the TOE shall go through a number of stages. The life-cycle stages of the TOE are summarised in Table 3.

Table 3 TOE Operational Life-Cycle stages

Stage	Description
Receipt	The user receives from Blancco a link to a .iso file, a SHA-256 checksum of the .iso file, TOE security guidance, and instructions on how to verify the authenticity of the .iso file.
Download	The user downloads the .iso file and stores it on a trusted, local computer. The authenticity of the .iso file is verified using the SHA-256 checksum. After successful downloading and verification, the .iso file is handled in accordance with the security practices of the user.
Configure	The user uses DECT to configure the .iso file (i.e. modifying the included preferences.xml file) to ensure conformance of the TOE with the user's security policies and practices.
Store on Media	Upon completion of the configuration of the .iso file, the file is stored on the bootable media of choice. If stored on a USB memory stick, a Blancco USB Creator tool must be used.
Boot to RAM and Operate	The Host PC is booted from the media to which the .iso file was stored. The TOE boots into the Host PC RAM bypassing the local operating system and makes available to the user the functions of the TOE given the availability of required licenses. The TOE may be used for erasing the drive of the Host PC or any other drive connected to the Host PC. Depending on the configuration the TOE may communicate with a BMC or with the HASP.
Terminate	Upon shutting down the Host PC, the TOE is erased from the RAM in which it executes, and the TOE becomes unoperational until the .iso file is used for booting up the same or a different Host PC. The .iso file may be reconfigured using DECT.

2.3.2 Usage and major security features of the TOE

The TOE software exists in two different forms: as a .iso file stored on a local media (whether bootable or not) and as an executable software in the RAM of the Host PC.

When stored as a .iso file in a host, the TOE is configured using a Blancco Drive Eraser Configuration Tool (DECT). There are no executable TOE functions at that stage and DECT writes the configuration changes to specific files within the .iso file. Specifically, this is the preferences.xml file. The two configuration files (config.xml and preferences.xml) are encrypted using 128-bit AES keys prior to being stored on the .iso file. The executable files of the .iso file contain the same keys so that the configuration files can be decrypted by the TOE when being executed.

Given the absence of executable TOE functions, the TOE cannot perform access control functions and the configuration of the .iso file must occur in a secure environment in accordance with the policies and practices governing the secure use of the TOE.

When the Host PC is booted from the media containing the .iso file, the TOE is executed on the RAM of the Host PC. The TOE performs self-tests to verify the authenticity and integrity of the executables at the boot time. If the self-tests pass, the functions of the TOE become available to the user and the user may access them either locally through the GUI of the TOE or remotely using the Blancco Management Console (BMC).

If the TOE is configured to be used with a BMC, the TOE establishes a HTTPS connection between itself and the BMC to ensure secure communication.

The TOE controls access to the functions using the software licenses and prevents the execution of the functions to which the user does not possess appropriate license. There is no identity-based user identification or authentication and the user of the TOE is identified using the licenses in the possession of the user.

If the TOE uses the BMC as a license container, then valid credentials (user authentication via <username, password> pair) must be provided to access the BMC and consume the licenses allowing the execution of the TOE functions. The credentials are stored in the TOE configuration files but the authentication is solely by the BMC, not by the TOE and is therefore not included in the scope of the TOE.

In most use cases the licenses are stored by the BMC or a HASP on behalf of the user but there exists also a use case where neither of the two is required and the TOE checks the authorisation based on the BIOS time and the validity period of the software. In this case the user is required to ensure that the BIOS time is accurate as the erasure report shall also include the time read from the BIOS clock and any report may appear obsolete if the time of erasure is inaccurate.

The essential security function of the TOE is to securely erase drives connected to the Host PC. Using the firmware primitives of each disk, the TOE implements a number of secure erasure algorithms and executes the one selected by the user on the selected drive. The TOE ensures that the drive data is completely erased and verifies the erasure result after each step in the erasure algorithm. The TOE also provides a basic set of tools to testing the hardware of the Host PC to assist in any diagnostics and in ensuring that the Host PC functions correctly.

The user is given a notification of the status of erasure and the TOE also generates a report of the erasure details. The report is digitally signed using a 2048-bit RSA private key to ensure that the recipient of the report can be assured of the authenticity of the report. The signing key is generated during the production of the TOE and cannot be changed or removed by the user. Only a legitimate BMC is capable of verifying the signature of the report.

BIOS configuration where the authorisation to use the TOE is based on BIOS time instead of the licenses is an exception to the above. In this case, the TOE generates an ephemeral 128-bit AES key which is used for encrypting some content of the report. The report with parts encrypted is signed using the report signing key as above. The TOE contains a 4096-bit RSA key (also generated at the production of the TOE) which is used for encrypting the ephemeral key. The encrypted ephemeral key together with the signed, partially encrypted report is sent to the BMC and can be, given access to the private key corresponding to the public key which was used for encrypting the ephemeral key, decrypted and viewed.

In addition to the key for digitally signing the reports, the TOE also contains a 256-bit symmetric AES key which the TOE uses, when applicable, for establishing a HTTPS

connection between itself and the BMC. The TOE authenticates to the BMC using a <username, password> pair which is stored in the configuration of the .iso file and made available for authenticating to the BMC when the TOE boots to the RAM of the Host PC.

User management is outside the scope of the TOE. All configuration of the TOE is done using DECT and once the TOE becomes operational and is executed in the Host PC, there are no user authentication and role assignment functions. Instead, all TOE functions are available all users and are not separated into management and operational functions.

All cryptographic keys except the ephemeral key used in the BIOS configuration are static and cannot be changed or destroyed. They are generated at the TOE production time by Blancco.

2.3.3 TOE Type

The ST does not claim conformance to any PP. Therefore, the ST does not claim TOE Type defined in any Protection Profile. Instead, the TOE Type is drive erasure software executing on a Host PC.

2.3.4 Non-TOE HW, SW and FW used by the TOE

The TOE is the secure erasure software consisting of all the necessary executable and other files delivered as an ISO image from which it can be installed on the media of choice. As such, the TOE needs the HW, SW and FW listed in Table 4 to function. All items are mandatory unless explicitly stated as optional.

Table 4 Non-TOE HW, FW and SW required by the TOE

Storage Environment	The TOE requires a storage environment on which the .iso file is stored upon receipt from Blancco. The authenticity of the received .iso file is checked in the storage environment and the configuration of the .iso file using DECT carried out. If appropriate in accordance with the security policies and practices of the user, Storage Environment may be the same host as the Execution Environment.
Execution Environment	The TOE requires an execution environment (the Host PC) on which the TOE software is executed. The TOE software may be executed on an x86 architecture PC. In the minimum the host PC must meet the following: – 1 GB of RAM in most cases. Erasing servers with 2+ drives requires more RAM. – PXE-booting requires 2 GB of RAM. – CD-drive or a CD-compatible drive for CD-booting. – USB-port for exporting / saving reports locally and/or USB-booting. – SVGA display and VESA compatible video card for graphical user interface. – (Optionally) Ethernet NIC, DHCP Server running on local network.

	– If the client software is running on a desktop, a sufficient PSU for all connected drives and hardware.
Baseline security software	Storage Environment and Execution Environment must be equipped with the necessary security software to ensure operation in accordance with the security governance policies and practices of the organization using the TOE.
Bootable Media	Blancco Drive Eraser may be booted from any bootable media to which the .iso file is written from the Storage Environment.
USB Flash drive and Blancco USB Creator Tool (optional)	The Bootable media may be a USB flash drive. In that case, the drive must be created with the Blancco USB Creator tool. If the user chooses the bootable USB flash drive option, then both the flash drive itself and the Blancco USB Creator tool are required.
Checksum verification software	The TOE is downloaded by the user and the user is provided with a cryptographic checksum of the ISO image of the TOE. The recipient must use appropriate verification software to verify the checksum.
ISO Image mounting tool	The TOE is delivered as an ISO image. If mounting the ISO image locally, either the operating system of the host PC must include a mounting software for ISO images, or an external ISO mounting software must be installed. There are several ways the mounting can be done depending on the specific environment in which the TOE is used but knowledge of how to perform this can be reasonably expected from the operators of the TOE. Common mechanisms can be used for mounting the ISO image include – Burning the ISO on a CD/DVD (can be made via Windows OS); – Making a bootable USB stick (needs the Blancco USB Configuration tool); – Copying/extracting the ISO on a special directory for PXE booting (needs a working PXE environment); – Loading and running the ISO on a virtual machine; and – Adding the ISO to an MSI package (special case for some Enterprises customers, this is usually made by Blancco personnel).
Drive to be erased	Blancco Drive Eraser is a tool for securely and completely erasing drives which are not part of the TOE. As such, the TOE must be connected to the drive that is to be erased. Drive Eraser can erase any connected drive (SATA, SCSI/SAS, FC, USB, eMMC, NVMe) as well as removable flash-based devices. Solid State Disks (SSD) and hybrid disks may be erased under caveats defined in the Blancco documentation.

DECT	Configuration of the TOE in the Storage Environment occurs through a dedicated tool, Blancco Drive Eraser Configuration Tool (DECT) which is not part of the TOE.
BMC (Optional)	Optionally, the TOE may be configured for operation from a Blancco Management Console (BMC) which is not part of the TOE. BMC must be version 4.8.0 or newer. If the TOE is configured for remote erasure, that remote erasure must be performed via BMC. The BMC may be standalone or cloud based.
HASP (Optional)	Some configurations of the TOE require a HASP which is used for storing the licenses. If a HASP is used, Blancco shall source a HASP and initialise it. After initialisation, it shall be delivered to the customer by courier and the licenses to be uploaded to the HASP shall be sent by email with the instructions on how to upload them to the HASP. A BMC may also use a HASP for storing the licenses associated to a user.

2.4 TOE description

This section provides the TOE Description. Physical scope of the TOE is described in Sect. 2.4.1 and the logical scope of the TOE in Sect. 2.4.2.

2.4.1 Physical Scope of the TOE

Physical scope of the TOE consists of the TOE software and Security Guidance for the TOE. The licenses are not part of the TOE but are used for controlling access to the TOE functions. The TOE is delivered protected by a cryptographic checksum, but that checksum is not part of the TOE.

The software constituting the TOE occurs in two different representations: In an ISO image (i.e. an .iso file, an image in accordance with ISO 9660) in which the TOE is delivered to the clients, and in an executable software as booted from the .iso file for execution at the RAM of the Host PC. These are both representations of the same TOE. Once the .iso file is stored on a bootable media and a Host PC is booted from that media, the executable software of the TOE runs on the Host PC.

The ISO image file names are the following:

- PC edition: `Drive_Eraser_691.iso`
- Server edition: `Drive_Eraser_server_691.iso`
- Enterprise edition: `Drive_Eraser_EE_691.iso`

The ISO image representation of the TOE is downloaded by the user from a secure web page to which the user is emailed a link. The user is also provided with a cryptographic checksum which can be used for verifying the integrity and authenticity of the ISO image. Once the ISO image has been successfully verified for authenticity, it must be stored and configured in accordance with the security practices of the organisation using the TOE.

Security Guidance of the TOE is included in the physical scope of the TOE and is the following:

`Blancco Drive Eraser v6.9.1 Common Criteria Guidance Supplement v4.0`

2.4.2 Logical Scope of the TOE

The logical scope of the TOE includes the following security functions:

Legitimate use The TOE implements measures to ensure that the TOE executables and the configuration files are not tampered with and any tampering is detected at the start up and the TOE always boots up to an authentic state. This together with basic diagnostics tools for the underlying hardware ensures that the TOE executables are authentic at each time the TOE is booted up and the configuration files are legitimate. Any tampering is detected with an overwhelming probability. If the program execution is terminated abruptly, the execution may continue from the state stored in the resume file once the TOE is booted up again assuming 1) the “Erasure Resume” functionality has been enabled via DECT and 2) a USB stick is plugged to the Host PC for storing the information to resume in a non-volatile manner. Furthermore, the TOE implements access control mechanisms to ensure that only legitimate accesses to the TOE functions are granted.

Protected communications. The TOE ensures that the reports generated are accurate and precisely reflect the outcome of the erasure. The report can be stored locally on a USB token attached to a Host PC or it can be sent to the BMC. In each case, the report is digitally signed to ensure that any attempt to modify it shall be detected with a high likelihood. The TOE also reports to the user the progress of erasure on the display. Each step in erasure is verified prior to reporting to the user to ensure that reliability of the communication so that only reliable erasure status is communicated to the user. The TOE also establishes a HTTPS channel between itself and the BMC (when BMC is used) to ensure that the communication between itself and a BMC cannot be interfered with.

Complete Erasure. The TOE implements a set of secure erasure functions for the drives connected to the Host PC. Using the firmware primitives of each disk, the TOE implements the erasure protocol logic to ensure secure erasure of the drive in full conformance with the applicable standards. To ensure complete erasure, the TOE also verifies the erasure results after each step and prior to completing the erasure.

The Blancco Drive Eraser v6.9.1 implements a large number of erasure algorithms but not all of them are included in the logical scope of the TOE. Inclusion in the logical scope of each algorithm implemented by the Blancco Drive Eraser v6.9.1 is given in Table 5.

Table 5 Erasure algorithms included in the logical scope of the TOE

Erasure standard	Rounds	Included
Air Force System Security Instruction 5020	4	No
Aperiodic random overwrite	1	No
Blancco SSD Erasure	**2+**	**Yes**
Bruce Schneier's Algorithm	7	No
BSI-2011-VS	1-2	No
BSI-GS	1-2	No
BSI-GSE	2-3	No
CESG CPA – Higher Level	**3**	**Yes**

Cryptographic Erasure	0	No
DoD 5220.22-M	**3**	**Yes**
DoD 5220.22-M ECE	**7**	**Yes**
NIST 800-88 Clear	**0-1**	**Yes**
NIST 800-88 Purge	**0**	**Yes**
Firmware Based Erasure	0	No
Extended Firmware Based Erasure	1	No
HMG Infosec Standard 5, Higher Standard	**3**	**Yes**
HMG Infosec Standard 5, Lower Standard	**1**	**Yes**
National Computer Security Center (NCSC-TG-025)	4	No
Navy Staff Office Publication (NAVSO P-5239-26)	3	No
NSA 130-1	3	No
OPNAVINST 5239.1A	3	No
Peter Gutmann's Algorithm	35	No
U.S. Army AR380-19	3	No
RCMP TSSIT OPS-II	8	No
Random byte overwrite (3x)	3	No
OPAL Cryptographic Erasure	1	No

3 Conformance Claims

3.1 Conformance Claims statement

The ST and TOE claim conformance to Common Criteria v3.1 Release 5 Part 1, Common Criteria v3.1, Release 5 Part 2, and Common Criteria v3.1 Release 5 Part 3.

Common Criteria v3.1 Release 5 Part 1 is fully identified in [CC Part 1], Common Criteria v3.1 Release 5 Part 2 in [CC Part 2] and Common Criteria v3.1 Release 5 Part 3 in [CC Part 3].

The ST is CC Part 2 conformant.

The ST is CC Part 3 conformant.

The ST claims conformance to the following Protection Profiles and Packages: **None**.

The ST claims package conformance to the following: **Evaluation Assurance Level EAL2**.

3.2 Conformance Claims Rationale

The ST does not claim conformance to any Protection Profile. Therefore, the Conformance Claims Rationale is not applicable.

4 Security Problem Definition

This section describes the security problem definition of the ST. The security problem definition is described in terms of threats, organizational security policies and assumptions. Each element is identified by a prefix and a short name, followed by a definition of the element.

Threats are identified using prefix T., organizational security policies are identified using prefix OSP. And assumptions are identified using prefix A.

4.1 Threats

The TOE concerns with the following threats:

T.NETWORK_ATTACK An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may engage in communications with the application software or alter communications between the application software and other endpoints in order to compromise it.

T.INCOMPLETE_ERASURE An attacker succeeds in manipulating the TOE through a legitimate interface to modify the behaviour of the TOE in a manner that leaves the erasure of a drive incomplete in a manner not detected by the operator of the TOE.

4.2 Organizational Security Policies

The following Organizational Security Policies govern the TOE:

OSP.SANITISATION The organization utilizing the TOE has defined and enforced a media sanitization policy which covers in the minimum 1) the timing when sanitization must occur, 2) allowed and disallowed erasure methods, 3) handling the sanitization of all volatile, non-volatile and EPROM/EEPROM memories, 4) handling of sanitization of classified data and the classifications before and after erasure, 5) sanitization of media when using encryption and decryption software, and 6) actions taken by Operators of the TOE in case of a sanitization failure to guide the administrators and operators in ensuring that only the erasure algorithms sufficient to meet the security requirements of the organization are used by the TOE.

OSP.NIST800-88_CLEAR An organisation using the NIST 800-88 Purge algorithm of the TOE must have defined a policy on whether a fallback into NIST 800-88 Clear is allowed or disallowed in case of a failure of NIST 800-88 Purge algorithm.

OSP.REMOTE_LOCAL An organization using the TOE has defined the allowable use cases for the TOE concerning local and remote erasure of the drives to determine whether both local and remote erasure are allowed, or only one of the TOE.

OSP.LOCAL_CLEAN An organization using the TOE has defined a security policy for the host in which the TOE is used. This policy must define the minimum security countermeasures required to be in place to reduce the likelihood of malicious software in the local host, including the firmware of the drive to be erased, preventing the TOE from successfully erasing the drive intended.

OSP.RAID Server and Enterprise editions of the TOE are capable of securely erasing RAID disks. Nevertheless, if the RAID disks remain switched on after the completion of an erasure, it is possible that the RAID control software restores some of the information of the disks from associated remote disks. The organisation using the TOE must ensure that their policies for handling erasure of RAID disks take this possibility into account and, if deemed unacceptable, define the measures required for removing the eventuality.

OSP.PDF The erasure reports generated by the TOE are digitally signed for authenticity. Each signature is computed for a .xml version of a report which is intended for exporting to a BMC. The report may also be stored locally on a USB token. If not intended for exporting to the BMC, the report may be stored in a .pdf format. However, the signature appearing on the .pdf report is that computed from the corresponding document in.xml format of the report, not from the .pdf of the same report. The organisation using the TOE must perform a risk assessment and determine whether saving the reports in .pdf is allowed or not and to ensure that the users of the TOE are aware of this policy.

4.3 Assumptions

The following assumptions govern the TOE:

A.PLATFORM The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying platform and whatever runtime environment it provides to the TOE.

A.PROPER_USE The user of the application software is not wilfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.

A.AUTHENTIC_DECT The TOE is only administered using a legitimate and authentic Drive Eraser Configuration Tool (DECT). If the administrator of the TOE suspect the DECT is not authentic, it shall not be used for administering the TOE until the authenticity has been established.

A.AUTHENTIC_BMC If the TOE is used for remote erasure of the drives, the remote erasure only takes place using a legitimate and authentic Blancco Management Console. If the operator of the TOE suspects that the BMC is not authentic, it shall not be used for remote erasure of drives until the authenticity has been established.

5 Security Objectives

This section describes the security objectives for the TOE and the security objectives for the environment of the TOE. Each security objective is given an identified consisting of a prefix and a short name. Security objectives for the TOE are identified by prefix O. and the security objectives for the environment are identified by prefix OE.

5.1 Security objectives for the TOE

The following security objectives are for the TOE:

O.PROTECTED_COMMS To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application.

O.COMPLETE_ERASURE The TOE ensures that upon successful completion of the erasure, the drive is fully erased in accordance with the selected erasure standard. The indicator of the completeness of the erasure can be relied upon and in any case of incomplete erasure, the indicator shall not indicate successful erasure.

O.LEGIT_USE The TOE ensures that it is only used in legitimate manners: either locally from the Graphical User Interface (GUI) or remotely using the Blancco Management Console (BMC).

5.2 Security objectives for the environment of the TOE

The following security objectives are applicable for the operational environment of the TOE:

OE.PLATFORM The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.

OE.PROPER_USE The user of the application software is not wilfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.

OE.PROPER_SETUP The administrator of the TOE only configure to a setup which is valid and in accordance with the policies of the organization using the TOE.

5.3 Security objectives rationale

5.3.1 Tracing of security objectives

The tracing of the security objectives to the threats, organizational security policies and assumptions is given in Table 6.

Table 6 Tracing of the security objectives to the threats, organizational security policies and assumptions

	T.NETWORK_ATTACK	T.INCOMPLETE_ERASURE	A.PLATFORM	A.PROPER_USE	A.AUTHENTIC_DECT	A.AUTHENTIC_BMC	OSP.SANITISATION	OSP.NIST800-88_CLEAR	OSP.REMOTE_LOCAL	OSP.LOCAL_CLEAN	OSP.RAID	OSP.PDF
O.PROTECTED_COMMS	X								X			
O.COMPLETE_ERASURE		X					X	X				
O.LEGIT_USE		X							X	X		
OE.PLATFORM			X							X		
OE.PROPER_USE				X					X		X	X
OE.PROPER_SETUP					X	X						

5.3.2 Justification of the tracing

O.PROTECTED_COMMS concerns with the use of trusted channels for communicating with external entities. This concerns with two external entities: The Drive Eraser Configuration Tool (DECT) which the administrator uses for configuring the TOE and the Blancco Management Console (BMC) that the operator may in addition to a local access via the GUI use for operating the TOE. Communication with DECT occurs prior to the TOE becoming operational and is therefore addressed by policy. Communication with external entities is protected when 1) all communication between the TOE and the BMC is protected, and 2) remote erasure of drives using the BMC is only allowed if the level of risk is deemed acceptable to the organization using the TOE. Concern (1) is ensured if T.NETWORK_ATTACK is prevented from occurring, concern (2) is prevented from occurring if OSP.REMOTE_LOCAL is defined and enforced by the organization.

O.COMPLETE_ERASURE concerns with ensuring that upon completion of a drive erasure either the drive is fully erased in accordance with the selected erasure standard or the operator of the TOE is provided with a reliable indication that the erasure was unsuccessful. The TOE implements a number of different erasure standards and the erasure is only sufficient if 1) the erasure standard selected fulfils the secure erasure objectives of the organisation using the TOE, 2) the erasure is completed in accordance with the selected erasure standard, and 3) the erasure is complete or the operator is given an unambiguous notification of the failure. Concern (1) is addressed if the organization using the TOE ensures the sufficient policies for erasure are defined i.e. OSPs OSP.SANITISATION and OSP.NIST800-88_CLEAR are defined and in place, and concern (2) is addressed if threat T.INCOMPLETE_ERASURE is prevented from occurring.

Note: It is plausible in the scenario where BMC is used for remote erasure that the TOE erasure of a drive is incomplete and the TOE sufficiently indicates that the erasure failed but a network attack modifying the contents of the notification succeeds in falsifying the notification and leading the operator of the TOE into believing that the erasure was successful. This is not a scenario addressed by O.COMPLETE_ERASURE but is deemed a communication security problem and is prevented from occurring in practice if O.PROTECTED_COMMS is fully enforced by the TOE.

O.LEGIT_USE concerns with ensuring that the TOE is only used in a legitimate manners: locally using the GUI or remotely using the BMC, and that sufficient countermeasure ensure that the behaviour of the TOE cannot be falsified by malicious agents. To ensure this, the TOE implements technical countermeasures for protecting itself from interference but also requires the organization using the TOE to ensure that the policies governing the use of the TOE and the level of acceptable risk when using the TOE are considered.

The countermeasures preventing falsification of the behavior of the TOE concern with dependable reporting of the success or failure of an erasure. For the TOE to enforce O.LEGIT_USE, the TOE must ensure that each erasure of a drive either is complete or the operator of the TOE is given a dependable notification of the failure of erasure. This is ensured by the TOE if threat T.INCOMPLETE_ERASURE is prevented from occurring.

The policy concerns require that the organization utilizing the TOE has considered the risk of using the TOE remotely and locally and defined a policy to allow or disallow remote erasure using the TOE. This is addressed if OSP.REMOTE_LOCAL is defined and enforced by the organization using the TOE. Additionally, the TOE must not be used to erasure the drives of any host but the organization using the TOE must define the minimum level of countermeasures required in the hosts whose drives are to be erased are sufficiently ‘clean’ to ensure that the risk of malicious software residing in the host and attacking the TOE is acceptable. This is addressed if OSP.LOCAL_CLEAN is defined and enforced by the organization using the TOE.

OE.PLATFORM concerns with acknowledging that the TOE is application level software which requires on the underlaying platform for execution and that there is a risk that the underlying platform is also running software attempting to prevent the TOE from achieving its security objectives. To fulfil this objective for the environment for the TOE, the organization using the TOE must 1) acknowledge the possibility of malicious processes being executed in the underlying platform and 2) assess the risk of malicious processes and define the minimum criteria for the trustworthiness of the platforms on which only the TOE may be used. Concern (1) is addressed by assumption A.PLATFORM and concern (2) is addressed by the organisation using the TOE defining and enforcing policy OSP.LOCAL_CLEAN.

OE.PROPER_USE concerns with the trustworthiness of the operators of the TOE. It is not possible for the TOE to enforce by technical means that only sound operation of the TOE is carried out. Therefore, it must be assumed that the operator of the TOE is not malicious and does not intentionally attempt to abuse of misuse the TOE, and at all times follows the guidance of the TOE. This is addressed by assumption A.PROPER_USE in the operational environment of the TOE. The organisation using the TOE must also define and enforce a policy on whether reports in .pdf are allowed when the reports are stored locally (OSP.PDF), how to properly handle RAID disks

(OSP.RAID) and whether the TOE can only be used locally or also remotely using a BMC (OSP.REMOTE_LOCAL).

OE.PROPER_SETUP concerns with ensuring that the organization using the TOE only uses the TOE with a legitimate Drive Eraser Configuration Tool (DECT) and, when allowed by the organization, legitimate Blancco Management Console (BMC) and that both tools are ensured to be authentic Blancco products. This is enforced by assumptions A.AUTHENTIC_DECT and A.AUTHENTIC_BMC in the operational environment of the TOE.

6 Extended component definition

This ST defines no extended components applicable to the TOE. Therefore, this section is not applicable and is omitted.

7 Statement of security requirements

This section defines the security requirements for the TOE. The security functional requirements are defined with reference to CC Part 2 and to Sect. 6. The security assurance requirements are defined with reference to a well-defined evaluation assurance package EAL2 defined in CC Part 3. The ST claims no extensions or augmentations to the package EAL2.

The statement of security functional requirements utilizes operations as defined for each applicable security functional requirement in CC Part 2 and Sect. 6. The notation for identifying the operations is as follows:

Iteration is identified by repeating the identifier of the security functional requirement with a string indicating a specific iteration separated from the SFR identification by a slash (e.g. FCS_COP.1/AES, FCS_COP.1/DSIG).

Refinement is identified by a) indicating in square brackets in bold font any added text, in form of [**Refinement: added text**] and b) indicating any removed words using ~~overstrike~~ font. Whenever a refinement is used, the rationale and justification of the refinement is given immediately after the statement of the security requirement.

Selection is identified by indicating the selected values in [**square brackets using bold font**].

Assignment is identified by indicating the assigned values in [***square brackets using bold, italic font***].

Application notes may be added after the formal statement of the security requirements to assist the reader in understanding the specific security requirement in the context of this particular TOE.

7.1 Statement of Security Functional Requirements

7.1.1 Class FCS: Cryptographic Support

7.1.1.1 FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [***random number generation***] and specified cryptographic key sizes [***128 bits***] that meet the following: [***none***].

Application note: The TOE only generates the ephemeral 128-bit AES key used for encrypting the erasure report in BIOS mode. All other keys are generated by Blancco during the production of the TOE and stored in the .iso file.

7.1.1.2 FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***overwriting with zeros***] that meets the following: [***none***].

Application note: The TOE only destroys the ephemeral 128-bit AES key used for encrypting the erasure report in the BIOS mode. All other keys are static and cannot be replaced or destroyed.

7.1.1.3 FCS_COP.1 Cryptographic operation

FCS_COP.1.1 The TSF shall perform [***cryptographic operations stated in Table 7]*** in accordance with a specified cryptographic algorithm [***stated in Table 7***] and

cryptographic key sizes [***stated in Table 7***] that meet the following: [***stated in Table 7***].

Table 7 Cryptographic operations

Operation	Key Generation	Key Destruction	Algorithm	Key	Standard
Decryption of config.xml	Blancco	N/A	AES	128	FIPS PUB 197
Decryption of preferences.xml	Blancco	N/A	AES	128	FIPS PUB 197
Encryption and decryption of resume file	Blancco	N/A	AES	128	FIPS PUB 197
Signing a digest of an erasure report in all modes	Blancco	N/A	RSA	2048	Probabilistic Signature Scheme
Computing message digest for erasure report for the purposes of digitally signing it	N/A	N/A	SHA-256	N/A	FIPS PUB 180-4
Encrypting erasure report (in BIOS mode only) using ephemeral key	TOE	TOE	AES	128	FIPS PUB 197
Encrypting ephemeral key used for encrypting erasure report (in BIOS mode only)	Blancco	N/A	RSA	4096	PKCS #1
Decrypting USB licenses	Blancco	N/A	AES	128	FIPS PUB 197
Computing hash value of TOE executables at boot up	N/A	N/A	SHA-256	N/A	FIPS PUB 180-4

Application note: Most cryptographic operations use keys embedded into the ISO image during the manufacturing of the TOE. These keys cannot be changed and they are not destroyed by the TOE. The exception is the ephemeral key used for encrypting the erasure report when the TOE is used in the BIOS mode. This key is generated by the TOE and used for encrypting the erasure report. The ephemeral key is encrypted using a 4096 bit RSA key and sent to the BMC together with the encrypted report. The ephemeral key is then destroyed and only if the BMC has access to the private RSA key corresponding to the key that was used for encrypting the ephemeral key can the user of the BMC recover the erasure report.

7.1.2 Class FDP: User Data Protection

7.1.2.1 FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce the [***Operation SFP***] on [
Subjects: User;
Information: TOE Function;
Operations: Executing a TOE Function
].

7.1.2.2 FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [***Operation SFP***] to objects based on the following: [
Subjects: User;

Object: TOE Function;
Security attributes of subject User: Licenses held;
Security attributes of object TOE Function: Licenses required;
].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [
User is only allowed to execute a TOE Function if any of the Licenses held by the user is equivalent to the License required for the execution of the requested TOE Function
].

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [
In the BIOS configuration, the TOE shall grant access to any user to execute any TOE function without checking the licenses if the time obtained from BIOS clock is within the TOE validity period
].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [
In the BIOS mode, the TOE shall only grant access to the user to a reduced erasure report
].

Application note: Reducing access to the erasure report in BIOS mode (in PDF format) is not based on the licenses (as licenses are not used in the BIOS mode) but on the fact that sensitive fields in the erasure report are encrypted using an ephemeral key generated for the encryption of the report and destroyed immediately after use. Therefore, the user has access to the report but cannot decrypt it without use of a BMC which can recover the encrypted ephemeral key sent to it together with the erasure report.

7.1.2.3 FDP_RIP.1 Residual Information Protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource [**refinement: assured by the TOE**] is made unavailable upon the [**deallocation of the resource from**] the following objects: [***Drive selected for erasure by the user of the TOE***].

Rationale: User of the TOE allocates a resource (i.e. a drive) for erasure by the TOE when the drive is selected for erasure. At that point of time no erasure takes place yet. Deallocation occurs when the user of the TOE commences with the actual erasure by clicking the appropriate button or otherwise triggering the erasure function. Thereupon, the TOE erases that drive in accordance with the selected erasure standard and performs the necessary verifications. Because the drive selected for erasure is not part of the TOE, FDP_RIP.1 is not directly applicable and must be refined to indicate that the TOE performs a secure erasure of the data on the resource but that resource is not part of the TSF but a resource which is assured by the TOE. Therefore, the refinement is necessary to ensure precise statement of the SFR.

7.1.3 Class FIA: Identification and Authentication

7.1.3.1 FIA_UID.2 User identification before any action

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

Application Note: The user is identified through the licenses held by the user. There are asset licenses and erasure licenses. All operations require in the minimum asset license, some operations require erasure license.

7.1.4 Class FMT: Security Management

7.1.4.1 FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the [***Operation SFP***] to restrict the ability to [**modify**] the security attributes [***Licenses held***] to [***none***].

7.1.4.2 FMT_MSA.3 Static attribute initialization

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

FMT_MSA.3.2 The TSF shall allow the [***none***] to specify alternative initial values to override the default values when an object or information is created.

Application note: The default value for the licences held by the user is none (which allows no access). The licenses only exist of successfully loaded from the license storage (HASP or BMC).

7.1.5 Class FPT: Protection of the TSF

7.1.5.1 FPT_RCV.1 Manual recovery

FPT_RCV.1.1 After [***unscheduled termination of TOE execution***] the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

7.1.5.2 FPT_TEE.1: Testing of external entities

FPT_TEE.1.1/Hardware The TSF shall run a suite of tests [**at the request of an authorized user**] to check the fulfillment of [

- ***charge capacity and charge cycles of all batteries connected to the machine;***
- ***functionality of the CPU by checking its calculation capabilities;***
- ***low and the extended memory of a computer***;
- ***CMOS checksum and the CMOS battery of the motherboard;***
- ***colour reproduction and condition of the display;***
- ***functioning of the pointing devices connected to the device;***
- ***functioning of the keyboard; the functioning of the PC loudspeaker;***
- ***reading, writing and blanking capabilities of the optical devices; and***
- ***functioning of the webcam***

]-.

FPT_TEE.1.2/Hardware If the test fails, the TSF shall [***Report to the user***]-.

Rationale: Both FPT_TEE.1.1 and FPT_TEE.1.2 definitions in CC Part 2 include an extra space before the full stop. The space before the full stop is removed for consistency with other SFR statements.

FPT_TEE.1.1/Erasure The TSF shall run a suite of tests [**periodically during normal operation**] to check the fulfillment of [***complete erasure of the drive***] .

FPT_TEE.1.2/Erasure If the test fails, the TSF shall [***Report to the user***] .

Application note: Verification of the erasure is performed after each round of erasure and the outcome is reported to the user. Hardware tests are performed and outcome reported to the Operator and when so requested by the Operator. The request may take a form of configuring the TOE to perform the tests at each start-up, but the TOE only performs the checks at an explicit request, whether through the GUI or the configuration of the start-up sequence. Therefore, the selection at FPT_TEE.1.1/Hardware only includes "at the request of an authorized user".

7.1.5.3 FPT_TST.1 TSF testing

FPT_TST.1.1 The TSF shall run a suite of self tests [**during initial start-up**] to demonstrate the correct operation of [**the TSF**].

FPT_TST.1.2 The TSF shall provide authorised users with the capability to verify the integrity of [[***config.xml file; preferences.xml file***]].

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of [**TSF**].

Application note: The TOE stores encrypted the config.xml and preferences.xml files so that their integrity can be verified by the user during the boot up of the TOE. If the integrity is violated, the decryption of the files produces files which cannot be used for configuring the TOE. A hash value is computed for the executables of the TOE and stored in the ISO image. At the boot up the TOE computes a corresponding checksum and verifies it against the checksum stored. If the two differ, the boot sequence is terminated.

7.1.6 Class FTP: Trusted Paths/Channels

7.1.6.1 FTP_ITC.1 Inter-TSF trusted channel

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 [**the TSF**] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [***all communication with BMC***].

Application note: If configured to do so, the TOE establishes a HTTPS connection to the BMC for all communication

7.2 Statement of Security Assurance Requirements

Security assurance requirements for the TOE constitute the evaluation assurance package EAL2 and are fully defined with reference to CC Part 3. The security assurance requirements constituting EAL2 are the following:

- Assurance Class ADV: Development
  - ADV_ARC.1 Security architecture description
  - ADV_FSP.2 Security-enforcing functional specification
  - ADV_TDS.1 Basic design
- Assurance Class AGD: Guidance documents
  - AGD_OPE.1 Operational user guidance

- AGD_PRE.1 Preparative procedures
- Assurance Class ALC: Life-cycle support
  - ALC_CMC.2 Use of a CM system
  - ALC_CMS.2 Parts of the TOE CM coverage
  - ALC_DEL.1 Delivery procedures
- Assurance Class ASE: Security Target evaluation
  - ASE_CCL.1 Conformance claims
  - ASE_ECD.1 Extended components definition
  - ASE_INT.1 ST introduction
  - ASE_OBJ.2 Security objectives
  - ASE_REQ.2 Derived security requirements
  - ASE_SPD.1 Security problem definition
  - ASE_TSS.1 TOE summary specification
- Assurance Class ATE: Tests
  - ATE_COV.1 Evidence of coverage
  - ATE_FUN.1 Functional testing
  - ATE_IND.2 Independent testing – sample
- Assurance Class AVA: Vulnerability assessment
  - AVA_VAN.2 Vulnerability analysis

7.3 Security Requirements Rationale

7.3.1.1 Security requirement dependency rationale

Each dependency of SFRs defined for the TOE is satisfied by the TOE. The satisfaction of dependencies for each SFR is given in Table 8.

Table 8 Security Functional Requirement dependencies

SFR	Dependencies	Justification
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1 and FCS_CKM.4 by the TOE. The TOE generates an ephemeral key which is used by the TOE and destroyed immediately after use.
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1 by the TOE. The TOE generates an ephemeral key which is used by the TOE and destroyed immediately after use.
FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	The TOE implements a number of cryptographic operations of which most use static keys which are part of the .iso file and cannot be created, changed or destroyed. For these keys, none of the dependencies are applicable. Some cryptographic functions are hash functions which require no keys. The TOE only generates one of the used keys by itself: the ephemeral key used for encrypting parts of the erasure report in BIOS mode. For this key,

		the dependency to FCS_CKM.1 and the dependency to FCS_CKM.4 are satisfied as the key is generated by the TOE and destroyed immediately after use.
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1 by the TOE
FDP_ACF.1	FDP_ACC.1 FMF_MSA.3	FDP_ACC.1 by the TOE FMT_MSA.3 by the TOE
FDP_RIP.1	No dependencies	Not applicable
FIA_UID.2	No dependencies	Not applicable
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1] FMT_SMF.1 FMT_SMR.1	FDP_IFC.1 by the TOE Not applicable. The TOE does not implement management functions as all management of the TOE is done using the DECT tool prior to the TOE being executed on the Host PC. Not applicable. The TOE does not maintain roles.
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 by the TOE Not applicable. The TOE does not maintain roles.
FPT_RCV.1	AGD_OPE.1	The functionality shall be described in the operational guidance for the TOE.
FPT_TEE.1/Hardware	No dependencies	Not applicable
FPT_TEE.1/Erasure	No dependencies	Not applicable
FPT_TST.1	No dependencies	Not applicable
FTP_ITC.1	No dependencies	Not applicable

7.3.2 Tracing of security objectives to Security Functional Requirements

The tracing of security objectives to the security functional requirements is given in Table 9.

Table 9 Tracing of Security Objectives to Security Functional Requirements

SFR	O.PROTECTED_COMMS	O.COMPLETE_ERASURE	O.LEGIT_USE	OE.PLATFORM	OE.PROPER_USE	OE.PROPER_ADMIN	OE.PROPER_SETUP
FCS_CKM.1		X	X				

FCS_CKM.4		X	X				
FCS_COP.1		X	X				
FDP_ACC.1			X				
FDP_ACF.1			X				
FDP_RIP.1		X					
FIA_UID.2			X				
FMT_MSA.1			X				
FMT_MSA.3			X				
FPT_RCV.1			X				
FPT_TEE.1/Hardware			X				
FPT_TEE.1/Erasure		X					
FPT_TST.1		X					
FTP_ITC.1	X						

O.PROTECTED_COMMS concerns with the use of a trusted channel for communicating with the Blancco Management Console (BMC) that the operator may in addition to a local access via the GUI use for operating the TOE. This objective is fulfilled by the TOE being capable of establishing secure channels (namely, a HTTPS session) between itself and BMC (FTP_ITC.1).

O.COMPLETE_ERASURE concerns with ensuring that upon completion of a drive erasure either the drive is fully erased in accordance with the selected erasure standard or the operator of the TOE is provided with a reliable indication that the erasure was unsuccessful. The TOE implements a number of different erasure standards and ensures that the selected drive is erased in accordance with the selected standard (FDP_RIP.1). Prior to the bootup the TOE verifies the executables to ensure correct operation (FPT_TST.1) and upon completion of the erasure, the TOE also verifies the erasure outcome to ensure the completeness of the erasure and the operator of the TOE is notified of any discrepancy (FPT_TEE.1/Erasure). Once the TOE reports either to a local file or to the BMC the erasure results, the report is digitally signed and, in some cases, encrypted to prevent tampering with the content (FCS_CKM.1, FCS_CKM.4, FCS_COP.1).

O.LEGIT_USE concerns with ensuring that the TOE is only used in a legitimate manners: locally using the GUI or remotely using the BMC, and that sufficient countermeasure ensure that the behaviour of the TOE cannot be falsified by malicious agents. While the prevention of malicious processes from falsifying the erasure results is largely a policy concern, the TOE does implement a set of tools for verifying the basic functioning of the underlying hardware to assist the operator of the TOE in asserting the authenticity of the platform (FPT_TEE.1/Hardware). In case of an unexpected termination of the execution of TOE software, the resume file maintained by the TOE can be used for restoring the TOE to a state close to the state in which the termination occurred (FPT_RCV.1). The TOE also establishes a HTTPS session between itself and a BMC to ensure illegitimate processes may not masquerade as TOE (FTP_ITC.1).

The TOE implements access control functions to ensure that each access request is investigated and only the legitimate ones allowed. For each request, the licenses of the operator are examined, and the operation is only granted is the available licenses match the license required for the operation (FDP_ACC.1, FDP_ACF.1). By default, any operation of is disallowed unless sufficient licenses are available and the TOE cannot be configured to allow operations if licenses are not present (FMT_MSA.3, FMT_MSA.1). Users are identified through the licenses and a license is required for each operation (FIA_UID.2).

An exception to the access control enforcement is when the TOE us used in the BIOS mode. Here the TOE has no access to the user's licenses, but the access control is based on the validity of the TOE as verified against the time acquired from the BIOS clock of the Host PC. In this case, the user of the TOE may perform erasure functions, but the TOE generates an ephemeral 128-bit AES key which is used for encrypting sensitive fields in the report (FCS_CKM.1). The sensitive fields are encrypted with the ephemeral key, the document signed digitally and the ephemeral key itself encrypted with a 4096-bit RSA public key (FCS_COP.1). The ephemeral key is destroyed from the TOE immediately after completion of the encryptions where it is required (FCS_CKM.4). The protected report is sent to the BMC in which it may be recovered if the BMC has access to the RSA private key corresponding to the public key used for encrypting the ephemeral key. This regulates access to the erasure report as the user of the TOE cannot view the sensitive content until gaining access to an appropriately set up BMC.

7.3.3 Justification for the Security Assurance Requirements

The Security Assurance Requirements selected for the TOE constitute a well-defined evaluation assurance package EAL2 and as such, are an internally consistent set of security assurance requirements.

8 TOE Summary Specification

An explanation of how the TOE meets the Security Functional Requirements at the level of detail suitable for the TOE Summary Specification is given in Table 10.

Table 10 Explanation of how the TOE meets the Security Functional Requirements

SFR	Justification
FCS_CKM.1 **FCS_CKM.4** **FCS_COP.1**	The TOE includes an OpenSSL library which provides a number of cryptographic primitives through a well-defined API. The library is used for implementing the cryptographic functions of the TOE: 1. The config.xml and preferences.xml files are encrypted with 128-bit AES keys when on the ISO image of the TOE. The keys are generated at the production of the TOE and stored on the ISO image. When the TOE boots up, these keys are released, and the two files decrypted by the TOE for use. The keys cannot be generated, destroyed or changed by the TOE. 2. When enabled in the configuration of the TOE, the resume file is encrypted using a 128-bit AES key and stored on a USB token. The same key is used for decrypting the resume file when the TOE operation is restored after an unexpected interruption. The key is generated by Blancco during the production of the TOE and cannot be generated, changed or destroyed by the TOE. 3. The erasure reports are digitally signed with a 2048-bit RSA private key using a probabilistic signature scheme RSA-PSS. The TOE computes a message digest of the erasure report using SHA-256 and that message digest is digitally signed using RSA-PSS. The RSA key used for the digital signature operation is generated during the production of the TOE and cannot be re-generated, changed or destroyed by the TOE. 4. In the BIOS mode, parts of the erasure report generated by the TOE are encrypted using a TOE-generated ephemeral 128-bit AES key. Upon completion of the encryption, the ephemeral key is encrypted using a 4096-bit RSA public key which is stored in the TOE during the production of the TOE. That RSA key cannot be created, changed or destroyed by the TOE. Upon completion of the encryption of the ephemeral key, the unencrypted ephemeral key is destroyed by overwriting with zeroes and the encrypted ephemeral key, partially encrypted erasure report, and a digital signature of the erasure report (generated using a 2048-bit RSA private key as above) are sent to the BMC.

	5. When stored on a USB token, the licenses are encrypted with a 128-bit AES key generated and stored on the TOE during the production of the TOE. That key cannot be created, modified or destroyed by the TOE but can be used for decrypting the licenses fetched from a USB token. 6. At the boot up, the TOE computes a SHA-256 hash value of the executables of the TOE and compares that to a reference value on the .iso file. If the comparison fails, the boot sequence is terminated.
FDP_ACC.1 **FDP_ACF.1**	The user of the TOE has a set of licenses which are required for accessing the functions of the TOE. The licenses may be stored on the BMC or on a HASP. For each controlled operation, the license requirement is coded into the TOE software. When a user requests for an operation, the licenses of the user are retrieved and checked against the licenses required for the operation. Only of the required license exists in the licenses of the user shall the operation be executed. The TOE is predominantly using HASP or BMC for holding the credentials. However, in some cases of large scale erasure possibly in offline erasures, the TOE may be sold in the BIOS clock based configuration. In this case (i.e. neither HASP nor BMC is available) the licenses are not checked but the user is granted access to all functions if the time of the BIOS clock is within the validity period of the TOE. However, in this case the erasure report is encrypted so that the user can only view the content using a properly configured BMC.
FDP_RIP.1 **FPT_TEE.1/Erasure**	When the TOE is running in the Host PC, the user of the TOE is given a list of all drives connected to the Host PC. The user may select any of the drives and any of the available erasure methods and trigger the execution of the erasure. The TOE executes the selected erasure algorithm on the selected drive. During the erasure, the TOE displays to the user the status of the erasure of each drive. The status of each drive erasure is displayed in the Drive's Progress Bar in words and in different colour. The status may be Not Started, Ongoing, Ongoing Firmware Command, Paused, Finished, Cancelled, or Failed. Status Finished implies a successful completion of the erasure. There is, however, a possibility of a malfunctioning drive firmware signalling successful erasure when the erasure was not complete. To ensure that the outcome of the erasure can be depended on, the TOE verifies the erasure outcome prior to reporting to the user. If for any reason the verification result is negative, the TOE shall report that the erasure Failed.

FIA_UID.2	The TOE does not operate on actual identities of users. Instead, each user is identified by the set of licenses they possess. The possession of licenses is used in determining whether a user is granted access to execute requested TOE functions or not except in the BIOS mode. Each operation requires in the minimum an asset license, but some operations also require erasure license.
FMT_MSA.1 **FMT_MSA.3**	Access control decisions are made by the TOE based in the user identity (expressed as licenses) and the licenses required for executing specific TOE functions. The TOE implements no function to modify the value of licenses held. The licenses are held either by a HASP or a BMC and not controlled by the TOE. Therefore, there are no TOE functions for modifying the licenses. There are also no licenses assigned to a user by default and in case there are no licenses loaded from the license storage, the user does not hold any licenses.
FPT_RCV.1	The TOE maintains an execution state which is periodically written on a USB token and encrypted. In case of an interrupted operation of the TOE, at the next boot up the TOE shall resume the operation from the point saved in the resume file.
FPT_TEE.1/Hardware	The TOE provides a set of functions to diagnose the hardware connected to the Host PC. These diagnostic functions may assist the user of the TOE in determining whether the hardware functions correctly and the Host PC may be used for the execution of the TOE functions.
FPT_TST.1	At the boot up of the TOE, the TOE computes a has value of the TOE executables and compares them to a reference value stored on the .iso file. If the comparison fails, the TOE shall terminate the boot up as the failure implies an attempted modification of the TOE executables (or the reference checksum).
FTP_ITC.1	The TOE contains a 256-bit AES key which is used for creating a HTTPS session between the TOE and a Blancco Management Console (BMC). Upon generation of an erasure report, if the TOE is configured to use BMC, the TOE establishes a HTTPS session between itself and the BMC and uses that session for exporting a drive erasure report to the BMC. The HTTPS session is also used for fetching license data when the TOE is configured to use a BMC as a source of licenses. The BMC provides an API which is called by the TOE over a HTTPS session.
FTP_TRP.1	The TOE contains a <username, password> pair which is used for authenticating to the Blancco Management Console (BMC). The TOE establishes a HTTPS session between the TOE and the BMC. Only upon successful establishment of the HTTPS session shall the <username, password> pair be sent to the BMC for authentication and the HTTPS

	session shall be used for all communication between the TOE and the BMC.