WipeDrive Version 9.1
Security Target

Version 1.3

2019-02-07

WhiteCanyon Software, Inc.

947 South 500 East, Suite 300
American Fork, UT 84003
USA

Document Changelog

Version	Date	Summary of changes
v1.0	2018-09-06	First submission to the evaluation process.
v1.1	2018-11-26	-Changed “TOE Type” section from 1.4 to 1.3. -Removed empty cells in Table 3 and reorganized it. -Fixed sub-section numbering of Section 4 (which previously contained two sections numbered 4.1). -Relocated Dependencies in SFR definitions. -Modified section 3.3 to include augmentation information for both ALC_FLR.2 and ASE_TSS.2. -Modified the objective O.ERASE to include information about ensuring that drives are reusable and that the wipe patterns are in compliance with the corresponding standard. -Added information about P.REUSE and P.STANDARD to Table 11-2. -Renamed Table 11-2 to be “Threat/Policy to Objective Mapping”. -Modified section 11.6.3 (“FDE_ERS”) to mention device re-use. -Added OE.SYSTIME to T.AUDIT_FAILURE’s entry in Table 11-2. -In sections 6.1.1.3 and 7.2.4.3, changed references of FDE_ERS_EXT.1 to FDE_ERS_EXT.1.1. -Added square brackets to the list in section 7.2.4.3, to clarify that the options listed in bold are the list of selection parameters mentioned in section 6.1.1.3. -In FAU_GEN.1.1, changed assignment to be italicized bold. -Made references to Sanitize Device more uniform. -Converted selection text to bold in FDP_RIP.1.1. -Added “Disk Erasure” security function to Table 7-1. -Added FAU_SAR.1 as a Security Functional Component for the “Security Audit” security function in Table 10-1. -Changed order of entries in Table 10-1 and corresponding entries in sections 10.1.
v1.2	2019-01-15	-Removed references to the TUI interface due to it not being included within our evaluation.

		-Fixed notation for FDE_RIP.1 selection and assignment statements. -replaced figure 1.1 with one that reflects the changes regarding the TUI. -Added parameters for FDE_ERS_EXT.1 selection
v1.3	2019-02-07	-Minor edit to the TOE Overview regarding the definition of the LiveCD keyword -Minor edit to subsection A.LOCAL

Table of Contents

List of Figures

List of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation. This ST targets Evaluation Assurance Level 2+.

1.1.1 ST Identification

ST Title	WipeDrive Version 9.1 Security Target
ST Version	1.3
ST Publication Date:	2019-02-07
ST Author	WhiteCanyon Software, Inc.

1.1.2 Document Organization

Chapter 1 of this ST provides identifying information for the WipeDrive and it includes a ST Introduction, ST Reference, ST Identification, TOE Reference, TOE Overview, and TOE Type.

Chapter 2 describes the TOE Description, which includes the physical and logical boundaries.

Chapter 3 describes the conformance claims made by this ST.

Chapter 4 describes the Security Problem Definition as it relates to threats, Operational Security Policies, and Assumptions met by the TOE.

Chapter 5 identifies the Security Objectives of the TOE and the Operational Environment.

Chapter 6 describes the Extended Security Functional and Assurance Requirements.

Chapter 7 describes the Security Functional Requirements (SFRs).

Chapter 8 describes the Security Assurance Requirements (SARs).

Chapter 9 is the TOE Summary Specification (TSS), a description of the functions provided by WipeDrive to satisfy the security functional and assurance requirements.

Chapter 10 is the TOE Summary Specification Rationale and provides a rationale, or pointers to a rationale, for security objectives, assumptions, threats, requirements, dependencies, and PP claims.

Chapter 11 is the Security Problem Definition Rationale and provides a rationale for the chosen EAL, any deviations from CC Part 2 with regards to SFR dependencies, and a mapping of threats to assumptions, objectives, and SFRs. It also identifies the items used to satisfy the Security Assurance Requirements for the evaluation.

1.1.3 Terminology

This section defines the terminology used throughout this ST. The terminology used throughout this ST is defined in Table 1-1: Terminology Definitions. This table is to be used by the reader as a quick reference guide for terminology definitions.

Terminology	Definition
Administrator	Any user of the TOE who maintains physical possession of the WipeDrive application
Administrator Definable Wipe Pattern	A concatenation of static primitives that is not persistent between boots.
ATA HPA	ATA Host Protected Area Refers to as a hidden protected area that is a section of a hard drive that is not normally visible to an Operating System
Kernel	The central component for most Operating Systems (in this case, Linux) that is primarily responsible for starting and stopping programs, handling the file system, as well as other low-level tasks most programs share.
LBA28/LBA48	A common scheme used for specifying the location of blocks of data stored on computer storage devices, generally secondary storage systems such as hard disks. LBA28/LBA48, in particular, refers to a logical block address that is 28- or 48-bits wide, resulting in a disk size limit.
LiveCD	A Linux-based compact disc/USB based on the Gentoo meta-distribution which is configured to start WipeDrive upon booting up. This term can also refer to the running application within memory disc of the target computer.
Log/Logging	Synonymous with audit/auditing

Preboot eXecution Environment (PXE)	An environment to boot computers using a network interface independently of available data storage devices (e.g. hard disks) or installed Operating Systems.
WhiteCanyon	Vendor
WipeDrive	Product

Table 1-1: Terminology Definitions

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 1-2: Acronym Definitions. This table is to be used by the reader as a quick reference guide for acronym definitions.

Acronym	Definition
ATA	Advanced Technology Attachment
BIOS	Basic Input/Output System
CC	Common Criteria
CCEVS	Common Criteria Evaluation and Validation Scheme
DCO	Device Configuration Overlay
DHCP	Dynamic Host Configuration Protocol
FTP	File Transfer Protocol
FMT	Functional Security Management
GUI	Graphical User Interface
GNU	Recursive acronym for *GNU's Not Unix*
HPA	Host Protected Area
JSON	JavaScript Object Notation
LBA	Logical Block Addressing
OS	Operating System
PXE	Preboot eXecution Environment
RPC	Remote Procedure Call Protocol
SCSI	Small Computer System Interface
SQL	Structured Query Language
TOE	Target of Evaluation
TSF	TOE Security Functionality
EXE	Windows Executable
UI	User Interface
TUI	Textual User Interface

Table 1-2: Acronym Definitions

1.1.5 References

[1] WipeDrive User Guide

[2] WipeDrive Admin Guide

[3] US DoD 5220.22-M (http://www.dss.mil/documents/odaa/nispom2006-5220.pdf)

[4] HMG IA Standard No. 5: Secure Sanitisation

[5] "Media Sanitation of the Technical Security Standards for Information Technology", RCMP (Royal Canadian Mounted Police)

[6] Army Regulation 380-19 (https://fas.org/irp/doddir/army/r380_19.pdf)

[7] Air Force System Security Instruction 5020

[8] German VSITR

[9] US Navy Staff Office Publication P-5329-26

[10] US National Computer Security Center TG-025

[11] CIS GOST P50739-95

[12] Australian Defense Signals Directorate ACSI-33 (X0-PD)

[13] NNSA NAP 14.1-C

[14] Canadian CSEC ITSG-06 (https://www.cse-cst.gc.ca/en/node/270/html/10572)

[15] US Air Force System Security Instruction 8580

[16] BSI-2011-VS

[17] NIST Special Publication 800-88 Revision 1: Guidelines for Media Sanitization (https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-88r1.pdf)

[18] Common Criteria for Information Technology Security Evaluation – "Evaluation methodology", CCMB-2017-04-004, Version 3.1 Revision 5, April 2017 (https://www.commoncriteriaportal.org/files/ccfiles/CEMV3.1R5.pdf)

[19] Common Criteria for Information Technology Security Evaluation – "Part 1: Introduction and general model", CCMB-2017-04-001, Version 3.1 Revision 5, April 2017 (https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R5.pdf)

[20] Common Criteria for Information Technology Security Evaluation – "Part 2: Security functional components", CCMB-2017-04-002, Version 3.1 Revision 5, April 2017 (https://www.commoncriteriaportal.org/files/ccfiles/CCPART2V3.1R5.pdf)

[21] Common Criteria for Information Technology Security Evaluation – "Part 3: Security assurance components", CCMB-2017-04-003, Version 3.1 Revision 5, April 2017 (https://www.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R5.pdf)

1.2 TOE Reference

The TOE is WipeDrive Version 9.1.

1.3 TOE Type

The TOE type for WipeDrive version 9.1 is a software application that provides Sensitive Data Protection. By completely and permanently erasing sensitive data from digital storage media, the TOE protects a user's data from being unwillingly disseminated when the media is repurposed, sold, or discarded.

1.4 TOE Overview

This Security Target (ST) defines the Information Technology (IT) security requirements for the TOE. The TOE is a Disk Sanitizing tool that permanently erases hard drive data, operating systems, program files, and all other file data from a system. WipeDrive also provides users with the ability to permanently delete all partitions and drive formats previously configured. The TOE provides 20 disk wipe functions:

- Standard Overwrite
- US DoD 5220.22-M 3-pass
- US DoD 5220.22-M 7-pass
- GB HMG Infosec Standard #5 Baseline
- GB HMG Infosec Standard #5 Enhanced
- Canadian RCMP TSSIT OPS-II Standard Wipe
- US Army AR380-19
- US Air Force System Security Instruction 5020
- German VSITR
- US Navy Staff Office Publication P-5329-26
- US National Computer Security Center TG-025
- CIS GOST P50739-95 version 2
- Australian Defense Signals Directorate ACSI-33 (X0-PD)
- SecureErase + 1 overwrite with verify or NNSA NAP 14.1-C
- Canadian CSEC ITSG-06
- US Air Force System Security Instruction 8580
- BSI-2011-VS
- SSD Smart wipe

- NIST 800-88r1
- Custom overwrite pattern

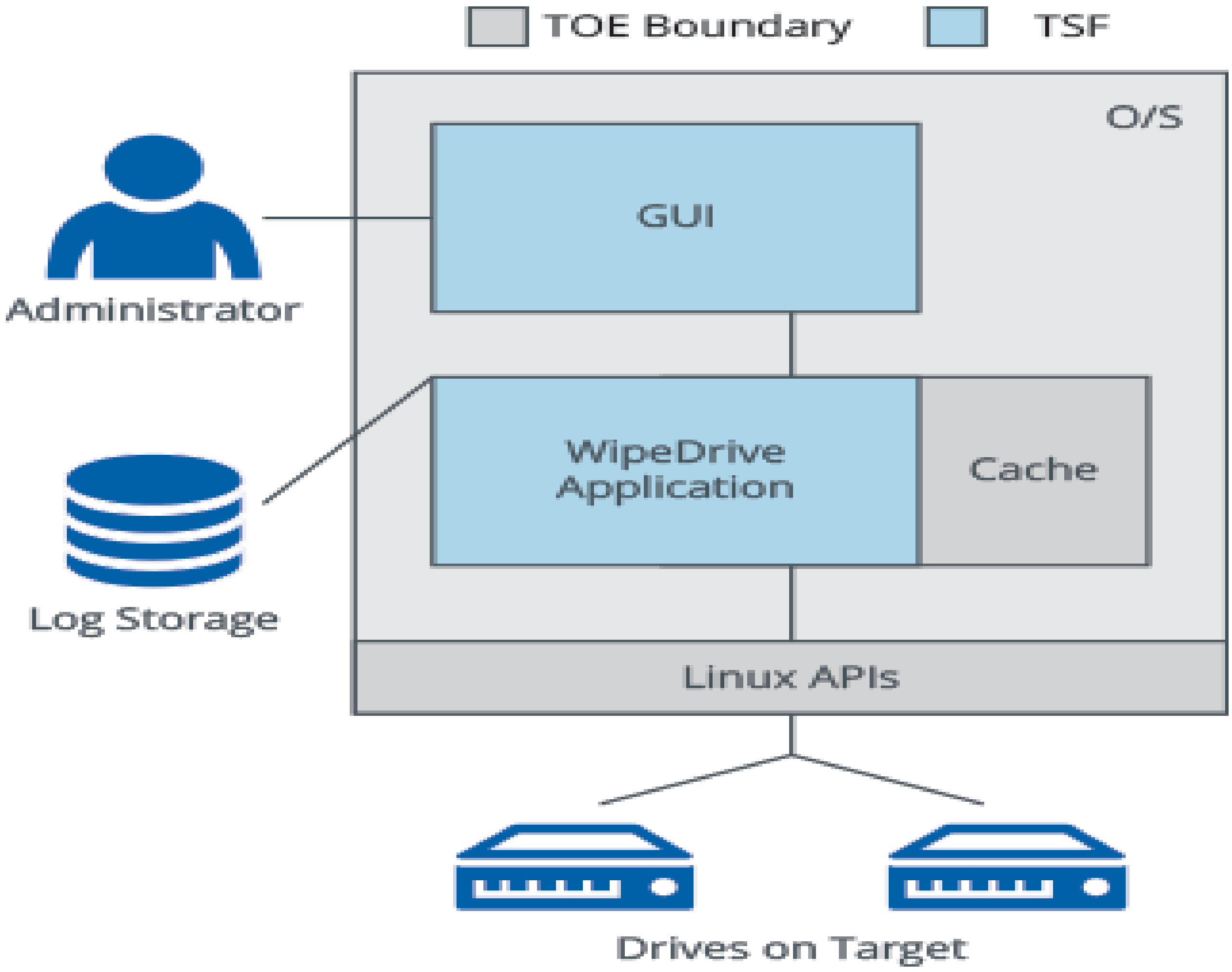
For more information on these wipe methods, please refer to Figure 9-1.

All wipe functions overwrite disk storage, or use special erasure commands native to the drives, to ensure no residual data remains. After the sanitization process has been completed, an audit log is created which compiles verifications that the information contained on the hard drive was in fact erased.

The TOE:

- Is a Linux based OS booted from a LiveCD, which resides in memory during runtime.
- Is a data protection and erasure tool that permanently wipes data from ATA, SCSI, USB, eMMC, SD, and NVMe-block devices. This includes traditional platter drives as well as SSDs.
- Allows users to create an audit log to capture verifications of the success or failure of hard drive erasure events
- Has the ability to wholly erase Operating Systems, program files, and all file data
- Utilizes user interfaces to allow administrators to graphically see the progress of probing, scanning, and erasure events
- Enables administrators to view sector data

Figure 1-1: TOE Boundary



As shown in Figure 1-1, administrators access the GUI in order to run the executable file for the WipeDrive application. Once the WipeDrive application has been executed, the cache stores data about scanned and probed devices in order to display the data to users. Scanning and probing are both performed during the initialization of the TOE. The WipeDrive application performs a scanning operation to discover attached devices. For each device that is discovered, a probe operation is run to enumerate device information.

The only users of the TOE are referred to as administrators. Administrators can execute commands to wipe drives by using the administrator definable wipe patterns. Verification of the success or failure of the wipe event is sent to the UI the user is currently using. Also, the audit log data collected from the wipe event is stored in/on a log storage device, which can be a portable flash/thumb drive, FTP server, SQL database (MySQL or MS SQL), Windows Share directory, or other media storage device

2 TOE Description

2.1 Evaluated Components of the TOE

The TOE provides for 4 distinct components.

2.1.1 WipeDrive Application

The WipeDrive application serves as a single executable file that is primarily responsible for:

- scanning the system for devices that can be erasure targets
- probing the discovered devices for capabilities
- erasing the devices, and performing related operations (such as removing ATA HPA, DCO areas, or Accessible Max Address settings)
- producing progress event messages for consumption by a UI for display to the user
- producing result messages for consumption by UI
- performs logging after the erasure of the media has completed

Note: Only a single WipeDrive application will be able to run on any single host at any one time.

2.1.2 User Interfaces (UI)

The user interface serves as the physical interfaces where controls are used to operate one or more instances of the back-end, each on a distinct host. The interfaces that are included in the evaluated configuration are:

- **GUI** – A graphical UI that is run on the same host as the back-end. This will be the default interface for x86 machines that framebuffer can be accessed.

2.1.3 Linux APIs

Linux APIs provide a logical interface between the application and the target drive(s). For example, when the TOE scans a disk, it relies on Linux to gather some of the data. This is a built-in function of the Operating System.

2.1.4 Third Party Programs

Optionally included with the Linux operating system are various programs that provide functionality utilized by WipeDrive.

2.2 Components in the Operational Environment

2.2.1 Log Storage

The Log Storage component is responsible for the storage of audit information. Log Storage refers to any external device with a file system that the TOE can access. Examples of these are a USB drive or a separate hard disk or partition upon the local machine being wiped.

2.2.1.1 Log Storage Formats

The Log Storage component supports several formats:

- **Regular** – a plain-text synopsis (free-form) of what activities were attempted and their result
- **Comma Separated Values (CSV) –** a plain-text file, delimited by commas, of what activities were attempted and their result in a tabular format
- **XML –** an XML file that contains both the activities that were attempted and their result as well as a brief system inventory harvested via invoking the lshw Linux utility
- **PDF –** a PDF file that contains both the activities that were attempted and their result; can optionally include a brief system inventory
- **HTML –** an HTML file that contains both the activities that were attempted and their result; can optionally include a brief system inventory
- **SQL –** a SQL query that inserts log data into a target database
- **Bootable Report –** a report that is written directly to the target drive in a way that when a system attempts to boot from the drive, the report is displayed

2.3 Non-TOE Hardware/Software/Firmware required by the TOE

The following non-TOE hardware/software/firmware are required by the TOE:

2.3.1 Hardware

WipeDrive must be run on hardware that meets or exceeds the requirements outlined in section 2.5. The hardware also must have physical connections that work with the media that is to be wiped.

2.3.2 Software

WipeDrive is an application that runs on an operating system. As noted in section 2.1.4, there are various 3rd party applications that may be utilized by WipeDrive. These include busybox, lshw, hdparm, smartctl, raid tools, sedutil-cli, and the Linux operating system.

2.3.3 Firmware

WipeDrive assumes that functional firmware is already loaded on the hard drives. The BIOS firmware must also be functioning correct in order for WipeDrive to be run.

2.4 Excluded From the TOE

The following optional products and components can be integrated with WipeDrive but are NOT included in the evaluated configuration. They provide no added security related functionality. They are separated into three categories: not installed, installed but requires a separate license, and installed but not part of the TSF.

2.4.1 Not Installed

There are no additional components for WipeDrive version 9.1 that are not installed.

2.4.2 Installed but Requires a Separate License

These components may be installed with WipeDrive v9.1 but require a separate license and are therefore not included in the TOE boundary.

VeriDrive – verifies that media have been wiped of data. VeriDrive is outside of the scope of this evaluation because only the wiping of ATA, SCSI, USB, eMMC, SD, and NVMe-block devices are being evaluated. Additionally, this component requires an additional license which is not provided in the evaluated configuration.

2.4.3 lnstalled But Not Part of the TSF

These components are installed with WipeDrive v9.1 but are not a part of the TSF.

Network GUI – A GUI that is run on any machine (usually on a PXE server) that can communicate with and control any number of WipeDrive applications running on other hosts. This is not part of the TSF because the communications are not secured by WipeDrive and the threat of being able to wipe remote hosts introduces an increased amount of risk.

TUI – A text-based UI, run on the same host as the back-end as part of the WipeDrive application. It is used primarily for systems that do not have framebuffer support – which is typical on many architectures other than x86.

Note that while the Network GUI capabilities may be technically “installed” as they are part of the LiveCD/EXE media, the act of deploying WipeDrive as a network-capable product requires a deliberate configuration effort on a dedicated server. The standard usage of the TOE via single session instances do not run a risk of “accidentally” utilizing or deploying this functionality.

2.5 Physical Boundary

The TOE includes the following hardware and software components:

2.5.1 Hardware Components

The following table identifies WipeDrive's hardware components and indicates whether or not each component is in the TOE.

TOE or Environment	Component	Description
TOE	LiveCD / EXE	A 130 MB Linux-based program based on the Gentoo meta-distribution which is configured to start WipeDrive upon booting up. Contained on the media is an executable file that takes initial input parameter, modifies the boot loader in order to add a Gentoo RAM disc, then reboots the system into the disc where the program is run.
Environment	Supported Operating Systems	Windows Mac PC running Linux UNIX
	System Requirements	CPU – 1 GHz RAM – 1 GB Other: • SVGA or higher video support
	Target device(s)	ATA, SCSI, USB, eMMC, SD, and NVMe-block device that has been identified as a candidate for erasure.
	Log Storage	Location in which the audit data is stored and is located separately from the TOE. The data can be stored on any form of file storage medium.

	External Server	A physical server that can utilize FTP or SQL to optionally be used to store logs of erasure events in lieu of the log storage file if desired.

Table 3 – Hardware Requirements for the TOE

Note: Newer computers often have a BIOS feature to 'enable' or 'disable' devices in the boot order list. It is important that the CD drive be enabled for WipeDrive to wipe the drive when using the LiveCD boot method. Even if the boot order is properly set to boot from CD before the hard drive, and the CD is a valid copy, WipeDrive will only run if the CD drive is 'enabled' in BIOS.

2.5.2 Memory Requirements

The following table identifies WipeDrive's memory requirements for the UNIX and Windows Operating System.

Component	**Operating System**	
	UNIX Variant	**Windows**
Memory (RAM)	1 GB	1 GB

Table 4 – Memory Requirements for the TOE

Note: In addition to the above requirements, storage space is needed for the log file. This could be stored on any form of file storage medium. The log file will contain the type of wipe, the size of the hard drive, and the timestamp of the wipe.

2.5.3 Software Components

The following table identifies WipeDrive's software components and indicates whether or not each component is in the TOE.

TOE or Environment	Component	Description
TOE	WipeDrive Version 9.1	The disk erasure tool currently being evaluated.
	GUI	Receives user commands in order to display options to users to run specific wipe operations. The GUI runs locally on the same host as the WipeDrive application (back-end).

	Linux API	Provides a logical interface between the WipeDrive software and target(s) on a drive.
Environment	Log Storage file	A flat file where verification of the success or failure of erasure events are stored.
	FTP Server	A remote FTP server which may be used to receive log data.
	SQL Server	A remote database SQL server which may be used to receive log data.

Table 5 – Software Components for the TOE

2.6 Logical Boundary

The logical boundaries of the TOE are described in the terms of the security functionalities that the TOE provides to the systems that utilize this product for information flow control.

The logical boundary of the TOE will be broken down into four security classes: Security Audit, Security Management, Disk Erasure, and User Data Protection. Listed below are the security functions with a listing of the capabilities associated with them:

2.6.1.1 Security Audit

The TOE generates and captures audit data which is used to provide further verification that an erasure event has occurred. Audit logs containing verification data (either denoting a success or failure) are stored internally to the WipeDrive application. The resulting output of a wipe operation is displayed in an easily interpretable manner. All audit operations can be associated with the administrator who performed that event. The TOE saves the audit events in a user-readable format outside of the TOE but is not responsible for facilitating the viewing of audit records except for a review of wipe results immediately following a wipe operation.

A “SecurityCode” is provided with XML and SQL logging, which is a way of determining if the log has been modified in a non-authorized way.

2.6.1.2 Security Management

The only users of the TOE are referred to as administrators. Administrators are the individuals who maintain physical access to the WipeDrive application, and, as a result, possess several management capabilities. Administrators are able to specify the location for audit storage (in the Log Storage component), specify the format in which this data is stored, create, run, view, or delete an administrator definable wipe pattern, scan for devices, view sector data, and get device info for all devices previously scanned.

The TOE is equipped to operate via various interfaces which are made available to administrators. The administrators of the TOE utilize these interfaces to perform the management functions listed above. The primary purposes of these interfaces are to:

1. Allow commands defined by the TOE to be invoked on the attached WipeDrive application;
2. Visually display the status of the attached WipeDrive application by interpreting the responses and notifications received; and
3. Create audit logs according to the user's preferences. The logs can be stored on any form of media that the user desires (e.g. a thumb drive or on an FTP server).

The TOE is primarily operated via the GUI interface. The GUI is also run on the same host as the back-end. This will be the default interface for x86 machines that framebuffer can be accessed.

2.6.1.3 Disk Erasure

The TOE is able to perform three distinct operations under the guise of Disk Erasure – scanning of devices, probing of devices, and the erasure of the devices. Scanning and probing are both performed during the initialization of the TOE while the probe operation is run each time a device is discovered. Administrators can execute commands via the GUI to wipe drives. The wipe command applies the administrator definable wipe pattern to each selected disk instance, which performs the overwrite operations directly on the disk.

2.6.1.4 User Data Protection

The TOE provides for the erasure of residual information. This erasure is initiated at the user-facing interfaces and requires communication with the information repository (disk). The erasure of residual information is performed when a deviceOp instance is executed – which is a direct result an administrator definable wipe pattern. No residual information will reside in the RAM subsequent to a wipe event.

2.6.1.5 Secure Boot

The WipeDrive ISO supports Secure Boot. If the computer's BIOS supports it, Secure Boot will check the signature of important OS files at boot time to make sure that they haven't been modified. Since the Linux OS files are included with the WipeDrive program, this helps ensure that the OS and kernel that the WipeDrive application and UI are running on top of wasn't altered before being booted.

3 Conformance Claims

3.1 CC Version

This ST is compliant with *Common Criteria for Information Technology Security Evaluation*, CCMB-2017-04-004, Version 3.1 Revision 5, April 2017

3.2 CC Part 2 Extended

This ST and Target of Evaluation (TOE) is CC Part 2 [20] extended for EAL2 to include all applicable NIAP and International interpretations through April 2017.

3.3 CC Part 3 Augmented

This ST and Target of Evaluation (TOE) is CC Part 3 [21] augmented with ALC_FLR.2 and ASE_TSS.2 for EAL2 to include all applicable NIAP and International interpretations through April 2017.

3.4 PP Claims

This ST does not claim Protection Profile (PP) conformance.

3.5 Package Claims

This TOE has a package claim of EAL 2.

3.6 Package Name Conformant or Package Name Augmented

This ST and Target of Evaluation (TOE) is conformant to EAL package claims augmented with ALC_FLR.2 and ASE_TSS.2**.**

3.7 Conformance Claim Rationale

There is no Conformance Claim rationale for this ST.

4 Security Problem Definition

4.1 Threats

The TOE itself has threats and the TOE is also responsible for addressing threats to the environment in which it resides. The assumed level of expertise of the attacker for all the threats is moderately sophisticated. The following are threats addressed by the TOE.

T.ADMIN_ERROR

An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.

T.AUDIT_COMPROMISE

A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.

T.AUDIT_FAILURE

A malicious user or process failure may cause the TOE to fail to record or improperly record audit data, thus masking a user's action.

T.RESIDUAL

Any person with access to a target environmental resource can access residual data, either due to a wipe operation being incomplete or a completed wipe operation being insufficient.

T.UNAUTH

An unauthorized user obtains the physical medium which contains the TOE and uses it to perform a wipe operation against an environmental resource which there has been no authorization to wipe.

4.2 Organizational Security Policies

The TOE addresses the organizational security policies described below.

P.REUSE

All drive data must be securely erased to allow the reuse of drives.

P.STANDARD

The TOE will be used to securely erase drive data in conformance with the standards of the organization.

4.3 Assumptions

The specific conditions listed in this section are assumed to exist in the environment in which the TOE is deployed. These assumptions are necessary as a result of practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

4.3.1 Personnel Assumptions

A.ADMIN

One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains.

A.NOEVIL

Users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.

4.3.2 Physical Assumptions

A.LOCAL

The TOE will be loaded onto the same physical machine as the target resource so that commands are not exposed over the network.

A.LOCATE

The physical medium which contains the TOE will be located in a secure location and physical custody is maintained by one or more authorized administrators.

4.3.3 Logical Assumptions

A.PATCHES

Administrators of the Operational Environment exercise due diligence to acquire updated versions of the TOE and patch the Operational Environment (e.g., OS and database) so they are not susceptible to attacks resulting in malfunction of the TOE or associated audit data.

5 Security Objectives

5.1 Security Objectives for the TOE

The following security objectives are to be satisfied by the TOE.

O.AUDIT

The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.

O.ERASE

The TOE will provide measures for erasing data contained on block devices on a target system as well as sufficient assurance that the desired data was erased, and that the erasure method was sufficient for permanent erasure. The TOE must erase the data in such a way that the block devices are reusable (i.e. the device is not destroyed) and in conformance with the standards of the wipe pattern selected by the administrator.

O.MANAGE

The TOE will provide authorized administrators with the resources to manage and monitor the set of disk wipe patterns made available to the TOE and the storage of audit data generated by the TOE.

O.ROBUST_ADMIN_GUIDANCE

The vendor will provide administrators with the necessary information for secure delivery and management of the TOE.

5.1.1 Security Objectives for the Operational Environment of the TOE

The following security objectives for the Operational environment of the TOE must be satisfied in order for the TOE to fulfill its security objectives.

OE.ADMIN

One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains.

OE.LOCAL

The TOE will be loaded onto the same physical machine as the target resource so that commands are not exposed over the network.

OE.LOCATE

The physical medium which contains the TOE will be located in a secure location and physical custody is maintained by one or more authorized administrators.

OE.NOEVIL

Users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the organization’s guidance documentation.

OE.PATCHES

Administrators of the Operational Environment exercise due diligence to acquire updated versions of the TOE and patch the Operational Environment (e.g., OS and database) so they are not susceptible to attacks resulting in malfunction of the TOE or associated audit data.

OE.SYSTIME

The operating environment will provide reliable system time.

6 Extended Security Functional Requirements

6.1 Extended Security Functional Requirements for the TOE

Security Function	Security Functional Components
Disk Erasure	FDE_SCN_EXT.1 Scan of Devices
	FDE_PRB_EXT.1 Probe of Devices
	FDE_ERS_EXT.1 Erasure of Devices

Table 6-1: Extended Security Functional Requirements for the TOE

6.1.1 Class FDE: Disk Erasure

The FDE family defines requirements for the scanning, probing, and erasure of devices. This family identifies the types of disk erasures capable of being performed (FDE_ERS_EXT.1) by enumerating the methods made available to users of the TOE. In addition to listing the wipe patterns the TOE can perform, the FDE_SCN_EXT.1 requirement scans a system for potential erasure targets. The FDE_PRB_EXT.1 requirement communicates with the devices that were located subsequent to the scan operation in order to discover that target's parameters. Both the FDE_SCN_EXT.1 and FDE_PRB_EXT.1 extended requirements are performed automatically as the WipeDrive application is run.

FDE_SCN_EXT.1 Scan of devices, requires the TSF to discover storage devices on a system based on some criteria.

FDE_PRB_EXT.1 Probe of devices, requires the TSF to identify certain parameter values on the devices it has discovered from the scanning process.

FDE_ERS_EXT.1 Erasure of devices, requires the TSF to erase devices which have been scanned and probed according to some specific overwrite sequence.

Management: FDE_SCN_EXT.1, FDE_PRB_EXT.1, FDE_ERS_EXT.1

The following actions could be considered for management actions in FMT:

a) Initiating a scan, probe, or erasure
b) Viewing the results of a scan (or the combination of a scan and a probe)
c) Managing (create/view/edit/delete) administrator definable wipe patterns

Audit: FDE_ERS_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Not Specified: target device erased, identify of individual performing erasure, type of erasure performed, data about the device, errors encountered during erasure operation.

The FDE_SCN_EXT.1, FDE_PRB_EXT.1, and FDE_ERS_EXT.1 requirements are extended because there is no security functional requirement described in CC Part 2 that directly applies to the primary functionality of the TOE. The most closely related security functional requirement in CC Part 2 is the FDP_RIP.1 requirement, which speaks to the allocation and deallocation of resources on a given system. This requirement does indeed speak to a portion of the primary functionality of the TOE, however it does not encapsulate all of the TOE's purpose. It is necessary to introduce the Disk Erasure family in order to fully capture the most important functionality of the TOE, i.e. the scanning of devices, the probing of devices, and the erasure of targets on a given drive.

6.1.1.1 FDE_SCN_EXT.1 Scan of Devices

Hierarchical to: No other components.

FDE_SCN_EXT.1.1 The TSF shall be able to discover all [***assignment: list of devices***] on a system as potential erasure targets.

Dependencies: No dependencies

6.1.1.2 FDE_PRB_EXT.1 Probe of Devices

Hierarchical to: No other components.

FDE_PRB_EXT.1.1 The TSF shall communicate with devices that are discovered as the result of the scan in order to determine the following parameters for the device:

- [***assignment: list of parameters for the device***]

Dependencies: FDE_SCN_EXT.1 Scan of devices

6.1.1.3 FDE_ERS_EXT.1 Erasure of Devices

Hierarchical to: No other components.

FDE_ERS_EXT.1.1 The TSF shall be able to erase devices that are discovered by a scan using one of the following sequences:

- **[Standard Overwrite**
- **US DoD 5220.22-M 3-pass**

- **US DoD 5220.22-M 7-pass**
- **GB HMG Infosec Standard #5 Baseline**
- **GB HMG Infosec Standard #5 Enhanced**
- **Canadian RCMP TSSIT OPS-II Standard Wipe**
- **US Army AR380-19**
- **US Air Force System Security Instruction 5020**
- **German VSITR**
- **US Navy Staff Office Publication P-5329-26**
- **US National Computer Security Center TG-025**
- **CIS GOST P50739-95 version 2**
- **Australian Defense Signals Directorate ACSI-33 (X0-PD)**
- **SecureErase + 1 overwrite with verify or NNSA NAP 14.1-C**
- **Canadian CSEC ITSG-06**
- **US Air Force System Security Instruction 8580**
- **BSI-2011-VS**
- **SSD Smart wipe**
- **NIST 800-88r1**
- **Custom overwrite pattern**]

Dependencies: FDE_SCN_EXT.1 Scan of Devices
FDE_PRB_EXT.1 Probe of Devices

6.2 Proper Dependencies

There are no dependencies that were derived from CC Part 2.

6.3 Extended Security Assurance Requirements

There are no extended Security Assurance Requirements in this ST.

7 Security Functional Requirements

7.1 Operations Defined

The notation, formatting, and conventions used in this security target (ST) are consistent with version 3.1 of the Common Criteria for Information Technology Security Evaluation.

All of the components in this ST are taken directly from Part 2 of the CC except the ones noted with "_EXT" in the component name. Font style and clarifying information conventions were developed to aid the reader.

The CC permits four functional component operations: assignment, iteration, selection, and refinement to be performed on functional requirements. These operations are defined in Common Criteria, Part 1 as:

7.1.1 Assignments Made

An assignment allows the specification of parameters and is specified by the ST author in [***italicized bold text***].

7.1.2 Iterations Made

An iteration allows a component to be used more than once with varying operations and is identified with the iteration number within parentheses after the short family name, FAU_GEN.1(1), FAU_GEN.1(2).

7.1.3 Selections Made

A selection allows the specification of one or more items from a list and is specified by the ST author in [**bold text**].

7.1.4 Refinements Made

A refinement allows the addition of details and is identified with "Refinement:" right after the short name. ~~The old text is shown with a strikethrough~~ and ***<u>the new text is</u> <u>specified by italicized bold and underlined text</u>***.

7.2 Security Functional Requirements for the TOE

Security Function	**Security Functional Components**
Security Audit	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
	FAU_SAR.1 Audit Review
Security Management	FMT_SMF.1 Specification of Management Functions
User Data Protection	FDP.RIP.1 Residual Information Protection

Disk Erasure	FDE_SCN_EXT.1 Scan of Devices
	FDE_PRB_EXT.1 Probe of Devices
	FDE_ERS_EXT.1 Erasure of Devices

Table 7-1: Security Functional Requirements for the TOE

7.2.1 Class FAU: Security Audit

7.2.1.1 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable Time Stamps

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the [**not specified**] level of audit; and

c) **[*erasure events*]**

Application Note: The audit functionality consists of two different parts:

> *The first is an audit log file that can be created by WipeDrive. If the program is configured to do so, this audit log will include information about the wiping of the hard drive (as defined in this section under FAU_GEN.1.2). Note that this functionality can be turned off by the user.*
>
> *The second is an audit file that is part of Linux's syslog file. WipeDrive will write to the syslog file when WipeDrive starts and finishes, when WipeDrive creates an audit log with wipe information, and if WipeDrive isn't configured to create an audit log. WipeDrive writing to Linux's syslog file cannot be disabled and runs automatically. An administrator can use Linux commands to copy and permanently store the syslog file.*

Application Note: System time is provided by the BIOS hardware clock. The BIOS clock is synchronized with the system clock contained on the Linux LiveCD.

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

d) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

e) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [***WipeDrive version number, drive model information, serial number of physical drive, current user (user-defined, name of person performing the wipe), computer ID (user-defined, name of the drive or system), type of operation performed (***administrator definable wipe pattern***), number of overwrites performed, date and time operation was completed, total elapsed time, operation result, total number of disk sector read/write errors, if any; total uncleaned or unreadable disk sectors, if any; number of Secure Erase passes, if any; number of Sanitize Device passes, if any; NIST Method type, if applicable; drive type (e.g. Platter, SSD)***].

Dependencies: No dependencies

Application Note: Users are self-identified when performing operations therefore the TOE does not ensure correct authentication.

7.2.1.2 FAU_GEN.2 User Identity Association

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit Data Generation
FIA_UID.1 Timing of Authentication

FAU_GEN.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

7.2.1.3 FAU_SAR.1 Audit Review

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit Data Generation

FAU_SAR.1.1 The TSF shall provide [***administrators***] with the capability to read [***information about the most recent wipe performed during the active session***].

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

7.2.2 Class FMT: Security Management

7.2.2.1 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [

- ***Specify location for audit storage***
- ***Specify format for log storage***
- ***Create an administrator definable wipe pattern***
- ***Delete an administrator definable wipe pattern***
- ***Run an administrator definable wipe pattern***
- ***View all administrator defined wipe patterns***
- ***Scan for devices***
- ***View sector data***
- ***Get device info for all devices previously scanned***
- ***Configure licensing options (e.g. Cloud code account, security dongle)*]**

Application Note: The security management functions can be initiated via the GUI

Application Note: Disk scanning and probing are processes that are started automatically upon initialization of the TOE.

Application Note: The TOE requires that a license be consumed for each media drive that WipeDrive erases. For example, if WipeDrive is to erase 3 media drives, then 3 licenses must be consumed before the erasure can begin. These licenses are stored on secure servers and can be accessed only by a unique 16-character "Cloud code". The server can be accessed directly by the TOE.

7.2.3 Class FDP: User Data Protection

7.2.3.1 FDP.RIP.1 Residual Information Protection

Hierarchical to: No other components

Dependencies: No dependencies

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a block device upon which the TSF is acting is made unavailable upon [**allocation of the resources to**] the following objects: [***any object created as the result of an administrator definable wipe pattern***].

Application Note: The administrator definable wipe pattern determines the target resource(s) and the method(s) by which its previous information content will be made unavailable. Residual data will be removed from system memory as well as the target resource(s).

7.2.4 Class FDE: Disk Erasure

7.2.4.1 FDE_SCN_EXT.1 Scan of Devices

Hierarchical to: No other components.

Dependencies: No dependencies

FDE_SCN_EXT.1.1 The TSF shall be able to discover all [***ATA, SCSI, USB, eMMC, SD, and NVMe-block devices***] on a system as potential erasure targets.

7.2.4.2 FDE_PRB_EXT.1 Probe of Devices

Hierarchical to: No other components.

Dependencies: FDE_SCN_EXT.1 Scan of devices

FDE_PRB_EXT.1.1 The TSF shall communicate with devices that are discovered as the result of the scan in order to determine the following parameters for the device:

- [***Model Name***
- ***Name of manufacturer***
- ***Serial Number***
- ***Drive Capacity***
- ***Drive Type (Platter, SSD, etc.)***
- ***Drive Capabilities and State (Secure Erase, Sanitize Device, number of reallocated sectors, etc.)]***

7.2.4.3 FDE_ERS_EXT.1 Erasure of Devices

Hierarchical to: No other components.

Dependencies: FDE_SCN_EXT.1 Scan of Devices

FDE_PRB_EXT.1 Probe of Devices

FDE_ERS_EXT.1.1 The TSF shall be able to erase devices that are discovered by a scan using one of the following sequences:

- **[Standard Overwrite**
- **US DoD 5220.22-M 3-pass**
- **US DoD 5220.22-M 7-pass**
- **GB HMG Infosec Standard #5 Baseline**
- **GB HMG Infosec Standard #5 Enhanced**
- **Canadian RCMP TSSIT OPS-II Standard Wipe**
- **US Army AR380-19**
- **US Air Force System Security Instruction 5020**
- **German VSITR**
- **US Navy Staff Office Publication P-5329-26**
- **US National Computer Security Center TG-025**
- **CIS GOST P50739-95 version 2**
- **Australian Defense Signals Directorate ACSI-33 (X0-PD)**
- **SecureErase + 1 overwrite with verify or NNSA NAP 14.1-C**
- **Canadian CSEC ITSG-06**
- **US Air Force System Security Instruction 8580**
- **BSI-2011-VS**
- **SSD Smart wipe**
- **NIST 800-88r1**
- **Custom overwrite pattern**]

8 Security Assurance Requirements

This section identifies the Security Assurance Requirement components met by the TOE. These assurance components meet the requirements for EAL2 augmented with ALC_FLR.2 and ASE_TSS.2.

Assurance Class	Assurance components

ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
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 **ALC_FLR.2 Flow reporting procedures**
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.2 TOE summary specification with architectural design summary
ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

The augmented components are set in **bold**.

9 TOE Summary Specification

The following sections identify the security functions of the TOE. They include Security Audit, Disk Erasure, Security Management, and User Data Protection,

9.1 Security Audit

The TOE generates audit logs which serve as verification of the erasure events that the administrator has performed. The Log Storage component of the TOE is responsible for the erasure of media. The Log Storage component supports several formats:

- **Regular** – a plain-text synopsis (free-form) of what activities were attempted and their result
- **Comma Separated Values (CSV)** – a plain-text file, delimited by commas, of what activities were attempted and their result in a tabular format

- **XML –** an XML file that contains both the activities that were attempted and their result as well as a brief system inventory harvested via invoking the lshw Linux utility
- **PDF –** a PDF file that contains both the activities that were attempted and their result; can optionally include a brief system inventory
- **HTML –** an HTML file that contains both the activities that were attempted and their result; can optionally include a brief system inventory
- **SQL** – a SQL query that inserts log data into a target database
- **Bootable Report –** a report that is written directly to the target drive in a way that when a system attempts to boot from the drive, the report is displayed

The TSF writes audit records in a format suitable for a TOE user to view and print the individual Disk Log Files that the TOE records from any other operating system. The TOE provides these in a user-readable format outside of the TOE but is not responsible for facilitating the viewing of those audit records.

9.1.1 Log Files

For verification of the wipe, the software allows logging to a USB drive. The log file will contain the type of wipe, the size of the hard drive and the date and time of the wipe.

Event logs of wiping sessions can be created and saved when using the program to wipe drives. The log file can be turned on/off and configured from the main WipeDrive menu via one of the available user interfaces. When fully configured, WipeDrive has the ability to log several details, including:

1. WipeDrive version number
2. Drive model information
3. Serial number of physical drive
4. Current User (user-defined, name of person performing the wipe)
5. Computer ID (user-defined, name of the drive or system)
6. Type of operation performed
7. Number of overwrites performed
8. Date & Time operation was completed
9. Total elapsed time (HH:MM:SS)
10. Operation Result (Success or Failure)
11. Total number of disk sector read/write errors, if any
12. Total uncleaned or unreadable disk sectors, if any
13. Number of Secure Erase passes, if any
14. Number of Sanitize Device passes, if any

15. NIST Method type (e.g. “Clear”, “Purge”)
16. Drive type (e.g. Platter, SSD)

The log can be found on the media the administrator has chosen to save the data on with a filename configured by the user. If an explicit filename format isn’t chosen by the user, a default name of “log” will be used. The program can be configured to create a log file for each drive that is wiped. Optionally, the logs from a single run of the program can be put into a single file.

9.1.2 Disk Errors

If the TOE reports errors during a wipe or verification, the application has encountered some issue reading or writing to the drive. This means the drive is beginning to fail. If errors are encountered, it is recommended that the computer be rebooted and the wipe process started again. If errors persist and the drive currently in use is used in the future, it is recommended that important data is backed up immediately and frequently after the initial backup. The drive could fail further or completely at any time.

9.2 Disk Erasure

The TOE erases data present by overwriting it with a particular pattern of data, thus eradicating the previous contents of the disk.

9.2.1 Patterns of Wipe Level Definitions

Each wipe pattern adheres to a specific approved standard, including official government and military standards in use today. Specific patterns such as all „ones‟ and all „zeros‟ are used in various wipe standards as defined below. The implementation of the wipe functions utilized is vendor-asserted. Some wipes are designed to use random data, or to include full verification of each character written. The following list details the 20 types of disk sanitization methods made available to administrators of the TOE:

- Standard Overwrite
    - A 1-pass overwriting algorithm that overwrites all data with a fixed value (0x00). If a firmware-based erase is supported by the drive (like Sanitize Device or Secure Erase), then the pattern will use one of those commands instead of overwriting with a fixed value.
- US DoD 5220.22-M 3-pass
    - A 3-pass overwriting algorithm where the first pass overwrites with zeroes (or a firmware-based erase is done, if supported by the drive), the next pass with ones, and the last writing pass with random bytes. A verify is then done.

- US DoD 5220.22-M 7-pass
  - A 7-pass overwriting algorithm where the first pass overwrites with zeroes (or, if supported, a firmware-based erase), the second pass overwrites with all ones, the third pass overwrites with pseudo-random data, the fourth pass overwrites with 0x97, the fifth pass overwrites with 0xC8, the sixth pass with 0x37, and the final pass overwrites with pseudo-random data.
- GB HMG Infosec Standard #5 Baseline
  - A 1-pass overwriting algorithm where data is overwritten using zeroes, and then verified.
- GB HMG Infosec Standard #5 Enhanced
  - A 3-pass overwriting algorithm where the first pass uses zeroes (or a firmware-based erase), the second uses ones, and the last pass uses pseudo-random bytes. The final pass of pseudo-random bytes is verified.
- Canadian RCMP TSSIT OPS-II Standard Wipe
  - A 7-pass overwriting algorithm featuring three alternating passes of zeroes (or a firmware-based erase) and ones, with the last pass using pseudo-random characters. The last pass is verified.
- US Army AR380-19
  - A 3-pass overwriting algorithm where the first pass is pseudo-random characters, the second pass is user defined, and the third pass is the inverse of that user definition.
- US Air Force System Security Instruction 5020
  - A 3-pass algorithm that first overwrites the target data with zeros (or does a firmware-based erase, if supported), then does another overwrite with all ones, and finally overwrite with a user-defined character.
- German VSITR
  - A 7-pass algorithm. First write all ones, second write zeros (and verifies), then ones, zeros (or firmware-based erase), then ones, then zeros (or firmware-based erase), then writes 0xAA.
- US Navy Staff Office Publication P-5329-26
  - A 3-pass overwriting algorithm where the first pass overwrites with zeroes, the next pass with ones, and the last pass with random bytes. Verify the final pass.
- US National Computer Security Center TG-025

- An overwriting algorithm which performs 3 overwrites where the first pass overwrites with zeroes, the next pass with ones, and the last pass with random bytes.

- CIS GOST P50739-95 version 2
  - A 1-pass algorithm which overwrites with pseudo-random characters.
- Australian Defense Signals Directorate ACSI-33 (X0-PD)
  - A 3-pass algorithm. Write with zeros and verify, write with all ones and verify, write with pseudo-random data.
- SecureErase + 1 overwrite with verify or NNSA NAP 14.1-C
  - This algorithm changes depending on whether the drive supports a firmware-based erase. If the drive supports a firmware-based erase, it will do a 2-pass algorithm: the firmware-based erase, and then all ones with verification. If the drive doesn't support a firmware-based erase (or if the firmware-based erase fails), then do a 3-pass algorithm: two passes of pseudo-random characters, followed by a pass of all ones. The final pass is verified.
- Canadian CSEC ITSG-06
  - A 3-pass overwriting algorithm. The first pass is all zeros (or a firmware-based erase), the second pass is all ones, and the final pass is writing pseudo-random characters. The last pass is verified.
- US Air Force System Security Instruction 8580
  - An 18-pass algorithm. It repeats the following sequence six times: first pass is zeros (or a firmware-based erase if supported), second pass is 0xAC, and the third pass is all ones. At the end, the 18th pass is verified.
- BSI-2011-VS
  - A 2-pass overwrite: first pass is overwriting with the BSI pattern (then verified), and the second pass is overwriting with zeros and then doing a quick verification (10%) of that pass.
- SSD Smart wipe
  - A proprietary wipe sequence designed specifically for SSDs. It involves a 3-pass overwrite: first pass is all zeros (or, if supported, a firmware-based erase), second pass is a special proprietary overwrite with random data, and the third pass writes all zeros.
- NIST 800-88r1
  - Attempts to achieve the "Purge" level of erasure on a drive by following the guidance outlined in the NIST SP 800-88 Rev1 document. This may include using

firmware-based commands (like Sanitize Device or Secure Erase functionality), as well as writing data to the drive. If a drive doesn't support the proper firmware-based commands, then this pattern will attempt to achieve the "Clear" level of erasure by writing over the drive.

- Custom overwrite pattern
  - A user can create their own wiping sequence.

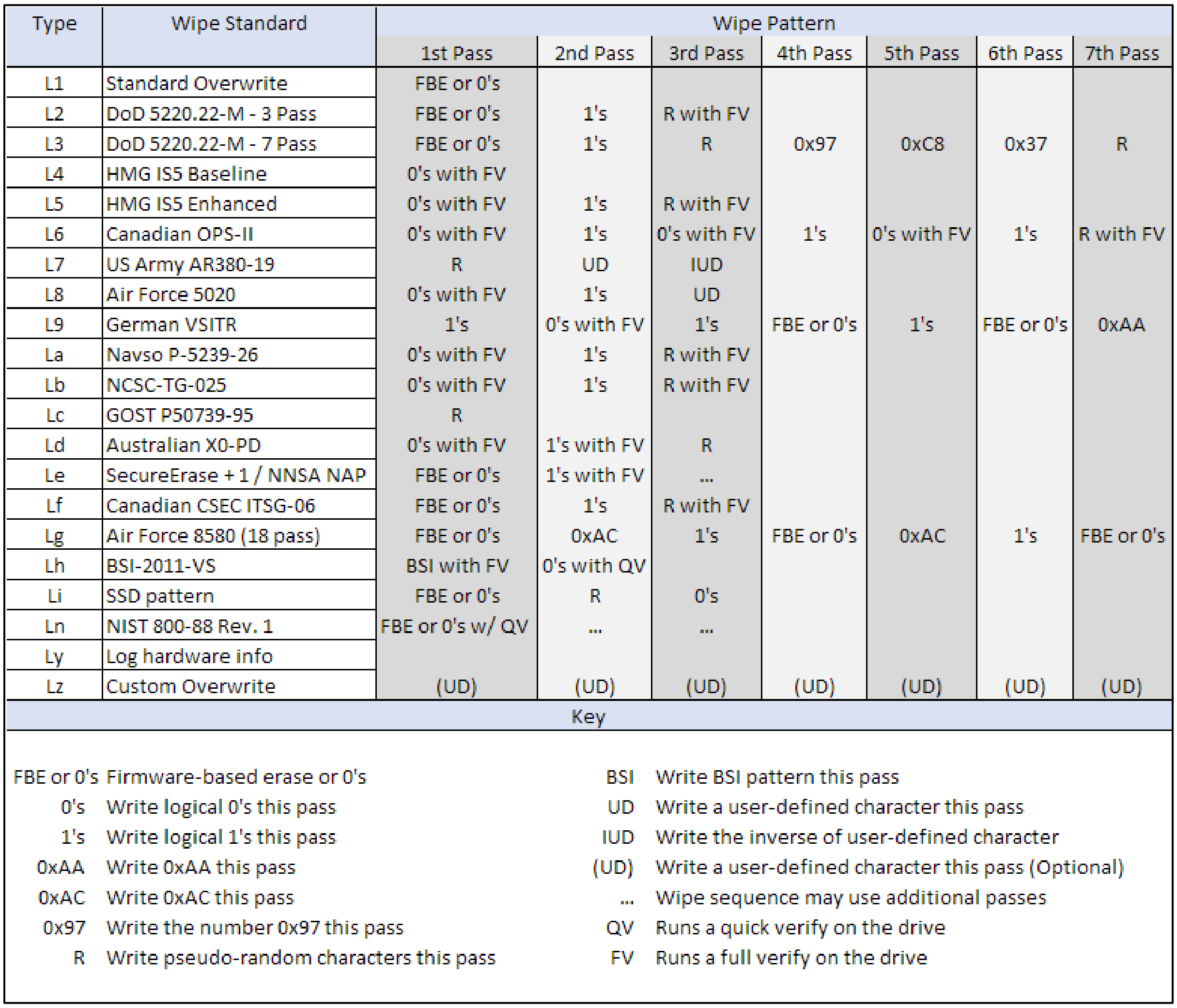
Figure 9-1 details each wipe method and its associated wipe pattern.

In order to perform a wipe of a hard drive, the following steps must be performed:

1. In order to wipe a target, a wipe pattern must be selected from an administrator defined list. A wipe pattern consists of disk operations. The following disk operations are supported:
   - Performed prior to methods mentioned in Figure 9-1:
     - ATA REMOVE HPA
     - ATA REMOVE DCO
     - ATA REMOVE ACCESSIBLE MAX ADDRESS

   Performed in conjunction with the methods mentioned in Figure 9-1:
     - Write value
     - Verify value
     - Write random
     - Verify random
     - Firmware-based commands (e.g. Enhanced Secure Erase, Sanitize Device)

   Alternatively, a wipe pattern can be defined by the user using an arbitrary collection of operations and patterns. These wipe patterns determine the specific method that is used to wipe data from the target.
2. Once the wipe pattern has been defined, a list of one or more target block devices on the system must be specified.
3. The wipe pattern sequence is executed against each target device in order.
4. During the execution of the wipe pattern sequence, progress is displayed to the UI.
5. Whenever a process is completed, a response is sent out, and the log data is received by any process listening for it. These processes comprise the preconfigured log source(s): one or more of standard output, UNIX pipe, or network socket.



Type	Wipe Standard	Wipe Pattern						
		1st Pass	2nd Pass	3rd Pass	4th Pass	5th Pass	6th Pass	7th Pass
L1	Standard Overwrite	FBE or 0's						
L2	DoD 5220.22-M - 3 Pass	FBE or 0's	1's	R with FV				
L3	DoD 5220.22-M - 7 Pass	FBE or 0's	1's	R	0x97	0xC8	0x37	R
L4	HMG IS5 Baseline	0's with FV						
L5	HMG IS5 Enhanced	0's with FV	1's	R with FV				
L6	Canadian OPS-II	0's with FV	1's	0's with FV	1's	0's with FV	1's	R with FV
L7	US Army AR380-19	R	UD	IUD				
L8	Air Force 5020	0's with FV	1's	UD				
L9	German VSITR	1's	0's with FV	1's	FBE or 0's	1's	FBE or 0's	0xAA
La	Navso P-5239-26	0's with FV	1's	R with FV				
Lb	NCSC-TG-025	0's with FV	1's	R with FV				
Lc	GOST P50739-95	R						
Ld	Australian X0-PD	0's with FV	1's with FV	R				
Le	SecureErase + 1 / NNSA NAP	FBE or 0's	1's with FV	...				
Lf	Canadian CSEC ITSG-06	FBE or 0's	1's	R with FV				
Lg	Air Force 8580 (18 pass)	FBE or 0's	0xAC	1's	FBE or 0's	0xAC	1's	FBE or 0's
Lh	BSI-2011-VS	BSI with FV	0's with QV					
Li	SSD pattern	FBE or 0's	R	0's				
Ln	NIST 800-88 Rev. 1	FBE or 0's w/ QV	...	...				
Ly	Log hardware info							
Lz	Custom Overwrite	(UD)	(UD)	(UD)	(UD)	(UD)	(UD)	(UD)

Key

FBE or 0's	Firmware-based erase or 0's	BSI	Write BSI pattern this pass
0's	Write logical 0's this pass	UD	Write a user-defined character this pass
1's	Write logical 1's this pass	IUD	Write the inverse of user-defined character
0xAA	Write 0xAA this pass	(UD)	Write a user-defined character this pass (Optional)
0xAC	Write 0xAC this pass	...	Wipe sequence may use additional passes
0x97	Write the number 0x97 this pass	QV	Runs a quick verify on the drive
R	Write pseudo-random characters this pass	FV	Runs a full verify on the drive

Figure 9-1: Wipe Patterns

9.3 Security Management

9.3.1 User-Accessible Interfaces

The user interfaces serve as the physical interface where controls are used to operate one or more instances of the WipeDrive application, each on a distinct host. The interfaces that are included in the evaluated configuration are:

- **GUI** – A graphical UI, ran on the same host as the WipeDrive Application. This will be the default interface for x86 machines which have access to framebuffer

9.3.2 Administrator Capabilities

Administrators of the TOE have the ability to perform the following operations:

- Specify location for audit storage
- Specify format for log storage
- Ability to view sector data

- Create an administrator definable wipe pattern
- Delete an administrator definable wipe pattern
- Run an administrator definable wipe pattern
- View all administrator definable wipe patterns
- Scan for devices
- Get device info for all devices previously scanned
- Configure licensing options (e.g. Cloud code account, security dongle)

If an individual has physical possession of the application, they are then considered to be an administrator of the TOE.

9.3.3 WipeDrive Operations

TOE users have the ability to perform three distinct operations when using the TOE – scanning a drive, probing a drive, and performing the erasure.

9.3.3.1 Drive Scanning

The steps necessary to scan a drive that is a candidate for erasure are listed below:

1. At boot, the OS will run a shell script to launch the UI and the backend (wipedrive).
2. The UI is loaded with parameters from an initial configuration file
3. wipedrive is started.
4. Once loaded, wipedrive executes a series of commands reserved for its startup sequence as defined in the configuration file, including drive scanning.
5. wipedrive I/O loop receives the instruction to scan the system for all applicable block devices
6. I/O loop passes this command to the operating system and third-party command line programs to determine what drives are available.
7. wipedrive also opens the /sys/block directory and gathers data from it. This data was originally created by the Linux kernel.
8. This data is cached in wipedrive for viewing in the UI and used as input for probing.

9.3.3.2 Drive Probing

The steps necessary to probe a drive that is a candidate for erasure are listed below:

1. For each item in /sys/block, the following sequence is performed:
2. First, it attempts to instantiate an NVMe device object, and gather data for a drive of that type. If this fails, the drive cannot be accessed through the NVMe interface.

3. Next, it attempts to instantiate an ATA device object whose constructor performs an ATA IDENTIFY DEVICE command. This command attempts to gather the 512b block of data which defines the drive information and is characteristic of ATA devices. If this fails, it cannot be accessed through the ATA command set.

4. Next, it attempts to instantiate a SCSI device object whose constructor performs a SCSI inquiry. This gathers the following data:
   a. Drive model information
      i. Manufacturer
      ii. Model name
      iii. Serial Number
      iv. Drive Capacity

5. If this fails, the process terminates.

6. Once basic data about the device is gathered, additional SCSI inquiries are run to determine additional information about the device.

7. When the process completes for each device, the data is cached for use in the GUI.

9.3.3.3 Drive Erasure

The steps necessary to wipe a drive that is a candidate for erasure are listed below:

2. In order to wipe a target, a wipe pattern must be selected from a pre-defined list. A wipe pattern consists of disk operations. The following disk operations are supported:
   - Performed prior to methods mentioned in Figure 9-1:
     - ATA REMOVE HPA
     - ATA REMOVE DCO
     - ATA REMOVE ACCESSIBLE MAX ADDRESS
   - Performed in conjunction with the methods mentioned in Figure 9-1:
     - Write value
     - Verify value
     - Write random
     - Verify random
     - Firmware-based commands (e.g. Enhanced Secure Erase, Sanitize Device)
   - Alternatively, a wipe pattern can be defined by the administrator using an arbitrary collection of operations, sequences,

and/or patterns. These wipe patterns determine the specific method that is used to wipe data from the target. [gather the operations which comprise specific patterns]

3. Once the wipe pattern has been defined, a list of one or more target block devices on the system must be specified.
4. The administrator definable wipe pattern inserts operations as necessary to identify a valid license is present and decrement the number of licenses remaining.
5. The wipe pattern sequence is executed against each target device in order.
6. During the execution of the wipe pattern sequence, progress is displayed to the UI.
7. Once all drives are wiped, if audit logging was configured, put the logging data in the appropriate log.

9.3.3.4 Booting from standalone ISO

The TOE can be run from the included Linux LiveCD for wiping an environment booted to a target environment. As a caveat, this may require the boot order to be changed in the BIOS (refer to the physical boundary section for more information on this).

9.4 User Data Protection

The TOE provides for the erasure of residual information. This erasure is initiated at the user-facing interfaces and requires communication with the information repository (disk). The erasure of residual information is performed when a deviceOp instance is executed – which is a direct result of the administrator definable wipe pattern. No residual information will reside in the RAM subsequent to a wipe event.

9.5 Self-Protection (ADV_ARC.1)

Domain separation is the security architecture property whereby the TSF defines separate security domains for administrators and for the TSF; it ensures that no user process can affect the contents of a security domain of another administrator or of the TSF.

The TSF is designed in such a manner that requires administrators to have physical possession of the WipeDrive application before any TSF-mediated operations can occur. Therefore, an individual's authorization is based on the possession of the Linux LiveCD containing the WipeDrive application. With the Linux LiveCD, the administrator is able to perform management functions through either the Local GUI or Console UI. There are no specific access control features or authentication methods in place as access is granted to the individual with possession of the Linux LiveCD.

All requests for protected resources, i.e. data located on drives on a target, are processed through the User Interface, either Local GUI or Console UI. When an erasure event is performed on a target, the verification data that denotes whether the erasure was a success or failure is stored in the cache located on the WipeDrive application. This data is then securely transmitted between TOE components to both the administrators interface and is placed in the Log Storage component. This allows the administrator to receive a graphical representation of the status of the erasure performed.

Administrators do have the ability to perform management functions remotely; however, the user interfaces related to remote access are not included in this evaluated configuration.

10 TOE Summary Specification Rationale

This section identifies the security functions provided by the TOE mapped to the security functional requirement components contained in this ST. This mapping is provided in the following table.

Security Function	Security Functional Components
SF.Security Audit	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
	FAU_SAR.1 Audit Review
SF.Security Management	FMT_SMF.1 Specification of Management Functions
SF.User Data Protection	FDP.RIP.1 Residual Information Protection
SF.Disk Erasure	FDE_SCN_EXT.1 Scan of Devices
	FDE_PRB_EXT.1 Probe of Devices
	FDE_ERS_EXT.1 Erasure of Devices

Table 10-1: Security Functional Requirements

10.1.1 Security Audit

The security audit function of the TOE enforces the FAU_GEN.1, FAU_GEN.2, and FAU_SAR.1 requirements. FAU_GEN.1 requires a reliable timestamp, which is provided by the Operating System that is bundled on the LiveCD.

By default, audit data is created by scanning, probing, and/or wiping of a target on a device. This data produces a verification message of the success or failure of this event; this notification is sent to the console, the user who performed the event, as well as it being stored in the Log Storage component. Along with the success or failure of events being recorded, the TSF records the date and time of that event, the type of event (i.e. erasure, probe, scan), the identity of the user performing the event, the serial number of the target drive, the wipe pattern sequence used to erase the drive, number of overwrites performed, elapsed time of the operation, physical blocks of target, and the logical blocks of the target. All audit operations listed above can optionally be linked to a user who caused the event.

In the evaluated configuration, this audit data contained in the Log Storage component could be stored on any media device, e.g. thumb drive or remotely on an FTP server or Windows share. When the TOE is configured to log audit data to an FTP server or SQL database (MySQL or MS SQL), the logs and all associated data are sent as clear-text.

The TSF will ensure that this information is logged in a clear and coherent manner so that the reader is able to accurately interpret the data. The TOE saves the audit events in a user-readable format outside of the TOE but is not responsible for facilitating the viewing of those audit records. Note that the Linux syslog audit functionality described under FAU_GEN.1.1 cannot be disabled, and runs automatically.

Users are self-identified when performing operations, therefore the TOE does not ensure correct authentication. As a result, self-identification does not necessitate having the FIA_UID.1 requirement.

10.1.2 Security Management

The management function of the TOE enforces the FMT_SMF.1 requirement.

The TOE provides management capabilities that only administrators can perform. The management functions available to these users can be initiated by the Local GUI.

10.1.3 User Data Protection

The user data protection function of the TOE enforces the FDP_RIP.1 requirement.

Subsequent to an erasure event on a targeted device, the TSF ensures that any previous information is made unavailable based upon the allocation of resources to any object created as a result of an administrator definable wipe pattern. This protects any information that may have remained after an erasure event.

10.1.4 Disk Erasure

The disk erasure function of the TOE enforces the FDE_SCN_EXT.1, FDE_PRB_EXT.1, and FDE_ERS_EXT.1 requirements.

The TSF is able to scan a system for (target(s) on (a)) devices that are eligible for erasure. More specifically, the Linux kernel recognizes all ATA, SCSI, USB, eMMC, SD, and NVMe-block devices on the given system as potential erasure targets. Probing is allowing the TSF to communicate with devices that are discovered as the result of a scan; this is done in order to determine the parameters for the device. Both scanning and probing of targets are performed during the initialization process of the TOE.

The TSF has the ability, once devices have been targeted through the probing and scanning process, to erase ATA, SCSI, USB, eMMC, SD, and NVMe-block devices by using any of the following 20 available wipe functions: Standard Overwrite, US DoD 5220.22-M 3-pass, US DoD 5220.22-M 7-pass, GB HMG Infosec Standard #5 Baseline, GB HMG Infosec Standard #5 Enhanced, Canadian RCMP TSSIT OPS-II Standard Wipe, US Army AR380-19, US Air Force System Security Instruction 5020, German VSITR, US Navy Staff Office Publication P-5329-26, US National Computer Security Center TG-025, CIS GOST P50739-95 version 2, Australian Defense Signals Directorate ACSI-33 (X0-PD), SecureErase + 1 overwrite with verify or NNSA NAP 14.1-C, CSEC ITSG-06, US Air Force System Security Instruction 8580, BSI-2011-VS, SSD Smart wipe, NIST 800-88r1, and Custom overwrite pattern.

11 Security Problem Definition Rationale

11.1 Security Objectives Rationale

The following table provides a mapping with rationale to identify the security objectives that address the stated assumptions and threats.

Assumption	Objective	Rationale
A. ADMIN There will be one or more authorized administrators assigned to install, configure, and manage the TOE and the security information it contains.	OE.ADMIN One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains.	OE.ADMIN maps to A. ADMIN in order to ensure that authorized administrators install, manage and operate the TOE in a manner that maintains its security objectives.
A.PATCHES System Administrators exercise due diligence to acquire updated versions of the TOE and patch the Operational environment (e.g. OS and database) so they are not susceptible to network attacks.	OE.PATCHES Administrators of the Operational Environment exercise due diligence to acquire updated versions of the TOE and patch the Operational Environment (e.g., OS and database) so they are not susceptible to attacks resulting in malfunction of the TOE or associated audit data.	OE.PATCHES maps to A. PATCHES in order to ensure that the authorized administrators properly patch the Operational environment in a manner that maintains its security objectives.
A.NOEVIL All users of the TOE are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.	OE.NOEVIL All users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the guidance documentation.	OE.NOEVIL directly maps to A.NOEVIL and ensures that all users of the TOE are properly trained in the configuration and usage of the TOE and will follow the guidance provided.
A.LOCATE The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	OE.LOCATE The TOE will be located within controlled access facilities that will prevent unauthorized physical access.	OE.LOCATE directly maps to A.LOCATE to ensure that those responsible for the TOE locate the TOE in a controlled access facility that will prevent unauthorized physical access.

A.LOCAL The TOE will be loaded onto the same physical machine as the target resource so that commands are not exposed over the network.	OE.LOCAL The TOE will be loaded onto the same physical machine as the target resource so that commands are not exposed over the network.	OE.LOCAL directly maps to A.LOCAL to ensure that the TOE will be loaded onto the same physical machine as the target resource so that commands are not exposed over the network.

Table 11-1: Assumption to Objective Mapping

Threat/Policy	**Objective**	**Rationale**
T.ADMIN_ERROR An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.	O.ROBUST_ADMIN_GUIDANCE The vendor will provide administrators with the necessary information for secure delivery and management of the TOE.	O.ROBUST_ADMIN_GUIDANCE (AGD_OPE.1, AGD_PRE.1, ALC_DEL.1) mitigates the risk of an administrator incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms by providing administrators with the necessary information for secure delivery and management of the TOE.
T.AUDIT_COMPROMISE A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.	O.AUDIT (FAU_GEN.1, FAU_GEN.2, and FAU_SAR.1) addresses T.AUDIT_COMPROMISE by providing the necessary measures to be put in place for recording security relevant events that will only assist authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE.
T.AUDIT_FAILURE A malicious user or process failure may cause the TOE to fail to record or improperly record audit data, thus masking a user's action.	O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.	O.AUDIT (FAU_GEN.1, FAU_GEN.2, FAU_SAR.1) addresses T.AUDIT_FAILURE by providing the necessary measures to be put in place for recording security relevant events that will assist authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE.

	OE.NOEVIL Users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.	OE.NOEVIL directly maps to T.AUDIT_FAILURE and ensures users of the TOE are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the organization's guidance documentation.
	OE.SYSTIME The operating environment will provide reliable system time.	OE.SYSTIME is necessary for correct timestamps to be included in the audit logs (see FAU_GEN.1). Without reliable time/date information, the audit log entries would not contain correct information about when logs were created or when logs failed to be created.
T.RESIDUAL Any person with access to a target environmental resource can access residual data, either due to a wipe operation being incomplete or a completed wipe operation being insufficient.	O.ERASE The TOE will provide measures for erasing data contained on block devices on a target system as well as sufficient assurance that the desired data was erased, and that the erasure method was sufficient for permanent erasure. The TOE must erase the data in such a way that the block devices are reusable (i.e. the device is not destroyed) and in conformance with the standards of the wipe pattern selected by the administrator.	O.ERASE (FDE_PRB_EXT.1, FDE_SCN_EXT.1, FDE_ERS_EXT.1, FDP_RIP.1) mitigates this risk by ensuring that once an erasure event has occurred, that no residual information should remain on the target being wiped.
T.UNAUTH An unauthorized user obtains the physical medium which contains the TOE and uses it to perform a wipe operation against an environmental resource which there has been no authorization to wipe.	O.MANAGE The TOE will provide authorized administrators with the resources to manage and monitor the set of disk wipe patterns made available to the TOE and the storage of audit data generated by the TOE.	O.MANAGE (FMT_SMF.1, FAU_SAR.1) mitigates the risk of unauthorized users being able to access confidential data because there are specific TSF-mediated actions that only authorized users will be able to perform. A cloud code is required in order to perform wipe operations, and as long as that code is secret an unauthorized user wouldn't be able to do an unauthorized wipe. It also mitigates the threat that unauthorized users modify the audit records by including a security code that can be used to help detect modifications made manually. Only authorized users will have access to the TOE. It

		mitigates the threat that unauthorized users would be able to read data contained in the audit records by limiting such management functions to administrators of the TOE.
P.REUSE All drive data must be securely erased to allow the reuse of drives.	O.ERASE The TOE will provide measures for erasing data contained on block devices on a target system as well as sufficient assurance that the desired data was erased, and that the erasure method was sufficient for permanent erasure. The TOE must erase the data in such a way that the block devices are reusable (i.e. the device is not destroyed) and in conformance with the standards of the wipe pattern selected by the administrator.	O.ERASE (FDE_ERS_EXT.1, FDP_RIP.1) ensures that this policy is met by ensuring that erasure is done in a way that the device can be reused without any risk of previous data being on the device.
P.STANDARD The TOE will be used to securely erase drive data in conformance with the standards of the organization.	O.ERASE The TOE will provide measures for erasing data contained on block devices on a target system as well as sufficient assurance that the desired data was erased, and that the erasure method was sufficient for permanent erasure. The TOE must erase the data in such a way that the block devices are reusable (i.e. the device is not destroyed) and in conformance with the standards of the wipe pattern selected by the administrator.	O.ERASE (FDE_ERS_EXT.1) ensures that this policy is met by making sure that the wipe patterns used to erase the drive are in compliance with the standards set by the organization.

Table 11-2: Threat/Policy to Objective Mapping

11.2 EAL 2 Justification

The threats that were chosen are consistent with attacker of medium attack potential, therefore EAL2 was chosen for this ST.

11.3 Requirement Dependency Rationale

All Security Functional Requirement component dependencies have been met by the TOE as defined by the CEM with the exception of FPT_STM.1 and FAU_GEN.1.
Rationale for these exclusions is included in Section 10.1.1 above.

11.4 Security Functional Requirements Rationale

The following table provides a mapping with rationale to identify the security functional requirement components that address the stated TOE and environment objectives.

Objective	Security Functional Component	Rationale
O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	AGD_OPE.1 Operational User Guidance	AGD_OPE.1 describes the proper use of the TOE from a user standpoint.
	AGD_PRE.1 Preparative Procedures	AGD_PRE.1 documents the procedures necessary and describes the steps required for the secure installation, generation, and start-up of the TOE.
	ALC_DEL.1 Delivery Procedures	ALC_DEL.1 describes product delivery and a description of all procedures used to ensure objectives are not compromised in the delivery process.
O.MANAGE The vendor will provide administrators with the necessary information for secure delivery and management of the TOE.	FMT_SMF.1 Specification of Management Functions	FMT_SMF.1 states that the administrator of the TOE will be able to perform various management functions.
O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.	FAU_GEN.1 Audit Data Generation	FAU_GEN.1 states that the TSF shall be able to generate an audit record of the start-up and shutdown of the audit functions.
	FAU_GEN.2 User Identity Association	FAU_GEN.2 ensures that the audit records associate a user identity with the auditable event.
	FAU_SAR.1 Audit Review	FAU_SAR.1 ensures that only authorized users of the TOE will be able to read the TOE's audit records. Additionally, FAU_SAR.1 ensures that this information will be presented to the user in a format that is coherent and easily understandable.
O.ERASE The TOE will provide measures for erasing data contained on block devices on a target system as well as sufficient	FDE_SCN_EXT.1 Scan of Devices	FDE_SCN_EXT.1 states that the TSF shall be able to scan a system for devices that are erasure targets.

assurance that the desired data was erased, and that the erasure method was sufficient for permanent erasure. The TOE must erase the data in such a way that the block devices are reusable (i.e. the device is not destroyed) and in conformance with the standards of the wipe pattern selected by the administrator.	FDE_PRB_EXT.1 Probe of Devices	FDE_PRB_EXT.1 states that the TSF shall communicate with devices that are discovered as the result of the scan in order to determine the parameters for the device.
	FDE_ERS_EXT.1 Erasure of Devices	FDE_ERS_EXT.1 states that the TSF shall be able to erase devices that are discovered by a scan using any combination of the 20 available wipe functions.
	FDP_RIP.1 Residual Information Protection	FDP_RIP.1 states that the TSF shall ensure that any previous information content of a resource is made unavailable upon the allocation of the resource to any object created as the result of an administrator definable wipe pattern.
	FAU_GEN.1 Audit Data Generation	FAU_GEN.1 states that the TSF shall be able to generate an audit record of the start-up and shutdown of the audit functions
	FAU_GEN.2 User Identity Association	FAU_GEN.2 ensures that the audit records associate a user identity with the auditable event.

Table 11-3**: Security Functional Requirements Rationale**

11.5 Assurance Measures

This section identifies the assurance measures provided by the developer in order to meet the security assurance requirement components for EAL2 augmented with ASE_TSS.2 and ALC_FLR.2. A description of each of the TOE assurance measures follows in Table 11-4.

Component	**Document(s)**	**Rationale**
ADV_ARC.1 Security Architecture Design	WipeDrive Version 9.1 – Security Design	This document describes the security architecture of the TOE.
ADV_FSP.2 Functional Specification with complete summary	• WipeDrive Version 9.1 - Functional Specification • WipeDrive Version 9.1 - Security Design	This document describes the functional specification of the TOE with complete summary.

ADV_TDS.1 Architectural Design	- WipeDrive Version 9.1 - Functional Specification	This document describes the architectural design of the TOE.
AGD_OPE.1 Operational User Guidance	WipeDrive Version 9.1 Enterprise User Guide sample-wd-options.cfg	This document describes the operational user guidance for the TOE.
AGD_PRE.1 Preparative Procedures	WipeDrive Version 9.1 Enterprise User Guide	This document describes the preparative procedures that need to be done prior to installing.

Component	**Document(s)**	**Rationale**
ALC_CMC.2 Authorizations Controls	WhiteCanyonSourceControl Management.git	This document describes the authorization controls for the TOE.
ALC_CMS.2 CM Scope	• Gitlab_screenshot • ConfigurationManagementParts	These documents describe the CM scope of the TOE.
ALC_DEL.1 Delivery Procedures	WhiteCanyonProductDeliveryDocumentation.docx Version 1.1	This document describes product delivery for and a description of all procedures used to ensure objectives are not compromised in the delivery process.
ALC_FLR.2 Flaw reporting procedures	WhiteCanyonFlawRemediationProcess.docx Version 1.1	This document provides the policies for issuing new releases of the TOE as corrective actions.
ASE_CCL.1 Conformance Claims	WipeDrive Version 9.1 Security Target (this document)	This document describes the CC conformance claims made by the TOE.
ASE_ECD.1 Extended Components Definition	WipeDrive Version 9.1 Security Target (this document)	This document provides a definition for all extended components in the TOE.
ASE_INT.1 Security Target Introduction	WipeDrive Version 9.1 Security Target (this document)	This document describes the Introduction of the Security Target.

ASE_OBJ.2 Security Objectives	WipeDrive Version 9.1 Security Target (this document)	This document describes all of the security objectives for the TOE.
ASE_REQ.2 Security Requirements	WipeDrive Version 9.1 Security Target (this document)	This document describes all of the security requirements for the TOE.
ASE_SPD.1 Security Problem Definition	WipeDrive Version 9.1 Security Target (this document)	This document describes the security problem definition of the Security Target.
ASE_TSS.2 TOE Summary Specification with architectural design summary	WipeDrive Version 9.1 Security Target (this document)	This document describes the TSS section of the Security Target.
Component	**Document(s)**	**Rationale**
ATE_COV.1 Analysis of Coverage	• Wipedrive Test Case Spreadsheet.xlsx • wd-9.1-enterprise-dongle-test-results.xlsx	This document provides an analysis of coverage for the TOE.
ATE_FUN.1 Functional Tests	• Wipedrive Test Case Spreadsheet.xlsx • wd-9.1-enterprise-dongle-test-results.xlsx	This document describes the functional tests for the TOE.
ATE_IND.2 Independent Testing	• CCLabIndependentTestResults (provided by the evaluation laboratory) • WhiteCanyon WipeDrive Version 9.1 Evaluation Team Test Report Version 1.0	This document describes the independent testing for the TOE.
AVA_VAN.2 Vulnerability Analysis	• Network-Activity-wd-9.1-enterprise-dongle.xlsx • Vulnerability Analysis WHITECANYON, INC. WIPEDRIVE VERSION 9.1 Version 1.0 (provided by the evaluation laboratory)	This document describes the vulnerability analysis of the TOE.

Table 11-4: Assurance Requirements Evidence

11.6 Extended Requirements Rationale

This TOE contains the following extended security functions:
FDE_SCN_EXT.1

FDE_PRB_EXT.1
FDE_ERS_EXT.1

11.6.1 FDE_SCN

FDE_SCN_EXT.1 was created to capture the basic functionality provided by the TOE. FDE_SCN_EXT.1 allows for the TOE to be able to scan a given system for devices, e.g. hard drives, partitions, that are targets for erasure. Through this requirement, the WipeDrive application is capable via the Linux kernel of recognizing all block devices located on a system as potential erasure targets. Scanning is performed automatically upon initialization of the WipeDrive application.

11.6.2 FDE_PRB

FDE_PRB_EXT.1 was created to capture the basic functionality provided by the TOE. FDE_PRB_EXT.1 allows for the TOE to communicate with devices that are discovered as a result of the scan in order to determine the parameters for the given device. Probing, along with scanning, is performed automatically upon initialization of the WipeDrive application.

11.6.3 FDE_ERS

FDE_ERS_EXT.1 was created to capture the basic functionality provide by the TOE. FDE_ERS_EXT.1 allows the administrators of the TOE to select from 20 different wipe patterns in order to erase devices (in a way so that it can be re-used) discovered through the scanning process.