

# High Sec Labs

## FH10N-4

Firmware Version 40404-0E7

## Security Target

*Doc No: 2149-001-D102A6B*

*Version: 1.3*

*27 February 2025*



*High Sec Labs Ltd.  
29 HaEshel St  
Caesarea,  
Israel 3079510*

### **Prepared by:**

*EWA-Canada, An Intertek Company  
1223 Michael Street North, Suite 200  
Ottawa, Ontario, Canada  
K1J 7T2*



# CONTENTS

<b>1</b>	<b>SECURITY TARGET INTRODUCTION .....</b>	<b>1</b>
1.1	DOCUMENT ORGANIZATION.....	1
1.2	SECURITY TARGET REFERENCE.....	1
1.3	TOE REFERENCE.....	2
1.4	TOE OVERVIEW.....	2
	1.4.1 TOE Environment .....	3
1.5	TOE DESCRIPTION .....	3
	1.5.1 USB Filter Evaluated Configuration .....	4
	1.5.2 Physical Scope .....	4
	1.5.3 Logical Scope.....	5
<b>2</b>	<b>CONFORMANCE CLAIMS.....</b>	<b>6</b>
2.1	COMMON CRITERIA CONFORMANCE CLAIM .....	6
2.2	PROTECTION PROFILE CONFORMANCE CLAIM .....	6
2.3	PACKAGE CLAIM.....	7
2.4	CONFORMANCE RATIONALE .....	7
<b>3</b>	<b>SECURITY PROBLEM DEFINITION .....</b>	<b>8</b>
3.1	THREATS .....	8
3.2	ORGANIZATIONAL SECURITY POLICIES .....	9
3.3	ASSUMPTIONS.....	9
<b>4</b>	<b>SECURITY OBJECTIVES.....</b>	<b>10</b>
4.1	SECURITY OBJECTIVES FOR THE TOE .....	10
4.2	SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT .....	13
4.3	SECURITY OBJECTIVES RATIONALE.....	14
<b>5</b>	<b>EXTENDED COMPONENTS DEFINITION .....</b>	<b>18</b>
<b>6</b>	<b>SECURITY REQUIREMENTS .....</b>	<b>19</b>
6.1	CONVENTIONS.....	19
6.2	SECURITY FUNCTIONAL REQUIREMENTS.....	19
	6.2.1 User Data Protection (FDP).....	20
	6.2.2 Protection of the TSF (FPT).....	23
6.3	SECURITY ASSURANCE REQUIREMENTS.....	24

6.4	SECURITY REQUIREMENTS RATIONALE.....	24
6.4.1	Security Functional Requirements Rationale.....	24
6.4.2	Dependency Rationale .....	25
6.4.3	Security Assurance Requirements Rationale.....	25
<b>7</b>	<b>TOE SUMMARY SPECIFICATION .....</b>	<b>26</b>
7.1	USER DATA PROTECTION .....	26
7.1.1	System Controller .....	26
7.1.2	Keyboard and Mouse Functionality.....	27
7.2	PROTECTION OF THE TSF .....	28
7.2.1	No Access to TOE .....	28
7.2.2	Anti-tampering Functionality .....	28
7.2.3	TSF Testing .....	29
<b>8</b>	<b>TERMINOLOGY AND ACRONYMS .....</b>	<b>30</b>
8.1	TERMINOLOGY .....	30
8.2	ACRONYMS.....	30
<b>9</b>	<b>REFERENCES.....</b>	<b>32</b>
	<b>ANNEX A – LETTER OF VOLATILITY .....</b>	<b>1</b>

## LIST OF TABLES

Table 1 – Non-TOE Hardware and Software .....	3
Table 2 – TOE Isolator Device and Features .....	4
Table 3 – Logical Scope of the TOE .....	5
Table 4 – Applicable Technical Decisions .....	7
Table 5 – Threats.....	9
Table 6 – Assumptions.....	9
Table 7 – Security Objectives for the TOE .....	13
Table 8 – Security Objectives for the Operational Environment.....	14
Table 9 – Security Objectives Rationale .....	17
Table 10 – Functional Families of Extended Components .....	18
Table 11 – Summary of Security Functional Requirements .....	20
Table 12 – Security Assurance Requirements.....	24

Table 13 – Functional Requirement Dependencies .....	25
Table 14 – Terminology .....	30
Table 15 – Acronyms .....	31
Table 16 – References .....	32

## LIST OF FIGURES

Figure 1 – Simplified Filter Diagram for a HID Filter .....	3
Figure 2 – USB Filter Evaluated Configuration .....	4

# 1 SECURITY TARGET INTRODUCTION

This Security Target (ST) defines the scope of the evaluation in terms of the assumptions made, the intended environment for the Target of Evaluation (TOE), the Information Technology (IT) security functional and assurance requirements to be met, and the level of confidence (evaluation assurance level) to which it is asserted that the TOE satisfies its IT security requirements. This document forms the baseline for the Common Criteria (CC) evaluation.

## 1.1 DOCUMENT ORGANIZATION

**Section 1, ST Introduction**, provides the Security Target reference, the Target of Evaluation reference, the TOE overview and the TOE description.

**Section 2, Conformance Claims**, describes how the ST conforms to the Common Criteria, Protection Profile (PP) and PP Module.

**Section 3, Security Problem Definition**, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.

**Section 4, Security Objectives**, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment in response to the problem defined by the security problem definition.

**Section 5, Extended Components Definition**, defines the extended components which are then detailed in Section 6.

**Section 6, Security Requirements**, specifies the security functional and assurance requirements that must be satisfied by the TOE and the IT environment.

**Section 7, TOE Summary Specification**, describes the security functions that are included in the TOE to enable it to meet the IT security functional requirements.

**Section 8 Terminology and Acronyms**, defines the acronyms and terminology used in this ST.

**Section 9 References**, provides a list of documents referenced in this ST.

## 1.2 SECURITY TARGET REFERENCE

<b>ST Title:</b>	High Sec Labs FH10N-4 Firmware Version 40404-0E7 Security Target
<b>ST Version:</b>	1.3
<b>ST Date:</b>	27 February 2025

## 1.3 TOE REFERENCE

<b>TOE Identification:</b>	High Sec Labs FH10N-4 Firmware Version 40404-0E7
<b>TOE Developer:</b>	High Sec Labs Ltd.
<b>TOE Type:</b>	Peripheral Sharing Device (Other Devices and Systems)

## 1.4 TOE OVERVIEW

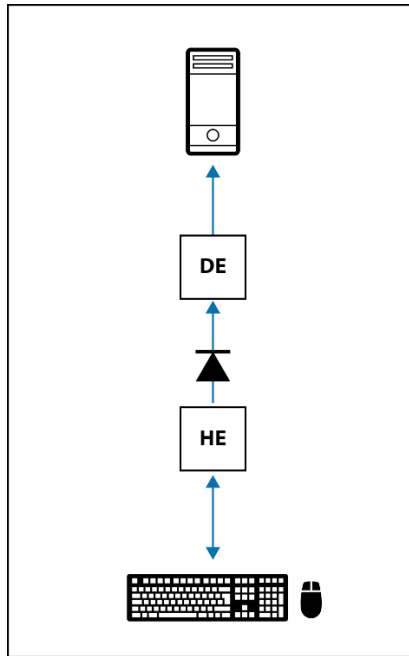
The High Sec Labs (HSL) Secure Universal Serial Bus (USB) Human Interface Device (HID) filter (FH10N-4) is connected between a computer and a USB keyboard/mouse to prevent the exploitation of data leakage.

The following security features are provided by the HSL FH10N-4 USB HID Filter:

- Keyboard and Mouse Security
  - The keyboard and mouse are isolated by dedicated, USB device emulation for each computer
  - One-way, peripheral-to-computer data flow is enforced through unidirectional optical data diodes
  - Communication from computer-to-keyboard/mouse is blocked
  - Non HID (Human Interface Device) data transactions are blocked
- Hardware Anti-Tampering
  - Special holographic tampering evident labels on the product's enclosure provide a clear visual indication if the product has been opened or compromised

The HSL USB HID filter device uses an isolated microcontroller to emulate connected peripherals in order to ensure one-way peripheral to computer data flow.

Figure 1 is a simplified block diagram showing the TOE keyboard and mouse data path for two ports. A Host Emulator (HE) communicates with the user keyboard via the USB protocol. The Host Emulator converts user keystrokes into unidirectional serial data. An isolated Device Emulator (DE) is connected to the data diode on one side and to the computer on the other side. Each key stroke or mouse action is converted by the DE into a bi-directional stream to communicate with the computer.



**Figure 1 – Simplified Filter Diagram for a HID Filter**

The TOE is a combined software and hardware TOE. A mapping showing the applicable Security Functional Requirements (SFRs) is included in Annex B.

### 1.4.1 TOE Environment

The following components are required for operation of the TOE in the evaluated configuration.

Component	Description
Connected Computer	General purpose computer
Keyboard	General purpose USB keyboard
Mouse	General purpose USB mouse
HSL KVM Cables	USB Type-A to USB Type-B (keyboard and mouse)

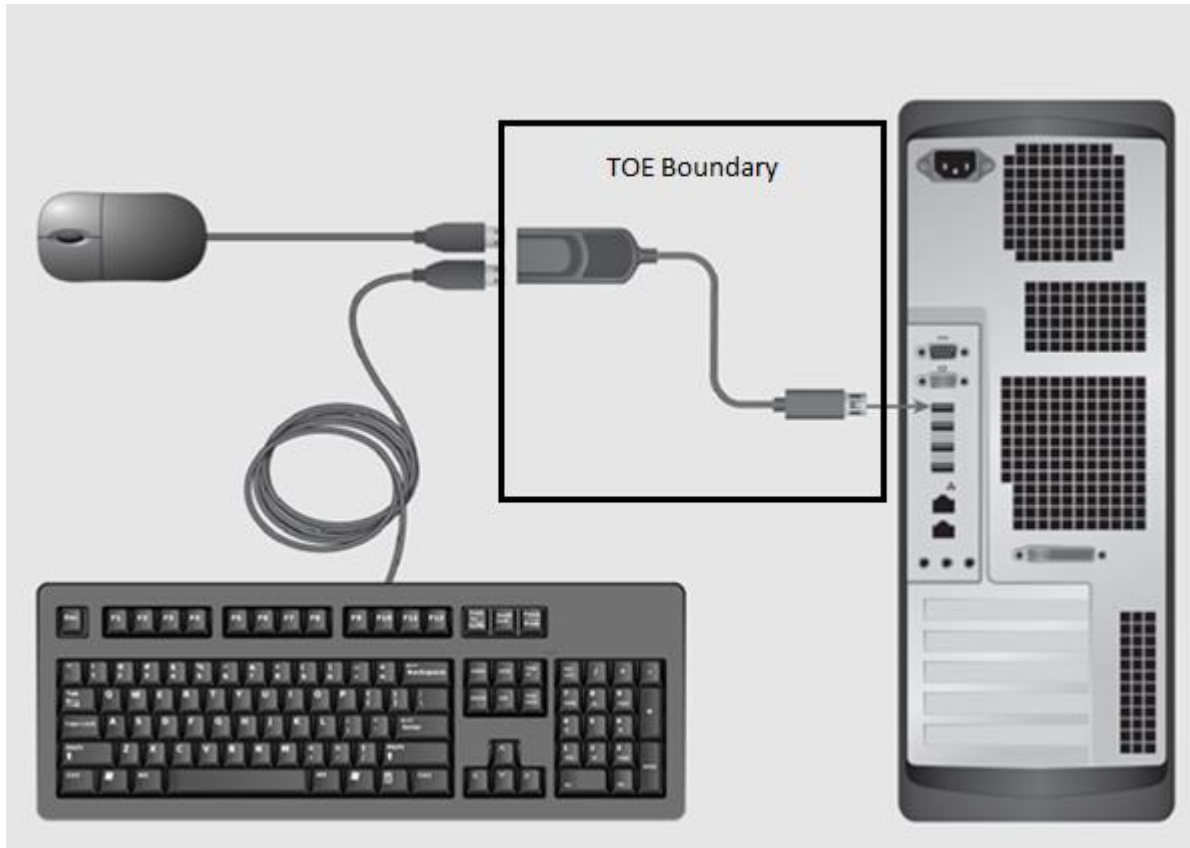
**Table 1 – Non-TOE Hardware and Software**

## 1.5 TOE DESCRIPTION

The TOE includes the following device type:

- HID USB Filter FH10N-4

### 1.5.1 USB Filter Evaluated Configuration



**Figure 2 – USB Filter Evaluated Configuration**

The HID USB Filter is connected to keyboard and mouse peripherals, and to the appropriate USB ports on the computer.

### 1.5.2 Physical Scope

The TOE consists of the following device.

Product Description	Part Number	Model	Tamper Evident labels	Active Anti-Tampering	Number of supported connected computers
USB HID Filter	CGA19436	FH10N-4	Yes	No	1

**Table 2 – TOE Isolator Device and Features**



### 1.5.2.1 TOE Delivery

The TOE, together with its corresponding cables are delivered to the customer via trusted carrier, such as Fed-Ex, that provides a tracking service for all shipments.

### 1.5.2.2 TOE Guidance

The TOE includes the following guidance documentation:

- HSL Quick Installation Guide HSL-s USB eLock Solution, HDC20403 Rev 1.1.

Guidance may be downloaded from the High Sec Labs website (<https://highseclabs.com/quick-start-guides/>) in .pdf format.

The following guidance is available upon request by emailing [support@highseclabs.com](mailto:support@highseclabs.com):

- High Sec Labs FH10N-4 Firmware Version 40404-0E7 Common Criteria Guidance Supplement, Version 0.4

## 1.5.3 Logical Scope

The logical boundary of the TOE includes all interfaces and functions within the physical boundary. The logical boundary of the TOE may be broken down by the security function classes described in Section 6. Table 3 summarizes the logical scope of the TOE.

Functional Classes	Description
User Data Protection	The TOE enforces unidirectional data flow for keyboard and mouse. The TOE ensures that only authorized peripheral devices may be used.
Protection of the TSF <sup>1</sup>	The TOE ensures a secure state in the case of failure, provides only restricted access, and performs self-testing. The TOE provides passive detection of physical attack.

**Table 3 – Logical Scope of the TOE**

---

<sup>1</sup> TOE Security Functionality

## 2 CONFORMANCE CLAIMS

### 2.1 COMMON CRITERIA CONFORMANCE CLAIM

This Security Target claims to be conformant to Version 3.1 of Common Criteria for Information Technology Security Evaluation according to:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components CCMB-2017-04-003, Version 3.1, Revision 5, April 2017

As follows:

- CC Part 2 extended
- CC Part 3 conformant

The Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017 has been taken into account.

### 2.2 PROTECTION PROFILE CONFORMANCE CLAIM

This ST claims exact conformance with the National Information Assurance Partnership (NIAP) PP-Configuration for Peripheral Sharing Device and Keyboard/Mouse Devices, 19 July 2019 [CFG\_PSD-KM\_V1.0].

This PP-Configuration includes the following components:

- Base-PP: Protection Profile for Peripheral Sharing Device, Version 4.0 [PP\_PSD\_V4.0]
- PP-Module: PP-Module for Keyboard/Mouse Devices, Version 1.0 [MOD\_KM\_V1.0].

TD	Name	PP affected	Relevant Y/N
TD0507	Clarification on USB plug type	[MOD_KM_V1.0]	Y
TD0518	Typographical errors in dependency Table	[PP_PSD_V4.0]	N FPT_STM.1 is not

			claimed in the ST
TD0583	FPT_PHP.3 modified for remote controllers	[PP_PSD_V4.0]	Y
TD0593	Equivalency Arguments for PSD	[MOD_KM_V1.0]	Y
TD0804	Clarification regarding Extenders in PSD Evaluations	[PP_PSD_V4.0]	N
TD0844	Addition of Assurance Package for Flaw Remediation V1.0 Conformance Claim	[PP_PSD_V4.0]	N

**Table 4 – Applicable Technical Decisions**

## 2.3 PACKAGE CLAIM

This Security Target does not claim conformance with any Package.

## 2.4 CONFORMANCE RATIONALE

The TOE USB HID Filter is inherently consistent with the Compliant Targets of Evaluation described in the [PP\_PSD\_V4.0] and in the PP module listed in Section 2.2, and with the PP-Configuration for Peripheral Sharing Device and Keyboard/Mouse Devices [CFG\_PSD-KM\_V1.0].

The security problem definition, statement of security objectives and statement of security requirements in this ST conform exactly to the security problem definition, statement of security objectives and statement of security requirements contained in [PP\_PSD\_V4.0] and the module listed in Section 2.2

## 3 SECURITY PROBLEM DEFINITION

### 3.1 THREATS

Table 5 lists the threats described in Section 3.1 of the [PP\_PSD\_V4.0]. Mitigation to the threats is through the objectives identified in Section 4.1, Security Objectives for the TOE.

Threat	Description
<b>T.DATA_LEAK</b>	A connection via the PSD <sup>2</sup> between one or more computers may allow unauthorized data flow through the PSD or its connected peripherals.
<b>T.SIGNAL_LEAK</b>	A connection via the PSD between one or more computers may allow unauthorized data flow through bit-by-bit signaling.
<b>T.RESIDUAL_LEAK</b>	A PSD may leak (partial, residual, or echo) user data between the intended connected computer and another unintended connected computer.
<b>T.UNINTENDED_USE</b>	A PSD may connect the user to a computer other than the one to which the user intended to connect.
<b>T.UNAUTHORIZED_DEVICES</b>	The use of an unauthorized peripheral device with a specific PSD peripheral port may allow unauthorized data flows between connected devices or enable an attack on the PSD or its connected computers.
<b>T.LOGICAL_TAMPER</b>	An attached device (computer or peripheral) with malware, or otherwise under the control of a malicious user, could modify or overwrite code or data stored in the PSD's volatile or non-volatile memory to allow unauthorized information flows.
<b>T.PHYSICAL_TAMPER</b>	A malicious user or human agent could physically modify the PSD to allow unauthorized information flows.
<b>T.REPLACEMENT</b>	A malicious human agent could replace the PSD during shipping, storage, or use with an alternate device that does not enforce the PSD security policies.

<sup>2</sup> Peripheral Sharing Device

Threat	Description
<b>T.FAILED</b>	Detectable failure of a PSD may cause an unauthorized information flow or weakening of PSD security functions.

**Table 5 – Threats**

## 3.2 ORGANIZATIONAL SECURITY POLICIES

There are no Organizational Security Policies applicable to this TOE.

## 3.3 ASSUMPTIONS

The assumptions required to ensure the security of the TOE are listed in Table 6.

Assumptions	Description
<b>A.NO_TEMPEST</b>	Computers and peripheral devices connected to the PSD are not TEMPEST approved.  The TSF may or may not isolate the ground of the keyboard and mouse computer interfaces (the USB ground). The Operational Environment is assumed not to support TEMPEST red-black ground isolation.
<b>A.PHYSICAL</b>	The environment provides physical security commensurate with the value of the TOE and the data it processes and contains.
<b>A.NO_WIRELESS_DEVICES</b>	The environment includes no wireless peripheral devices.
<b>A.TRUSTED_ADMIN</b>	PSD Administrators and users are trusted to follow and apply all guidance in a trusted manner.
<b>A.TRUSTED_CONFIG</b>	Personnel configuring the PSD and its operational environment follow the applicable security configuration guidance.
<b>A.USER_ALLOWED_ACCESS</b>	All PSD users are allowed to interact with all connected computers. It is not the role of the PSD to prevent or otherwise control user access to connected computers. Computers or their connected network shall have the required means to authenticate the user and to control access to their various resources.

**Table 6 – Assumptions**

## 4 SECURITY OBJECTIVES

The purpose of the security objectives is to address the security concerns and to show which security concerns are addressed by the TOE, and which are addressed by the environment. Threats may be addressed by the TOE or the security environment or both. Therefore, the CC identifies two categories of security objectives:

- Security objectives for the TOE
- Security objectives for the environment

### 4.1 SECURITY OBJECTIVES FOR THE TOE

This section identifies and describes the security objectives that are to be addressed by the TOE, and traces each Security Functional Requirement (SFR) back to a security objective of the TOE.

Security Objective	Description		
<b>O.COMPUTER_INTERFACE_ISOLATION</b>	<p>The PSD shall prevent unauthorized data flow to ensure that the PSD and its connected peripheral devices cannot be exploited in an attempt to leak data. The TOE-Computer interface shall be isolated from all other PSD-Computer interfaces while TOE is powered.</p> <p>Addressed by:</p> <table border="1"> <tr> <td>MOD_KM</td><td>FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1</td></tr> </table>	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1
MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1		
<b>O.COMPUTER_INTERFACE_ISOLATION_TOE_UNPOWERED</b>	<p>The PSD shall not allow data to transit a PSD-Computer interface while the PSD is unpowered.</p> <p>Addressed by:</p> <table border="1"> <tr> <td>MOD_KM</td><td>FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1</td></tr> </table>	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1
MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1		
<b>O.USER_DATA_ISOLATION</b>	<p>The PSD shall route user data, such as keyboard entries, only to the computer selected by the user. The PSD shall provide isolation between the data flowing from the peripheral device to the selected computer and any non-selected computer.</p> <p>Addressed by:</p> <table border="1"> <tr> <td>MOD_KM</td><td>FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1</td></tr> </table>	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1
MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1		
<b>O.NO_USER_DATA_RETENTION</b>	<p>The PSD shall not retain user data in non-volatile memory after power up or, if supported, factory reset.</p> <p>Addressed by:</p>		

Security Objective	Description					
	PP_PSD	FDP_RIP_EXT.1				
	MOD_KM	FDP_RIP.1/KM				
<b>O.NO_OTHER_EXTERNAL_INTERFACES</b>	<p>The PSD shall not have any external interfaces other than those implemented by the TSF.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FDP_PDC_EXT.1</td></tr></table>		PP_PSD	FDP_PDC_EXT.1		
PP_PSD	FDP_PDC_EXT.1					
<b>O.LEAK_PREVENTION_SWITCHING</b>	<p>The PSD shall ensure that there are no switching mechanisms that allow signal data leakage between connected computers.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FDP_SWI_EXT.1</td></tr></table>		PP_PSD	FDP_SWI_EXT.1		
PP_PSD	FDP_SWI_EXT.1					
<b>O.AUTHORIZED_USAGE</b>	<p>The TOE shall explicitly prohibit or ignore unauthorized switching mechanisms, either because it supports only one connected computer or because it allows only authorized mechanisms to switch between connected computers. Authorized switching mechanisms shall require express user action restricted to console buttons, console switches, console touch screen, wired remote control, and peripheral devices using a guard. Unauthorized switching mechanisms include keyboard shortcuts, also known as "hotkeys," automatic port scanning, control through a connected computer, and control through keyboard shortcuts. Where applicable, the results of the switching activity shall be indicated by the TSF so that it is clear to the user that the switching mechanism was engaged as intended.</p> <p>A conformant TOE may also provide a management function to configure some aspects of the TSF. If the TOE provides this functionality, it shall ensure that whatever management functions it provides can only be performed by authorized administrators and that an audit trail of management activities is generated.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FDP_SWI_EXT.1</td></tr><tr><td>MOD_KM</td><td>FDP_FIL_EXT.1/KM</td></tr></table>		PP_PSD	FDP_SWI_EXT.1	MOD_KM	FDP_FIL_EXT.1/KM
PP_PSD	FDP_SWI_EXT.1					
MOD_KM	FDP_FIL_EXT.1/KM					
<b>O.PERIPHERAL_PORTS_ISOLATION</b>	<p>The PSD shall ensure that data does not flow between peripheral devices connected to different PSD interfaces.</p> <p>Addressed by:</p>					

Security Objective	Description					
	PP_PSD	FDP_APC_EXT.1				
	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1				
<b>O.REJECT _UNAUTHORIZED _PERIPHERAL</b>	<p>The PSD shall reject unauthorized peripheral device types and protocols.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FDP_PDC_EXT.1</td></tr><tr><td>MOD_KM</td><td>FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1, FDP_SWI_EXT.3, FDP_PDC_EXT.2/KM</td></tr></table>		PP_PSD	FDP_PDC_EXT.1	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1, FDP_SWI_EXT.3, FDP_PDC_EXT.2/KM
PP_PSD	FDP_PDC_EXT.1					
MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1, FDP_SWI_EXT.3, FDP_PDC_EXT.2/KM					
<b>O.REJECT _UNAUTHORIZED _ENDPOINTS</b>	<p>The PSD shall reject unauthorized peripheral devices connected via a Universal Serial Bus (USB) hub.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FDP_PDC_EXT.1</td></tr><tr><td>MOD_KM</td><td>FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1</td></tr></table>		PP_PSD	FDP_PDC_EXT.1	MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1
PP_PSD	FDP_PDC_EXT.1					
MOD_KM	FDP_APC_EXT.1/KM, FDP_FIL_EXT.1/KM, FDP_PDC_EXT.1, FDP_RDR_EXT.1					
<b>O.NO_TOE_ACCESS</b>	<p>The PSD firmware, software, and memory shall not be accessible via its external ports.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FPT_NTA_EXT.1</td></tr></table>		PP_PSD	FPT_NTA_EXT.1		
PP_PSD	FPT_NTA_EXT.1					
<b>O.TAMPER _EVIDENT _LABEL</b>	<p>The PSD shall be identifiable as authentic by the user and the user must be made aware of any procedures or other such information to accomplish authentication. This feature must be available upon receipt of the PSD and continue to be available during the PSD deployment. The PSD shall be labeled with at least one visible unique identifying tamper-evident marking that can be used to authenticate the device. The PSD manufacturer must maintain a complete list of manufactured PSD articles and their respective identification markings' unique identifiers.</p> <p>Addressed by:</p> <table><tr><td>PP_PSD</td><td>FPT_PHP.1</td></tr></table>		PP_PSD	FPT_PHP.1		
PP_PSD	FPT_PHP.1					
<b>O.ANTI_TAMPERING</b>	<p>The PSD shall be physically enclosed so that any attempts to open or otherwise access the internals or modify the connections of the PSD would be evident, and optionally</p>					



Security Objective	Description		
	<p>thwarted through disablement of the TOE. Note: This applies to a wired remote control as well as the main chassis of the PSD.</p> <p>Addressed by:</p> <table> <tr> <td>PP_PSD</td><td>FPT_PHP.1</td></tr> </table>	PP_PSD	FPT_PHP.1
PP_PSD	FPT_PHP.1		
<b>O.SELF_TEST</b>	<p>The PSD shall perform self-tests following power up or powered reset.</p> <p>Addressed by:</p> <table> <tr> <td>PP_PSD</td><td>FPT_TST.1</td></tr> </table>	PP_PSD	FPT_TST.1
PP_PSD	FPT_TST.1		
<b>O.SELF_TEST_FAIL_TOE_DISABLE</b>	<p>The PSD shall enter a secure state upon detection of a critical failure.</p> <p>Addressed by:</p> <table> <tr> <td>PP_PSD</td><td>FPT_FLS_EXT.1, FPT_TST_EXT.1</td></tr> </table>	PP_PSD	FPT_FLS_EXT.1, FPT_TST_EXT.1
PP_PSD	FPT_FLS_EXT.1, FPT_TST_EXT.1		
<b>O.SELF_TEST_FAIL_INDICATION</b>	<p>The PSD shall provide clear and visible user indications in the case of a self-test failure.</p> <p>Addressed by:</p> <table> <tr> <td>PP_PSD</td><td>FPT_TST_EXT.1</td></tr> </table>	PP_PSD	FPT_TST_EXT.1
PP_PSD	FPT_TST_EXT.1		
<b>O.EMULATED_INPUT</b>	<p>The TOE shall emulate the keyboard and/or mouse functions from the TOE to the connected computer.</p> <p>Addressed by:</p> <table> <tr> <td>MOD_KM</td><td>FDP_PDC_EXT.2/KM, FDP_PDC_EXT.3/KM</td></tr> </table>	MOD_KM	FDP_PDC_EXT.2/KM, FDP_PDC_EXT.3/KM
MOD_KM	FDP_PDC_EXT.2/KM, FDP_PDC_EXT.3/KM		
<b>O.UNIDIRECTIONAL_INPUT</b>	<p>The TOE shall enforce unidirectional keyboard and/or mouse device's data flow from the peripheral device to only the selected computer.</p> <p>Addressed by:</p> <table> <tr> <td>MOD_KM</td><td>FDP_UDF_EXT.1/KM</td></tr> </table>	MOD_KM	FDP_UDF_EXT.1/KM
MOD_KM	FDP_UDF_EXT.1/KM		

**Table 7 – Security Objectives for the TOE**

## 4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

This section identifies and describes the security objectives that are to be addressed by the IT environment or by non-technical or procedural means.

Security Objective	Description
<b>OE.NO_TEMPEST</b>	The operational environment will not use TEMPEST approved equipment.
<b>OE.PHYSICAL</b>	The operational environment will provide physical security, commensurate with the value of the PSD and the data that transits it.
<b>OE.NO_WIRELESS_DEVICES</b>	The operational environment will not include wireless keyboards, mice, audio, user authentication, or video devices.
<b>OE.TRUSTED_ADMIN</b>	The operational environment will ensure that trusted PSD Administrators and users are appropriately trained.
<b>OE.TRUSTED_CONFIG</b>	The operational environment will ensure that administrators configuring the PSD and its operational environment follow the applicable security configuration guidance.

**Table 8 – Security Objectives for the Operational Environment**

## 4.3 SECURITY OBJECTIVES RATIONALE

The security objectives rationale describes how the assumptions and threats map to the security objectives.

Threat or Assumption	Security Objective(s)	Rationale
T.DATA_LEAK	O.COMPUTER_INTERFACE_ISOLATION	Isolation of computer interfaces prevents data from leaking between them without authorization.
	O.COMPUTER_INTERFACE_ISOLATION_TOE_UNPOWERED	Maintaining interface isolation while the TOE is in an unpowered state ensures that data cannot leak between computer interfaces.
	O.USER_DATA_ISOLATION	The TOE's routing of data only to the selected computer ensures that it will not leak to any others.
	O.NO_OTHER_EXTERNAL_INTERFACES	The absence of additional external interfaces ensures that there is no unexpected method by which data can be leaked.
	O.PERIPHERAL_PORTS_ISOLATION	Isolation of peripheral ports prevents data from leaking between them without authorization.

Threat or Assumption	Security Objective(s)	Rationale
	O.UNIDIRECTIONAL_INPUT	The TOE's enforcement of unidirectional input for keyboard/mouse data prevents leakage of computer data through a connected peripheral interface.
T.SIGNAL_LEAK	O.COMPUTER_INTERFACE_ISOLATION	Isolation of computer interfaces prevents data leakage through bit-wise signaling because there is no mechanism by which the signal data can be communicated.
	O.NO_OTHER_EXTERNAL_INTERFACES	The absence of additional external interfaces ensures that there is no unexpected method by which data can be leaked through bitwise signaling.
	O.LEAK_PREVENTION_SWITCHING	The TOE's use of switching methods that are not susceptible to signal leakage helps mitigate the signal leak threat.
	O.UNIDIRECTIONAL_INPUT	The TOE's enforcement of unidirectional input for keyboard/mouse data prevents leakage of computer data through bit-by-bit signaling to a connected peripheral interface.
T.RESIDUAL_LEAK	O.NO_USER_DATA_RETENTION	The TOE's lack of data retention ensures that a residual data leak is not possible.
T.UNINTENDED_USE	O.AUTHORIZED_USAGE	The TOE's support for only switching mechanisms that require explicit user action to engage ensures that a user has sufficient information to avoid interacting with an unintended computer.
T.UNAUTHORIZED_DEVICES	O.REJECT_UNAUTHORIZED_ENDPOINTS	The TOE's ability to reject unauthorized endpoints mitigates the threat of unauthorized devices being used to communicate with connected computers.
	O.REJECT_UNAUTHORIZED_PERIPHERAL	The TOE's ability to reject unauthorized peripherals mitigates the threat of unauthorized devices being used to communicate with connected computers.

Threat or Assumption	Security Objective(s)	Rationale
	O.EMULATED_INPUT	The TOE's emulation of keyboard/mouse data input ensures that a connected computer will only receive this specific type of data through a connected peripheral.
T.LOGICAL_TAMPER	O.NO_TOE_ACCESS	The TOE's prevention of logical access to its firmware, software, and memory mitigates the threat of logical tampering.
	O.EMULATED_INPUT	The TOE's emulation of keyboard/mouse data input prevents logical tampering of the TSF ensuring that only known inputs to it are supported.
T.PHYSICAL_TAMPER	O.ANTI_TAMPERING	The TOE mitigates the threat of physical tampering through use of an enclosure that provides tamper detection functionality.
	O.TAMPER_EVIDENT_LABEL	The TOE mitigates the threat of physical tampering through use of tamper evident labels that reveal physical tampering attempts.
T.REPLACEMENT	O.TAMPER_EVIDENT_LABEL	The TOE's use of a tamper evident label that provides authenticity of the device mitigates the threat that it is substituted for a replacement device during the acquisition process.
T.FAILED	O.SELF_TEST	The TOE mitigates the threat of failures leading to compromise of security functions through self-tests of its own functionality.
	O.SELF_TEST_FAIL_TOE_DISABLE	The TOE mitigates the threat of failures leading to compromise of security functions by disabling all data flows in the event a failure is detected.
	O.SELF_TEST_FAIL_INDICATION	The TOE mitigates the threat of failures leading to compromise of security functions by providing users with a clear indication when it is in a failure state and should not be trusted.
A.NO_TEMPEST	OE.NO_TEMPEST	If the TOE's operational environment does not include TEMPEST approved

Threat or Assumption	Security Objective(s)	Rationale
		equipment, then the assumption is satisfied.
A.NO_PHYSICAL	OE.PHYSICAL	If the TOE's operational environment provides physical security, then the assumption is satisfied.
A.NO_WIRELESS_DEVICES	OE.NO_WIRELESS_DEVICES	If the TOE's operational environment does not include wireless peripherals, then the assumption is satisfied.
A.TRUSTED_ADMIN	OE.TRUSTED_ADMIN	If the TOE's operational environment ensures that only trusted administrators will manage the TSF, then the assumption is satisfied.
A.TRUSTED_CONFIG	OE.TRUSTED_CONFIG	If TOE administrators follow the provided security configuration guidance, then the assumption is satisfied.
A.USER_ALLOWED_ACCESS	OE.PHYSICAL	If the TOE's operational environment provides physical access to connected computers, then the assumption is satisfied.

**Table 9 – Security Objectives Rationale**

## 5 EXTENDED COMPONENTS DEFINITION

The extended components definition is presented in Appendix C of the Protection Profile for Peripheral Sharing Device [PP\_PSD\_V4.0] and in the module for keyboard/mouse devices [MOD\_KM\_V1.0].

The families to which these components belong are identified in the following table:

Functional Class	Functional Families	Protection Profile Modules
User Data Protection (FDP)	FDP_APC_EXT Active PSD Connections	[PP_PSD_V4.0]
	FDP_FIL_EXT Device Filtering	[PP_PSD_V4.0] [MOD_KM_V1.0]
	FDP_PDC_EXT Peripheral Device Connection	[MOD_KM_V1.0]
	FDP_RDR_EXT Re-Enumeration Device Rejection	[MOD_KM_V1.0]
	FDP_RIP_EXT Residual Information Protection	[PP_PSD_V4.0]
	FDP_UDF_EXT Unidirectional Data Flow	[MOD_KM_V1.0]
Protection of the TSF (FPT)	FPT_FLS_EXT Failure with Preservation of Secure State	[PP_PSD_V4.0]
	FPT_NTA_EXT No Access to TOE	[PP_PSD_V4.0]
	FPT_TST_EXT TSF Testing	[PP_PSD_V4.0]

**Table 10 – Functional Families of Extended Components**

## 6 SECURITY REQUIREMENTS

Section 6 provides security functional and assurance requirements that must be satisfied by a compliant TOE.

### 6.1 CONVENTIONS

The CC permits four types of operations to be performed on functional requirements: selection, assignment, refinement, and iteration. This is defined as:

- Assignment: Indicated by surrounding brackets and underline, e.g., [assigned item].
- Selection: Indicated by surrounding brackets and italics, e.g., [*selected item*].
- Refinement: Refined components are identified by using **[bold surrounded by brackets]** for additional information, or ~~[strikeout surrounded by brackets]~~ for deleted text.
- Iteration: Iteration operations for iterations within the Protection Profile and associated modules are identified with a slash ('/') and an identifier (e.g. "/KM"). Where multiple iterations of the SFR are required within the ST, a number is appended to the SFR identifier (e.g. "FDP\_CDS\_EXT.1(1)").

Extended SFRs are identified by the inclusion of "EXT" in the SFR name.

The CC operations already performed in the PP and PP modules are reproduced in plain text and not denoted in this ST. The requirements have been copied from the PP and PP modules and any remaining operations have been completed herein. Refer to the PP and PP modules to identify those operations.

### 6.2 SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for this ST consist of the following components.

Class	Identifier	Name	Source
User Data Protection (FDP)	FDP_APC_EXT.1/KM	Active PSD Connections	[MOD_KM_V1.0]
	FDP_FIL_EXT.1/KM	Device Filtering (Keyboard/Mouse)	[MOD_KM_V1.0] <sup>3</sup>

<sup>3</sup> There is no modification to this SFR in the [MOD\_KM\_V1.0]. However, there are additions to the Peripheral Device Connections associated with this SFR and additional evaluation activities.

Class	Identifier	Name	Source
	FDP_PDC_EXT.1	Peripheral Device Connection	[PP_PSD_V4.0] [MOD_KM_V1.0]
	FDP_PDC_EXT.2/KM	Authorized Devices (Keyboard/Mouse)	[MOD_KM_V1.0]
	FDP_PDC_EXT.3/KM	Authorized Connection Protocols (Keyboard/Mouse)	[MOD_KM_V1.0]
	FDP_RDR_EXT.1	Re-Enumeration Device Rejection	[MOD_KM_V1.0]
	FDP_RIP_EXT.1	Residual Information Protection	[PP_PSD_V4.0]
	FDP_RIP.1/KM	Residual Information Protection (Keyboard Data)	[MOD_KM_V1.0]
	FDP_SWI_EXT.1	PSD Switching	[PP_PSD_V4.0]
	FDP_UDF_EXT.1/KM	Unidirectional Data Flow (Keyboard/Mouse)	[MOD_KM_V1.0]
Protection of the TSF (FPT)	FPT_FLS_EXT.1	Failure with Preservation of Secure State	[PP_PSD_V4.0]
	FPT_NTA_EXT.1	No Access to TOE	[PP_PSD_V4.0]
	FPT_PHP.1	Passive Detection of Physical Attack	[PP_PSD_V4.0]
	FPT_TST.1	TSF testing	[PP_PSD_V4.0]
	FPT_TST_EXT.1	TSF Testing	[PP_PSD_V4.0]

**Table 11 – Summary of Security Functional Requirements**

## 6.2.1 User Data Protection (FDP)

### 6.2.1.1 FDP\_APC\_EXT.1/KM Active PSD Connections

**FDP\_APC\_EXT.1.1/KM** The TSF shall route user data only to the interfaces selected by the user.



**FDP\_APC\_EXT.1.2/KM** The TSF shall ensure that no data or electrical signals flow between connected computers whether the TOE is powered on or powered off.

**FDP\_APC\_EXT.1.3/KM** The TSF shall ensure that no data transits the TOE when the TOE is powered off.

**FDP\_APC\_EXT.1.4/KM** The TSF shall ensure that no data transits the TOE when the TOE is in a failure state.

### 6.2.1.2 FDP\_FIL\_EXT.1/KM Device Filtering (Keyboard/Mouse)

**FDP\_FIL\_EXT.1.1/KM** The TSF shall have [*fixed*] device filtering for [*keyboard, mouse*] interfaces.

**FDP\_FIL\_EXT.1.2/KM** The TSF shall consider all PSD KM blacklisted devices as unauthorized devices for [*keyboard, mouse*] interfaces in peripheral device connections.

**FDP\_FIL\_EXT.1.3/KM** The TSF shall consider all PSD KM whitelisted devices as authorized devices for [*keyboard, mouse*] interfaces in peripheral device connections only if they are not on the PSD KM blacklist or otherwise unauthorized.

### 6.2.1.3 FDP\_PDC\_EXT.1 Peripheral Device Connection

**FDP\_PDC\_EXT.1.1** The TSF shall reject connections with unauthorized devices upon TOE power up and upon connection of a peripheral device to a powered-on TOE.

**FDP\_PDC\_EXT.1.2** The TSF shall reject connections with devices presenting unauthorized interface protocols upon TOE power up and upon connection of a peripheral device to a powered-on TOE.

**FDP\_PDC\_EXT.1.3** The TOE shall have no external interfaces other than those claimed by the TSF.

**FDP\_PDC\_EXT.1.4** The TOE shall not have wireless interfaces.

**FDP\_PDC\_EXT.1.5** The TOE shall provide a visual or auditory indication to the User when a peripheral is rejected.

### 6.2.1.4 FDP\_PDC\_EXT.2/KM Authorized Devices (Keyboard/Mouse)

**FDP\_PDC\_EXT.2.1/KM** The TSF shall allow connections with authorized devices and functions as defined in Appendix E [**of [MOD\_KM\_V1.0]**] and [  

- *no other devices*

]  
] upon TOE power up and upon connection of a peripheral device to a powered-on TOE.

**FDP\_PDC\_EXT.2.2/KM** The TSF shall allow connections with authorized devices presenting authorized interface protocols as defined in Appendix E [**of [MOD\_KM\_V1.0]**] and [  
]

- *no other devices*

] upon TOE power up and upon connection of a peripheral device to a powered-on TOE.

#### **6.2.1.5 FDP\_PDC\_EXT.3/KM Authorized Connection Protocols (Keyboard/Mouse)**

**FDP\_PDC\_EXT.3.1/KM** The TSF shall have interfaces for the [*USB (keyboard), USB (mouse)*] protocols.

**FDP\_PDC\_EXT.3.2/KM** The TSF shall apply the following rules to the supported protocols: the TSF shall emulate any keyboard or mouse device functions from the TOE to the connected computer.

#### **6.2.1.6 FDP\_RDR\_EXT.1 Re-Enumeration Device Rejection**

**FDP\_RDR\_EXT.1.1** The TSF shall reject any device that attempts to enumerate again as a different unauthorized device.

#### **6.2.1.7 FDP\_RIP\_EXT.1 Residual Information Protection**

**FDP\_RIP\_EXT.1.1** The TSF shall ensure that no user data is written to TOE non-volatile memory or storage.

#### **6.2.1.8 FDP\_RIP.1/KM Residual Information Protection (Keyboard Data)**

**FDP\_RIP.1.1/KM** The TSF shall ensure that any keyboard data in volatile memory is purged upon switching computers.

#### **6.2.1.9 FDP\_SWI\_EXT.3 Tied Switching**

**FDP\_SWI\_EXT.3.1** The TSF shall ensure that connected keyboard and mouse peripheral devices are always switched together to the same connected computer.

#### **6.2.1.10 FDP\_SWI\_EXT.1 PSD Switching**

**FDP\_SWI\_EXT.1.1** The TSF shall ensure that [*the TOE supports only one connected computer, switching can be initiated only through express user action*].

#### **6.2.1.11 FDP\_UDF\_EXT.1/KM Unidirectional Data Flow (Keyboard/Mouse)**

**FDP\_UDF\_EXT.1.1/KM** The TSF shall ensure [*keyboard, mouse*] data transits the TOE unidirectionally from the TOE [*keyboard, mouse*] peripheral interface(s) to the TOE [*keyboard, mouse*] interface.

## 6.2.2 Protection of the TSF (FPT)

### 6.2.2.1 FPT\_FLS\_EXT.1 Failure with Preservation of Secure State

**FPT\_FLS\_EXT.1.1** The TSF shall preserve a secure state when the following types of failures occur: failure of the power-on self-test and [*no other failures*].

### 6.2.2.2 FPT\_NTA\_EXT.1 No Access to TOE

**FPT\_NTA\_EXT.1.1** TOE firmware, software, and memory shall not be accessible via the TOE's external ports, with the following exceptions: [*no other exceptions*].

### 6.2.2.3 FPT\_PHP.1 Passive Detection of Physical Attack

**FPT\_PHP.1.1** The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

**FPT\_PHP.1.2** The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

### 6.2.2.4 FPT\_TST.1 TSF Testing

**FPT\_TST.1.1** The TSF shall run a suite of self tests during initial start-up and at the conditions [*no other conditions*] to demonstrate the correct operation of user control functions and [*no other functions*].

**FPT\_TST.1.2** The TSF shall provide authorized users with the capability to verify the integrity of [*TSF data*].

**FPT\_TST.1.3** The TSF shall provide authorized users with the capability to verify the integrity of [*TSF*].

### 6.2.2.5 FPT\_TST\_EXT.1 TSF Testing

**FPT\_TST\_EXT.1.1** The TSF shall respond to a self-test failure by providing users with a [*visual, auditory*] indication of failure and by shutdown of normal TSF functions.

## 6.3 SECURITY ASSURANCE REQUIREMENTS

The assurance requirements are summarized in Table 12.

Assurance Class	Assurance Components	
	Identifier	Name
Development (ADV)	ADV_FSP.1	Basic Functional Specification
Guidance Documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-Cycle Support (ALC)	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM Coverage
Security Target Evaluation (ASE)	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
Tests (ATE)	ATE_IND.1	Independent Testing - Conformance
Vulnerability Assessment (AVA)	AVA_VAN.1	Vulnerability Survey

**Table 12 – Security Assurance Requirements**

## 6.4 SECURITY REQUIREMENTS RATIONALE

### 6.4.1 Security Functional Requirements Rationale

Table 7 provides a mapping between the SFRs and Security Objectives.

## 6.4.2 Dependency Rationale

Table 13 identifies the Security Functional Requirements and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency.

SFR	Dependencies	Rationale Statement
FDP_APC_EXT.1/KM	None	N/A
FDP_FIL_EXT.1/KM	FDP_PDC_EXT.1	Included
FDP_PDC_EXT.1	None	N/A
FDP_PDC_EXT.2/KM	FDP_PDC_EXT.1	Included
FDP_PDC_EXT.3/KM	FDP_PDC_EXT.1	Included
FDP_RDR_EXT.1	FDP_PDC_EXT.1	Included
FDP_RIP_EXT.1	None	N/A
FDP_RIP.1/KM	None	N/A
FDP_SWI_EXT.1	None	N/A
FDP_UDF_EXT.1/KM	FDP_APC_EXT.1	Included
FPT_FLS_EXT.1	FPT_TST.1 FPT_PHP.3	Included N/A (only required if anti-tamper is selected in FPT_FLS_EXT.1.1).
FPT_NTA_EXT.1	None	N/A
FPT_PHP.1	None	N/A
FPT_TST.1	None	N/A
FPT_TST_EXT.1	FPT_TST.1	Included

**Table 13 – Functional Requirement Dependencies**

## 6.4.3 Security Assurance Requirements Rationale

The TOE assurance requirements for this ST consist of the requirements indicated in the [PP\_PSD\_V4.0].

## 7 TOE SUMMARY SPECIFICATION

This section provides a description of the security functions and assurance measures of the TOE that meet the TOE security requirements.

### 7.1 USER DATA PROTECTION

#### 7.1.1 System Controller

Each device includes a System Controller which is responsible for device management, system control security functions, and device monitoring.

The System Controller includes a microcontroller with internal non-volatile, Read Only Memory (ROM). The controller function manages the TOE functionality through a pre-programmed state machine loaded on the ROM as read-only firmware during product manufacturing.

The TOE supports only one connected computer.

**TOE Security Functional Requirements addressed:** FDP\_SWI\_EXT.1.

##### 7.1.1.1 Active PSD Connections

The TOE ensures that data flows only one-way between the peripherals and the connected computer. No data transits the TOE when the TOE is powered off, or when the TOE is in a failure state. A failure state occurs when the TOE fails a self-test when powering on.

**TOE Security Functional Requirements addressed:** FDP\_APC\_EXT.1/KM.

##### 7.1.1.2 Connected Computer Interfaces

The connected computers are attached to the TOE as follows:

- The TOE connects to the keyboard and mouse port using a USB A to USB B cable. The USB A end attaches to the computer, and the USB B end attaches to the TOE

**TOE Security Functional Requirements addressed:** FDP\_PDC\_EXT.1.

### 7.1.1.3 Residual Information Protection

The Letter of Volatility is included as Annex A.

**TOE Security Functional Requirements addressed:** FDP\_RIP\_EXT.1.

## 7.1.2 Keyboard and Mouse Functionality

### 7.1.2.1 Keyboard and Mouse Enumeration

The TOE determines whether or not a peripheral device that has been plugged into the keyboard and mouse peripheral ports is allowed to operate with the TOE. The TOE uses optical data diodes to enforce a unidirectional data flow from the user peripherals to the coupled host and uses isolated device emulators to prevent data leakage through the peripheral switching circuitry.

The Static Random Access Memory (SRAM) in the host and device emulator circuitry stores USB Host stack parameters and up to the last 4 key codes. User data may be briefly retained; however, there are no data buffers. Data is erased during power off of the device.

The TOE supports USB Type A HID's on keyboard and mouse ports. The USB bidirectional communication protocol is converted into a unidirectional proprietary protocol, and is then converted back into the USB bidirectional protocol to communicate with the coupled computer host.

A USB keyboard is connected to the TOE keyboard host emulator through the console keyboard port. The keyboard host emulator is a microcontroller which enumerates the connected keyboard and verifies that it is a permitted device type. Once the keyboard has been verified, the USB keyboard sends scan codes, which are generated when the user types. These scan codes are converted by the keyboard host emulator into a proprietary protocol data stream that is combined with the data stream from the mouse host emulator.

Similarly, the USB mouse is connected to the TOE mouse host emulator through the USB mouse port. The mouse host emulator is a microcontroller which enumerates the connected mouse and verifies that it is a permitted device type. Once the mouse device has been verified, it sends serial data generated by mouse movement and button use. The mouse serial data is converted by the mouse host emulator into a proprietary protocol data stream that is combined with the data stream from the keyboard host emulator.

**TOE Security Functional Requirements addressed:** FDP\_PDC\_EXT.3/KM, FDP\_UDF\_EXT.1/KM.

### 7.1.2.2 Keyboard and Mouse Data Stream

The combined mouse and keyboard data stream is passed through an optical data diode and routed to the host device emulator. The optical data diode is an opto-coupler designed to physically prevent reverse data flow.

Device emulators are USB enabled microcontrollers that are programmed to emulate a standard USB keyboard and mouse composite device. The combined

data stream is converted back to bidirectional data before reaching the host computer.

Since the keyboard and mouse function are emulated by the TOE, the connected computer is not able to send data to the keyboard that would allow it to indicate that Caps Lock, Num Lock or Scroll Lock are set. These are indicated on the TOE front panel, on the right hand side.

### **7.1.2.3 Keyboard and Mouse Compatible Device Types**

The TOE employs fixed device filtering and accepts only USB HID devices at the keyboard and mouse peripheral ports. Only USB Type A connections are permitted. The TOE does not support a wireless connection to a mouse, keyboard or USB hub.

**TOE Security Functional Requirements addressed:** FDP\_PDC\_EXT.1, FDP\_PDC\_EXT.2/KM, FDP\_FIL\_EXT.1/KM.

### **7.1.2.4 Re-Enumeration Device Rejection**

If a connected device attempts to re-enumerate as a different USB device type, it will be rejected by the TOE. The TOE will reject devices which are not allowed at any time during operation and start-up. This is indicated by an LED on the TOE next to the Keyboard and mouse ports. This LED shows a solid green light for an accepted device, flickering green light during enumeration, and no light for a rejected device.

**TOE Security Functional Requirements addressed:** FDP\_RDR\_EXT.1.

## **7.2 PROTECTION OF THE TSF**

### **7.2.1 No Access to TOE**

The connected computer does not have access to TOE firmware or memory.

The TOE microcontroller runs from internal protected flash memory. Firmware cannot be updated from an external source. Firmware cannot be read or rewritten through the use of Joint Test Action Group (JTAG) tools. Firmware is executed on Static Random Access Memory (SRAM) with the appropriate protections to prevent external access and tampering of code or stacks.

**TOE Security Functional Requirements addressed:** FPT\_NTA\_EXT.1.

### **7.2.2 Anti-tampering Functionality**

The TOE provides passive anti-tampering functionality.

#### **7.2.2.1 Passive Detection of Physical Tampering**

The TOE enclosure was designed specifically to prevent physical tampering and features fitted molded plastic parts.



Additionally, each device is fitted with one or more holographic Tampering Evident Labels placed at critical locations on the TOE enclosure. If the label is removed, the word 'VOID' appears on both the label and the product surface.

**TOE Security Functional Requirements addressed:** FPT\_PHP.1.

### 7.2.3 TSF Testing

The TOE performs a self-test at initial start-up to verify the integrity of the microcontroller firmware.

If the self-test fails, the LEDs on the front panel blink to indicate the failure. The TOE remains in a disabled state until the self-test is rerun and passes.

**TOE Security Functional Requirements addressed:** FPT\_FLS\_EXT.1(1), FPT\_TST.1, FPT\_TST\_EXT.1.

## 8 TERMINOLOGY AND ACRONYMS

### 8.1 TERMINOLOGY

The following terminology is used in this ST:

Term	Description
KM	KM refers to the requirements for Keyboard/Mouse Devices.

**Table 14 – Terminology**

### 8.2 ACRONYMS

The following acronyms are used in this ST:

Acronym	Definition
CC	Common Criteria
DE	Device Emulator
EEPROM	Electrically Erasable Programmable Read-Only Memory
HE	Host Emulator
HID	Human Interface Device
HSL	High Sec Labs
IT	Information Technology
JTAG	Joint Test Action Group
KM	Keyboard, Mouse
LED	Light Emitting Diode
NIAP	National Information Assurance Partnership
OTP	One Time Programming
PP	Protection Profile
PSD	Peripheral Sharing Device
ROM	Read Only Memory
SFR	Security Functional Requirement
SRAM	Static Random Access Memory
ST	Security Target
TOE	Target of Evaluation

Acronym	Definition
TSF	TOE Security Functionality
USB	Universal Serial Bus

**Table 15 – Acronyms**

## 9 REFERENCES

Identifier	Title
[CC]	Common Criteria for Information Technology Security Evaluation – <ul style="list-style-type: none"><li>• Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017</li><li>• Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017</li><li>• Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017</li></ul>
[CEM]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, CCMB-2017-04-004, Version 3.1 Revision 5, April 2017
[PP_PSD_V4.0]	Protection Profile for Peripheral Sharing Device, Version: 4.0, 2019-07-19
[MOD_KM_V1.0]	PP-Module for Keyboard/Mouse Devices, Version 1.0, 2019-07-19
[CFG_PSD-KM_V1.0]	PP-Configuration for Peripheral Sharing Device and Keyboard/Mouse Devices, 19 July 2019

**Table 16 – References**

## ANNEX A – LETTER OF VOLATILITY

The table below provides volatility information and memory types for the High Sec Labs Peripheral Sharing Devices with keyboard and mouse. User data is not retained in any TOE device when the power is turned off.

Product Model	No. in each product	Function, Manufacturer and Part Number	Power Source (if not the TOE)	Storage Type	Size	Volatility	Contains User Data
FH10N-4	1	System Controller, Host emulator: NXP MIMXRT1052DVJ6B	Connected computer	Embedded SRAM <sup>1</sup>	512KB	Volatile	May contain user data
				Embedded Flash <sup>2</sup>	256KB	Non-Volatile	No user data
				Embedded EEPROM <sup>3</sup>	4KB	Non-Volatile	No user data
				OTP Memory	512bytes	Non-Volatile	No user data
	1	Boot flash for USB Controller	System Controller, Host Emulator	External Flash	2MB	Volatile	No user data
	1	Device emulator: ST Microelectronics STM32F070C6T6	Connected computer	Embedded SRAM <sup>1</sup>	6KB	Volatile	May contain user data
				Embedded Flash <sup>2</sup>	32KB	Non-Volatile	No user data
				Embedded EEPROM <sup>3</sup>	4KB	Non-Volatile	No user data

### Notes:

<sup>1</sup> SRAM stores USB Host stack parameters and up to the last 4 key-codes. Data is erased during power off of the device. Device emulators receive power from the connected computer and therefore devices are powered on as long as the associated computer is powered on and connected.

<sup>2</sup> Flash storage is used to store firmware code. It contains no user data. Flash storage is permanently locked by fuses after initial programming to prevent rewriting. It is an integral part of the microcontroller together with SRAM and EEPROM.

<sup>3</sup> EEPROM is used to store operational parameters. They contain no user data. These devices receive power from the computer connected to the TOE, and therefore are powered on as long as the associated computer is powered on and connected.