Pivot3, Inc.

Acuity 10.4

Security Target

Evaluation Assurance Level (EAL): EAL2+
Document Version: 0.6

Prepared for:

Pivot3, Inc.
221 W. Sixth Street
Suite 750
Austin, TX 78701
United States of America

Phone: +1 855 236 6187
www.pivot3.com

Prepared by:

Corsec Security, Inc.
13921 Park Center Road
Suite 460
Herndon, VA 20171
United States of America

Phone: +1 703 267 6050
www.corsec.com

Table of Contents

1. Introduction 4
1.1 Purpose 4
1.2 Security Target and TOE References 5
1.3 Product Overview 5
1.4 TOE Overview 7
1.4.1 TOE Components 8
1.5 TOE Environment 9
1.6 TOE Description 9
1.6.1 Physical Scope 9
1.6.2 Logical Scope 10
2. Conformance Claims 12
3. Security Problem 13
3.1 Threats to Security 13
3.2 Organizational Security Policies 13
3.3 Assumptions 14
4. Security Objectives 15
4.1 Security Objectives for the TOE 15
4.2 Security Objectives for the Operational Environment 15
4.2.1 IT Security Objectives 15
4.2.2 Non-IT Security Objectives 16
5. Extended Components 17
5.1 Extended TOE Security Functional Components 17
5.1.1 Class EXT_FHA: Extended High Availability 17
5.2 Extended TOE Security Assurance Components 18
6. Security Requirements 19
6.1 Conventions 19
6.2 Security Functional Requirements 19
6.2.1 Class FAU: Security Audit 20
6.2.2 Class FDP: User Data Protection 21
6.2.3 Class FIA: Identification and Authentication 22
6.2.4 Class FMT: Security Management 23
6.2.5 Class FPT: Protection of the TSF 24
6.2.6 Class FRU: Resource Utilization 24
6.2.7 Class EXT_FHA: Extended High Availability Functionality 25
6.3 Security Assurance Requirements 25
7. TOE Summary Specification 27
7.1 TOE Security Functionality 27
7.1.1 Security Audit 28
7.1.2 User Data Protection 28
7.1.3 Identification and Authentication 29
7.1.4 Security Management 29
7.1.5 Protection of the TSF 30
7.1.6 Resource Utilization 30

7.1.7 Extended High Availability Functionality 30
8. Rationale 32
8.1 Conformance Claims Rationale 32
8.2 Security Objectives Rationale 32
8.2.1 Security Objectives Rationale Relating to Threats 32
8.2.2 Security Objectives Rationale Relating to Policies 34
8.2.3 Security Objectives Rationale Relating to Assumptions 34
8.3 Rationale for Extended Security Functional Requirements 35
8.4 Rationale for Extended TOE Security Assurance Requirements 35
8.5 Security Requirements Rationale 35
8.5.1 Rationale for Security Functional Requirements of the TOE Objectives 36
8.5.2 Security Assurance Requirements Rationale 38
8.5.3 Dependency Rationale 38
9. Acronyms 40

List of Figures

Figure 1 – Pivot3 Hyperconvergence Architecture 6
Figure 2 – Deployment Configuration of the TOE 8
Figure 3 – EXT_FHA: Extended High Availability Class Decomposition 17
Figure 4 – EXT_FHA Component Testing family decomposition 17

List of Tables

Table 1 – ST and TOE References 5
Table 2 – CC and PP Conformance 12
Table 3 – Threats 13
Table 4 – Organizational Security Policies 14
Table 5 – Assumptions 14
Table 6 – Security Objectives for the TOE 15
Table 7 – IT Security Objectives 15
Table 8 – Non-IT Security Objectives 16
Table 9 – Extended TOE Security Functional Requirements 17
Table 10 – TOE Security Functional Requirements 19
Table 11 – Assurance Requirements 25
Table 12 – Mapping of TOE Security Functionality to Security Functional Requirements 27
Table 13 – Quick Diagnostic Log Audit Record Contents 28
Table 15 – Threats: Objectives Mapping 32
Table 16 – Policies: Objectives Mapping 34
Table 17 – Assumptions: Objectives Mapping 34
Table 18 – Objectives: SFRs Mapping 36
Table 19 – Functional Requirements Dependencies 38
Table 20 – Acronyms 40

1. Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the ST organization. The TOE is the Pivot3, Inc. (Pivot3) Acuity 10.4 and will hereafter be referred to as Acuity or the TOE throughout this document. The TOE is designed to combine virtualization capabilities with a global storage pool to create a complete virtual infrastructure. Customers can use the virtual infrastructure to host and store a variety of Virtual Machines (VMs) that can offer applications and services to suit customer needs.

The TOE runs on a heavily customized, Linux-based VM that maintains direct access to storage to provide these capabilities. Managed through a vCenter Plugin, Acuity leverages vCenter's Single Sign-On (SSO) for user management and authentication. Available storage in the environment is pooled globally and is only accessible to other VMs running on Pivot3 nodes. Acuity uses a fault tolerance technique called Scalar Erasure Coding (SEC) to provide fault tolerance that exceeds the reliability metrics provided by traditional RAID[1] designs. This allows Acuity to simultaneously lose up to three disks or one entire node plus one disk across the global storage pool with no loss of data.

1.1 Purpose

This ST is divided into nine sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functionality and describes the physical and logical scope for the TOE as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any Common Criteria (CC), Protection Profile (PP), and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, organizational security policies, and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Requirements (Section 6) – Presents the SFRs and SARs to which the TOE adheres.
- TOE Summary Specification (Section 7) – Describes the security functions provided by the TOE that satisfy the SFRs and objectives.
- Rationale (Section 8) – Presents the rationale for the security objectives, requirements, and SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms (Section 9) – Defines the acronyms used within this ST.

[1] RAID – Redundant Array of Independent Disks

1.2 Security Target and TOE References

Table 1 below shows the ST and TOE references.

Table 1 – ST and TOE References

ST Title	*Pivot3, Inc. Acuity 10.4 Security Target*
ST Version	Version 0.6
ST Author	Corsec Security, Inc.
ST Publication Date	September 27, 2018
TOE Reference	Pivot3 Acuity 10.4 consisting of: • Pivot3 Acuity Software Platform pvt00.10.4.0.082E • Pivot3 vCenter Plugin v10.4.0.2207 • Pivot3 USB[2] Key identified by a unique serial number

1.3 Product Overview

Pivot3 is a provider of Dynamic Hyperconverged Infrastructure (DHCI) solutions that combine VMware vSphere virtualization technology with a linear-scaling all-flash or hybrid-flash and traditional spinning disk storage array. The system is composed of COTS[3] enterprise hardware. The hardware is operated from a VM running a heavily customized, Linux-based controller called the Acuity Software Platform. The Acuity Software Platform maintains direct access to the underlying storage, bypassing the virtualization layer, and provides a storage cluster available to VMs running on any Pivot3 system (node) within a cluster. Each node contains a VMware ESXi hypervisor on which the VMs execute. The Acuity Software Platform presents storage to the hypervisor as an iSCSI[4] target, simplifying the administration of connecting storage to VMs and eliminating the need to use complicated LUN[5] masking to access storage.

In an Acuity system, resources can be modularized by placing them into logical units called Virtual Performance Groups (vPGs). vPGs allow specific combinations of resources to be logically grouped to provide different tiers of service. This allows Acuity to meet the needs of each virtualized application while maintaining overall business goals of performance and availability.

Hyperconvergence is an emerging technology that refers to complete systems that provide both compute resources for running a VM infrastructure and shared storage for use by VMs. Hyperconverged solutions run entirely on x86 servers with commodity internal solid-state and hard-disk drives for storage. Customers deploy the system as appliances that scale in a linear fashion; each node added to a vPG contributes a fixed amount of computational power and storage capacity. Hyperconvergence relies on software defined storage as an underlying technology, which allows the storage within individual servers to be shared across every node in a vPG. Figure 1 below shows Pivot3's hyperconvergence architecture.

[2] USB – Universal Serial Bus
[3] COTS – Commercial off-the-Shelf
[4] iSCSI – Internet Small Computer Systems Interface
[5] LUN – Logical Unit Number

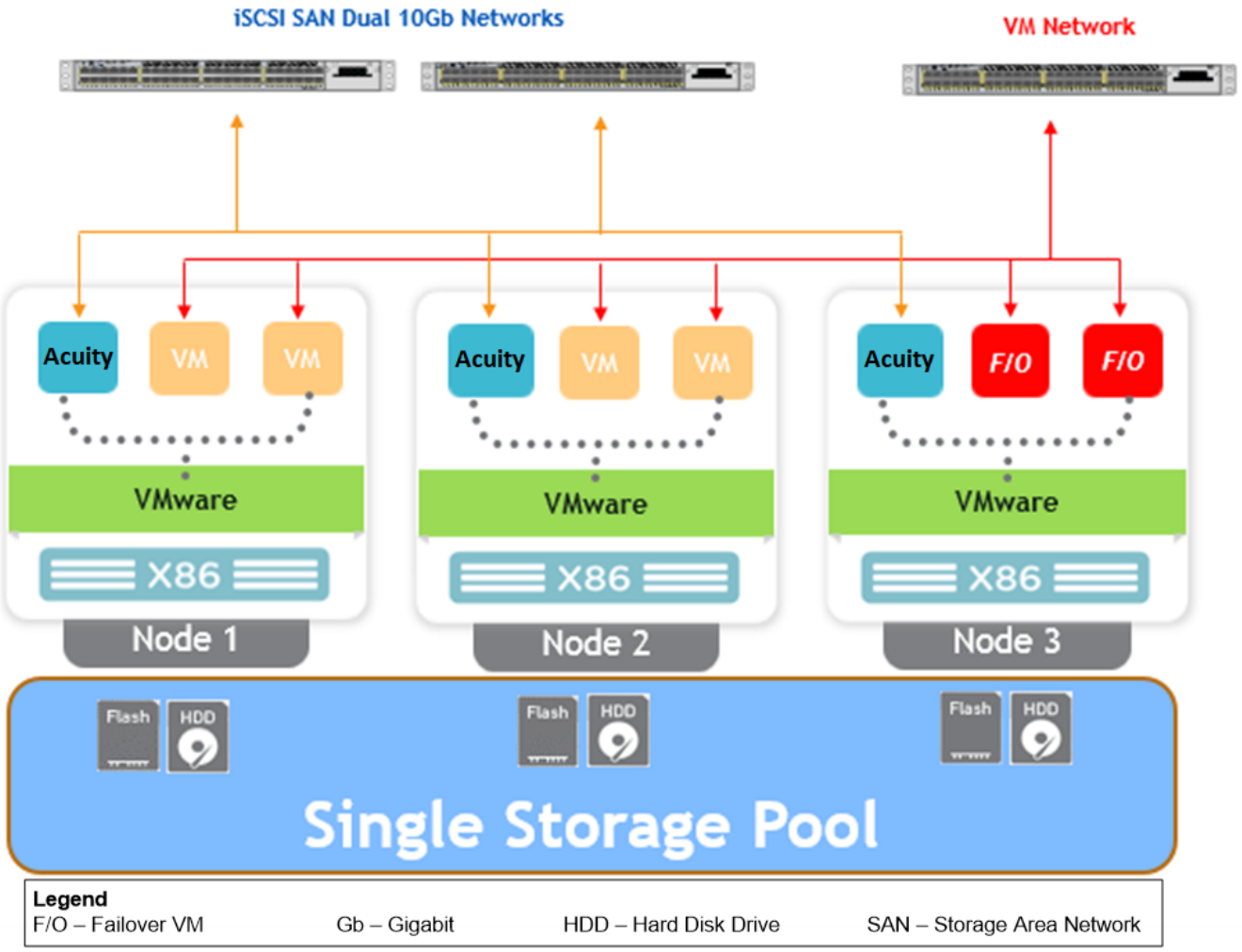

Figure 1 – Pivot3 Hyperconvergence Architecture

Pivot3 differentiates Acuity from competitors' products by pooling together all compute, RAM[6], cache, performance, and storage resources into one shared infrastructure. The shared storage is distributed across the entire vPG, and each VM is able to utilize the performance and capacity of the entire infrastructure. VMs never have to move data just because a VM was moved on the hypervisor. By leveraging the resources of the entire vPG, performance improves for the entire infrastructure, rather than just where the fastest disks or processors reside. This increases the value of Acuity's scalability by allowing customers to add only the resources they need and maximize the resulting performance improvements.

Integration with NVMe[7] PCIe[8] flash cards in the multi-tier architecture gives Acuity better I/O[9] request performance than conventional DHCI by using a persistent storage and read/write cache areas in the NVMe flash. NVMe flash is logically divided into two cache areas: a read/write mirrored journal split between the two

[6] RAM – Random Access Memory
[7] NVMe – Non-Volatile Memory Express
[8] PCIe – Peripheral Component Interconnect Express
[9] I/O – Input/Output

accelerator nodes and a read cache area unique to active volumes owned by an accelerator node. Prioritized by QoS[10], the I/O request performance becomes even faster for high priority VMs. The two NVMe PCIe flash cards in the accelerator nodes operates in "Active/Active" mode[11], ensuring maximum performance is provided to the VMs.

Acuity leverages SEC, which uses advanced mathematical formulas to distribute data across all drives and nodes within the storage system, to protect the vPG from data loss. SEC allows the vPG to recover from catastrophic hardware failures exceeding the limitations of traditional RAID techniques. Acuity is configured with SEC level 3 for data resiliency, which allows up to three disks or one node plus one disk can fail simultaneously without the vPG experiencing any loss of data.

The system uses a dual 10 Gb iSCSI SAN as the backbone for node and VM communications in the vPG. Network equipment is redundant, providing multiple I/O paths, and scales with the size of the vPG. Since redundant equipment operates in "Active/Active" mode[12], maximum bandwidth is provided for each VM's communications. Network communications are handled peer-to-peer across nodes, which means there is no "master node" to bottleneck traffic.

Acuity is managed through the vCenter Plugin. Administrators can perform tasks such as creating and managing vPGs, provisioning and managing storage, setting permissions, defining QoS metrics, updating the Acuity Software Platform, and configuring network settings from within the vSphere Web Client. Additionally, the vCenter Plugin provides a dashboard that allows administrators to monitor performance, health, capacity, events, and tasks. Acuity leverages vCenter to provide authentication, auditing, and some user management capabilities.

1.4 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. This section provides a context for the TOE evaluation by identifying the TOE type, describing the TOE, and defining the specific evaluated configuration.

The TOE is Acuity, which is a software and hardware TOE based on a heavily-modified version of Linux. The TOE boundary includes the Acuity Software Platform running within a VM on the ESXi hypervisor in the TOE environment. The TOE boundary also includes the Pivot3 vCenter Plugin, which runs on the vCenter Server, as well as a specialized Pivot3 USB Key, which provides the virtual data store for the Acuity Software Platform. The storage, performance, and VM management capabilities mentioned above are also included in the TOE. The TOE also controls access to storage through iSCSI attributes and permissions. The data resiliency provided by SEC and other high availability features are contained within the TOE boundary. The TOE can perform user authentication on results passed to it from vCenter, although vCenter's SSO capabilities are not included within the TOE boundary. The TOE also synchronizes with the system time of the ESXi host to provide reliable time stamps for audit records that it generates.

[10] QoS – Quality of Service
[11] Active/Active for the NVMe PCIe accelerator cards is a high availability mode where both cards actively process I/O requests during normal operation.
[12] Active/Active for the networking equipment is a high availability mode where all nodes act as backup nodes and actively process I/O requests during normal operation.

In the evaluated configuration, the TOE includes three Acuity nodes. Figure 2 shows the details of the deployment configuration of the TOE.

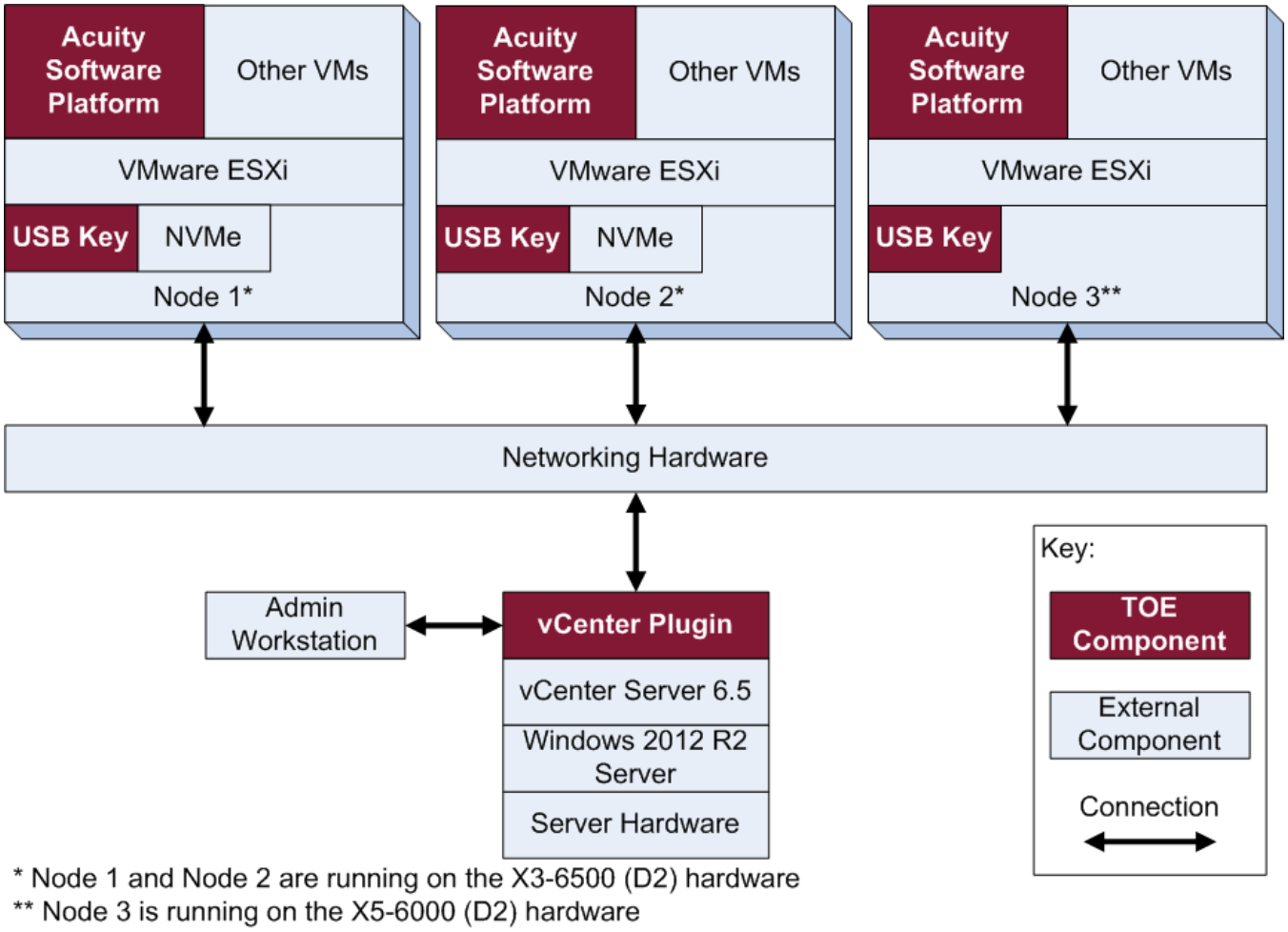

Figure 2 – Deployment Configuration of the TOE

1.4.1 TOE Components

The TOE consists primarily of three components:

- Pivot3 Acuity Software Platform – This is the core component of the TOE, providing all major storage management functionality for the system. This also includes the underlying operating system, which provides basic functionality such as system time, networking, and basic storage.
- Pivot3 vCenter Plugin – This is a Java-based service and GUI[13] plugin for the vSphere Web Client that provides the management interface for the TOE.
- Pivot3 USB Key – This is a hardware USB Key that provides the ESXi datastore for the Acuity VM files. This allows the VM to be stored outside of the vPG storage, maximizing the space available to users.

[13] GUI – Graphical User Interface

1.5 TOE Environment

The TOE makes use of the VMware ESXi v6.5.0 hypervisor to provide virtualization functionality to run VMs. The ESXi hypervisor is installed on two X3-6500 (D2) servers and one X5-6000 (D2) server. Each of the X3-6500 (D2) servers have an NVMe PCIe flash card to provide accelerated data transfer. The TOE also uses VMware vCenter v6.5.0 to provide several management functions that include creating and deleting users, reviewing the audit log, and evaluating credentials passed to the system during authentication. It uses the vSphere API[14] (provided through the vSphere Web Services SDK[15]) to allow management interactions with the vCenter server. TOE administrator accounts are local to the vCenter server.

All the compute, RAM, storage, and networking hardware resources are outside the boundary of the TOE and therefore a part of the TOE environment. The TOE manages these entities via software mechanisms such as access control metadata on iSCSI volumes.

The TOE is capable of scaling linearly with other instances of the TOE (other Acuity nodes), but additional nodes are a part of the TOE environment. The evaluated configuration includes a total of three Acuity nodes.

The networking hardware required to run the TOE includes two redundant Enterprise-class 10 Gb Ethernet switches configured with isolated networks as follows:

- General purpose VM network
- Storage network accessible to Acuity and vCenter
- Management network accessible to Acuity, vCenter, and ESXi

In the evaluated configuration, guest VMs consuming the environment's storage also utilize the hardware of the Acuity nodes and thus are installed in the VMware datastore provided by the nodes.

1.6 TOE Description

This section primarily addresses the physical and logical components of the TOE that are included in the evaluation.

1.6.1 Physical Scope

Figure 2 above illustrates the physical scope and the physical boundary of the overall solution and ties together the components of the TOE.

The TOE is a software and hardware TOE that contains a full VM as well as a management plugin for vCenter and a proprietary USB Key. All the TOE components run on the X3-6500 (D2) and X5-6000 (D2) server platforms in a three-node configuration. The TOE components are the same as those specified in section 1.4.1, and consist of the Acuity Software Platform (including a Linux-based kernel), vCenter Plugin, and USB Key. The TOE boundary specifically does not contain any of the third-party software that the TOE relies upon in its environment as described in section 1.5.

[14] API – Application Programming Interface
[15] SDK – Software Development Kit

1.6.1.1 TOE Software/Hardware

The TOE components are uniquely identified and formatted as follows:

- Pivot3 Acuity Software Platform – With reference pvt00.10.4.0.082E in a virtual disk image formatted for VMWare
- Pivot3 vCenter Plugin – With reference v10.4.0.2207 as a *.EXE formatted file
- Pivot3 USB Key – With reference [SERIAL] where "SERIAL" is a unique multi-character serial number as a hardware USB Key

The TOE software is packaged on the Pivot3 USB Key and is delivered to customers along with the Pivot3 node hardware, which is a required part of the TOE environment.

1.6.1.2 Guidance Documentation

The following PDF[16] formatted guides, which are available for download through the Pivot3 support portal after a customer account is created, are required reading and part of the TOE:

- *Pivot3 Acuity 10.4 Setup & User Guide;* Document Version 1.1; May 25, 2018
- *Pivot3, Inc. Acuity 10.4 Guidance Documentation Supplement; Evaluation Assurance Level (EAL): EAL2+;* Document Version: v0.3

1.6.2 Logical Scope

The logical boundary of the TOE will be broken down into the following security classes, which are further described in sections 6 and 7 of this ST. The logical scope also provides the description of the security features of the TOE. The SFRs implemented by the TOE are usefully grouped under the following Security Function Classes:

- Security Audit
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the TSF
- Resource Utilization
- Extended High Availability Functionality

1.6.2.1 Security Audit

The TOE provides the capability to generate audit records for startup and shutdown of the TOE, as well as configuration changes. Since the TOE operates in a distributed configuration, startup and shutdown of each individual node is recorded in the audit records for each node. The TOE also generates audit records to detect card, node, and disk hardware failures.

1.6.2.2 User Data Protection

The TOE provides access controls that limit the ability to access VM storage to authorized ESXi hosts. If a simultaneous failure of up to three drives or one node plus one drive occurs, the TOE detects the failure and rebuilds the data. This is done by leveraging SEC techniques. The TOE also provides access controls that permit

[16] PDF – Portable Document Format

rollback functionality for modified or deleted files, folders, volumes, or VMs that are stored within a vPG. This is limited to the retained snapshots created using the snapshot feature.

1.6.2.3 Identification and Authentication

The TOE verifies that administrators have been identified before granting them access to any TOE management functionality.

1.6.2.4 Security Management

Administrators manage the TOE via the vCenter Plugin. This is the only management interface provided to administrators. Through this interface, administrators can configure the various access controls and configuration settings. A dashboard that can be used to get at-a-glance information about the status of the TOE is also available.

1.6.2.5 Protection of the TSF

The TOE implements SEC to ensure the continued secure operation of its storage array if a simultaneous failure of up to three disks or one node plus one disk occurs. Data consistency is guaranteed when the data is replicated between parts of the TOE. The TOE also utilizes redundant NVMe PCIe flash cards in the environment and ensures continued operation is one of the cards fails.

The TOE provides reliable time stamps for its own use.

1.6.2.6 Resource Utilization

The TOE ensures that simultaneous failure of up to three disks or one node plus one disk does not result in loss of user data. Storage will continue to operate with no interruption of service in the event of such failures. The TOE also ensures that failure of one active NVMe PCIe flash card does not result in loss of user data. I/O requests will continue to be processes by the remaining card.

The TOE requires a QoS policy to be selected when a volume is created. This policy assigns the priority for each volume and controls the access to all sharable resources in the environment.

1.6.2.7 Extended High Availability Functionality

The TOE performs self-tests to determine when disk or node failures occur and checks to ensure that data remains in an unmodified state. SEC techniques allow the TOE to monitor data and rebuild data when data loss occurs. The TOE also performs a self-test to determine connectivity between the accelerator nodes to determine if one of the NVMe PCIe flash cards has failed. If one of the cards fails, all I/O requests are directed to the remaining accelerator node.

2. Conformance Claims

This section and Table 2 provide the identification for any CC, PP, and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in section 8.1.

Table 2 – CC and PP Conformance

Common Criteria (CC) Identification and Conformance	*Common Criteria for Information Technology Security Evaluation*, Version 3.1, Release 5, April 2017; CC Part 2 extended; CC Part 3 conformant; PP claim (none); Parts 2 and 3 Interpretations of the Common Evaluation Methodology (CEM) as of 5/16/2018 were reviewed, and no interpretations apply to the claims made in this ST.
PP Identification	None
Evaluation Assurance Level	EAL2+ augmented with Flaw Remediation Procedures (ALC_FLR.2)

3. Security Problem

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It provides the statement of the TOE security environment, which identifies and explains all:

- Known and presumed threats countered by either the TOE or by the security environment
- Organizational security policies to which the TOE must comply
- Assumptions about the secure usage of the TOE, including physical, personnel, and connectivity aspects

3.1 Threats to Security

This section identifies the threats to the IT[17] assets against which protection is required by the TOE or by the security environment. The threat agents are divided into two categories:

- Attackers who are not TOE users: They have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters, and no physical access to the TOE.
- TOE users: They have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters, and physical access to the TOE. (TOE users are, however, assumed not to be willfully hostile to the TOE.)

Both are assumed to have a low level of motivation. The IT assets requiring protection are the TSF[18] and user data stored on the TOE and transmitted to hosts on the protected network. Both the confidentiality and integrity of the data must be protected. Removal, diminution, and mitigation of the threats are through the objectives identified in section 4, Security Objectives. Table 3 below lists the applicable threats.

Table 3 – Threats

Name	Description
T.DATA_CORRUPTION	Data could become corrupted or TOE security compromised due to hardware failure.
T.UNAUTH	An unprivileged user may gain access to TSF data on the TOE, even though the user is not authorized in accordance with the TOE security policy.
T.UNINTENDED_ACCESS	An attacker and/or user of the TOE functionality could access VM storage they are not authorized to access.

3.2 Organizational Security Policies

An Organizational Security Policy (OSP) is a set of security rules, procedures, or guidelines imposed by an organization on the operational environment of the TOE. Table 4 below lists the OSPs that are presumed to be

[17] IT – Information Technology
[18] TSF – TOE Security Functionality

imposed upon the TOE or its operational environment by any organization implementing the TOE in the CC evaluated configuration.

Table 4 – Organizational Security Policies

Name	Description
P.ACCESS	Organizations deploying the TOE must ensure that a defense-in-depth strategy is used to further control user access to TOE resources by applying the appropriate management and user access control policies for the TOE environment.

3.3 Assumptions

This section describes the security aspects of the intended environment for the evaluated TOE. The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user guidance. Table 5 lists the specific conditions that are required to ensure the security of the TOE and are assumed to exist in an environment where the TOE is employed.

Table 5 – Assumptions

Name	Description
A.ADMIN_AUTH	The TOE environment provides a secure repository of users that are authorized to manage the TOE.
A.ADMIN_PROTECT	The TOE environment provides the workstation used to manage the TOE that is free of malicious software.
A.LOCATE	The physical environment must be suitable for supporting a computing device in a secure setting.
A.MANAGE	There are one or more competent individuals assigned to manage the TOE and the security of the information it contains.
A.NETWORK	The TOE environment provides the network infrastructure required for its operation that is appropriately secured and protected from interference or tampering.
A.NOEVIL	The administrators who manage the TOE are non-hostile, appropriately trained, and follow all guidance.
A.PLATFORM	The TOE environment provides the hardware and hypervisor necessary for the operation of the TOE.
A.PROTECT	The TOE software will be protected from unauthorized modification.

4. Security Objectives

Security objectives are concise, abstract statements of the intended solution to the problem defined by the security problem definition (see section 3). The set of security objectives for a TOE form a high-level solution to the security problem. This high-level solution is divided into two part-wise solutions: the security objectives for the TOE and the security objectives for the TOE's operational environment. This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security Objectives for the TOE

The specific security objectives for the TOE are listed in Table 6.

Table 6 – Security Objectives for the TOE

Name	Description
O.ACCESS	The TOE must implement rules to govern access to stored user data and system resources.
O.ADMIN	The TOE must include a set of functions that allow efficient management of its functions, security attributes, and TSF data, ensuring that only authorized TOE administrators (in defined roles) may exercise such control.
O.AUDIT	The TOE must record security relevant events and (if applicable) associate each event with the identity of the administrator that caused the event.
O.IDENTIFICATION	The TOE must be able to identify administrators using the authentication mechanisms in the TOE environment prior to allowing any access to TOE administrative functions and TSF data. An administrator's username must be associated with every management action.
O.TSF_PROTECT	The TOE must protect its functions and TSF data to ensure its security policies are enforced and capabilities are intact when drive or node failures occur. It also must provide for the ability to check that its nodes are operating correctly.
O.USER_DATA_PROTECT	The TOE must ensure the integrity of stored user data by monitoring for errors and automatically correcting them or providing functionality to restore user data from a snapshot.

4.2 Security Objectives for the Operational Environment

This section describes the environmental objectives.

4.2.1 IT Security Objectives

Table 7 lists the IT security objectives that are to be satisfied by the environment.

Table 7 – IT Security Objectives

Name	Description
OE.ACCESS	The TOE environment must implement access control mechanisms to protect the confidentiality of TOE user data accessed by users.
OE.ADMIN_PROTECT	The administrator workstation must be protected from any external interference or tampering.

Name	Description
OE.AUTH	The TOE environment must provide a secure repository of user accounts used to manage the TOE.
OE.NETWORK	The TOE environment must be implemented such that the TOE is appropriately secured and protected from interference or tampering.
OE.PLATFORM	The TOE hardware and hypervisor must support all required TOE functions.
OE.PROTECT	The TOE environment must protect itself and the TOE from external interference or tampering.

4.2.2 Non-IT Security Objectives

Table 8 lists the non-IT environment security objectives that are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through the application of procedural or administrative measures.

Table 8 – Non-IT Security Objectives

Name	Description
NOE.MANAGE	Sites deploying the TOE will provide competent, non-hostile TOE administrators who are appropriately trained and follow all administrator guidance. TOE administrators will ensure the system is used securely.
NOE.PHYSICAL	The TOE and the environment upon which it relies is located within a controlled access facility.

5. Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in section 6.1.

5.1 Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE. The extended SFRs are organized by class. Table 9 identifies all extended SFRs implemented by the TOE.

Table 9 – Extended TOE Security Functional Requirements

Name	Description
EXT_FHA_TST.1	Component testing

5.1.1 Class EXT_FHA: Extended High Availability

Extended High Availability ensures that the TOE provides high availability capabilities to minimize the downtime experienced in the event of an error. The EXT_FHA: Extended High Availability function class was modeled after the CC FPT: Protection of the TSF class. The extended family FHA_TST: Component Testing was modeled after the CC family FPT_TST: TSF self-test.

Figure 3 – EXT_FHA: Extended High Availability Class Decomposition

5.1.1.1 Component Testing (EXT_FHA_TST)

Family Behavior

This family defines the requirements for high availability tests that should be available to assist in determining if there's been an error that hampers the proper functioning of the TOE.

Component Leveling

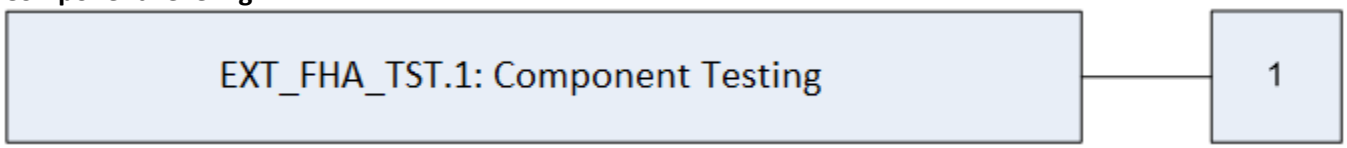

Figure 4 – EXT_FHA Component Testing family decomposition

EXT_FHA_TST.1 Component Testing provides the capability for the TOE to perform tests to ensure its proper function.

Management: EXT_FHA_TST.1
The following actions could be considered for the management functions in FMT:

- Management of the high availability settings for the TOE.

Audit: EXT_FHA_TST.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

- Basic: Execution of the tests and the results of the tests.

EXT_FHA_TST.1 Component testing
Hierarchical to: No other components
EXT_FHA_TST.1.1
The TSF shall run a suite of self tests [selection: during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions [assignment: *conditions under which self test should occur*]] to demonstrate the correct operation of [assignment: *functionality covered by self tests*].
Dependencies: No dependencies

5.2 Extended TOE Security Assurance Components

There are no extended TOE security assurance components defined for this evaluation.

6. Security Requirements

This section defines the SFRs and SARs met by the TOE. These requirements are presented following the conventions identified in Section 6.1.

6.1 Conventions

There are several font variations used within this ST. Selected presentation choices are discussed here to aid the Security Target reader.

The CC allows for assignment, refinement, selection, and iteration operations to be performed on security functional requirements. All of these operations are used within this ST. These operations are performed as described in Part 2 of the CC and are shown as follows:

- Completed assignment statements are identified using [*italicized text within brackets*].
- Completed selection statements are identified using [*underlined text within brackets*].
- Refinements are identified using **bold text**. Any text removed is stricken (Example: ~~TSF Data~~) and should be considered as a refinement.
- Extended Functional and Assurance Requirements are identified using "EXT_" at the beginning of the short name.
- Iterations are identified by appending a letter in parentheses following the component title. For example, FAU_GEN.1(a) Audit Data Generation would be the first iteration and FAU_GEN.1(b) Audit Data Generation would be the second iteration.

6.2 Security Functional Requirements

This section specifies the SFRs for the TOE. This section organizes the SFRs by CC class. Table 10 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement.

Table 10 – TOE Security Functional Requirements

Name	Description	S	A	R	I
FAU_GEN.1	Audit data generation	✓	✓		
FDP_ACC.1	Subset access control		✓		
FDP_ACF.1	Security attribute based access control		✓		
FDP_ROL.1	Basic rollback		✓		
FDP_SDI.2	Stored data integrity monitoring and action		✓		
FIA_UID.2	User identification before any action				
FMT_MSA.1	Management of security attributes	✓	✓		
FMT_MSA.3	Static attribute initialization	✓	✓		
FMT_MTD.1	Management of TSF data	✓	✓		

Name	Description	S	A	R	I
FMT_SMF.1	Specification of management functions		✓		
FMT_SMR.1	Security roles		✓		
FPT_FLS.1(a)	Failure with preservation of secure state (for disks)		✓		✓
FPT_FLS.1(b)	Failure with preservation of secure state (for NVMe cards)		✓		✓
FPT_STM.1	Reliable time stamps				
FRU_FLT.1(a)	Degraded fault tolerance for disks		✓		✓
FRU_FLT.1(b)	Degraded fault tolerance for cards		✓		✓
FRU_PRS.2	Full priority of service				
EXT_FHA_TST.1	Component testing	✓	✓		

Note: S=Selection; A=Assignment; R=Refinement; I=Iteration

6.2.1 Class FAU: Security Audit

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. [*Operations performed through the vCenter Plugin including:*

   i. *Modifying vPGs*
   ii. *Creating/expanding volumes*
   iii. *Mounting of a volume's snapshot*
   iv. *Setting QoS parameters*
   v. *Setting access control lists*
   vi. *Setting CHAP[19] secrets*
   vii. *Applying Acuity Software Platform updates*

d. *Events captured in the quick diagnostic logs including:*

   i. *Drive failures*
   ii. *Volume failures*
   iii. *Node failures*
   iv. *NVMe PCIe flash card failures*
   v. *Node reboot*
   vi. *Node shutdown*
   vii. *Node failure to answer heartbeat*
   viii. *iSCSI initiator login/logout*].

[19] CHAP – Challenge Handshake Authentication Protocol

FAU_GEN.1.2

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

a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*no other audit relevant information*].

6.2.2 Class FDP: User Data Protection

FDP_ACC.1 Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1

The TSF shall enforce the [*Virtual Storage Access Control SFP*[20]] on [

- *Subjects:*
  i. *ESXi host where the Acuity Software Platform VM is running*
- *Objects:*
  i. *Storage volumes*
- *Operations:*
  i. *Read*
  ii. *Write*

].

FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

FDP_ACF.1.1

The TSF shall enforce the [*Virtual Storage Access Control SFP*] to objects based on the following: [

- *Subject Attributes:*
  i. *iSCSI host IQN (iSCSI Qualified Name)*
  ii. *Client CHAP secret*
- *Object Attributes:*
  i. *Logical volume identifier*
  ii. *Server CHAP secret*

].

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- *If the iSCSI host IQN associated with an initiator is set to "None", access is denied.*

[20] SFP – Security Functional Policy

- *If the iSCSI host IQN associated with an initiator is set to "Read/Write", then the initiator is granted read/write access.*
- *The initiator and target must validate each other's CHAP secrets for the connection to be established and access to be granted to storage. If either of these checks fails, access is denied.*

].

FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [*no additional rules*].

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the [*no additional rules*].

Application note: *In iSCSI terminology, the initiator is the host or client that is accessing the storage target. Within the context of the TOE, this could mean any host that is granted access to the data, whether it is an Acuity node or generic iSCSI client residing on the Acuity node hardware. It is the responsibility of the TOE administrator to configure access control to iSCSI initiators in accordance with their organizational security policy.*

FDP_ROL.1 Basic rollback

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]

FDP_ROL.1.1

The TSF shall enforce [*Virtual Storage Access Control SFP*] to permit the rollback of the [*modification or deletion*] on the [*files, folders, volumes, or VMs within a vPG*].

FDP_ROL.1.2

The TSF shall permit operations to be rolled back within the [*snapshots created for a particular storage volume*].

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies

FDP_SDI.2.1

The TSF shall monitor user data stored in containers controlled by the TSF for [*simultaneous node and disk failures not exceeding three disk failures or one node failures plus one disk failure*] on all objects, based on the following attributes: [*wide striping SEC parity data*].

FDP_SDI.2.2

Upon detection of a data integrity error, the TSF shall [*rebuild the data*].

6.2.3 Class FIA: Identification and Authentication

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies

FIA_UID.2.1

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

6.2.4 Class FMT: Security Management

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control or
FDP_IFC.1 Subset information flow control]
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.1.1

The TSF shall enforce the [*Virtual Storage Access Control SFP*] to restrict the ability to [*query, modify*] the security attributes [*access control lists and CHAP secrets*] to [*SuperUser Administrators or Users who have been assigned permissions to modify the host access groups for a vPG*].

FMT_MSA.3 Static attribute initialization

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3.1

The TSF shall enforce the [*Virtual Storage Access Control SFP*] to provide [*restrictive*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The TSF shall allow the [*SuperUser Administrator*] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

Dependencies: FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MTD.1.1

The TSF shall restrict the ability to [*query, modify, delete, [create]*] the [*vPGs, volumes, QoS parameters, host access groups, and CHAP secrets*] to [*SuperUser Administrators. Read Only users may query TSF data but no other actions are allowed. Users may only query/modify vPG settings for the vPG to which they are assigned*].

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: [

- *Create/modify vPGs*
- *Create/expand volumes*
- *Create/expand datastores*
- *Create/mount snapshots*
- *Rollback/restore files, folders, volumes, or VMs from a snapshot*
- *Set QoS parameters*
- *Set access control lists*

- *Setting CHAP secrets*
- *Perform Acuity Software Platform updates*

].

FMT_SMR.1 Security roles
Hierarchical to: No other components.
Dependencies: FIA_UID.1 Timing of identification
FMT_SMR.1.1
The TSF shall maintain the roles [*User, SuperUser Administrator, Read Only*].
FMT_SMR.1.2
The TSF shall be able to associate users with roles.

6.2.5 Class FPT: Protection of the TSF

FPT_FLS.1(a) Failure with preservation of secure state (for disks)
Hierarchical to: No other components.
Dependencies: No dependencies.
FPT_FLS.1(a).1
The TSF shall preserve a secure state when the following types of failures occur: [*simultaneous failure of up to three disks or one complete node plus one disk*].

FPT_FLS.1(b) Failure with preservation of secure state (for NVMe cards)
Hierarchical to: No other components.
Dependencies: No dependencies.
FPT_FLS.1(b).1
The TSF shall preserve a secure state when the following types of failures occur: [*failure of one active NVMe PCIe flash card*].

FPT_STM.1 Reliable time stamps
Hierarchical to: No other components.
Dependencies: No dependencies
FPT_STM.1.1
The TSF shall be able to provide reliable time stamps.

6.2.6 Class FRU: Resource Utilization

FRU_FLT.1(a) Degraded fault tolerance for disks
Hierarchical to: No other components.
Dependencies: FPT_FLS.1(a) Failure with preservation of secure state (for disks)
FRU_FLT.1(a).1
The TSF shall ensure the operation of [*the storage arrays*] when the following failures occur: [*simultaneous failure of up to three disks or one node plus one disk*].

FRU_FLT.1(b) Degraded fault tolerance for cards
Hierarchical to: No other components.
Dependencies: FPT_FLS.1(b) Failure with preservation of secure state (for NVMe cards)

FRU_FLT.1(b).1

The TSF shall ensure the operation of [*I/O requests*] when the following failures occur: [*failure of one active NVMe PCIe flash card*].

FRU_PRS.2 Full priority of service

Hierarchical to: FRU_PRS.1 Limited priority service

Dependencies: No dependencies

FRU_PRS.2.1

The TSF shall assign a priority to each subject in the TSF.

FRU_PRS.2.2

The TSF shall ensure that each access to all shareable resources shall be mediated on the basis of the subject's assigned priority.

6.2.7 Class EXT_FHA: Extended High Availability Functionality

EXT_FHA_TST.1 Component testing

Hierarchical to: No other components

Dependencies: No dependencies

EXT_FHA_TST.1.1

The TSF shall run a suite of self tests [during initial start-up, periodically during normal operation] to demonstrate the correct operation of [*node connectivity and accelerator node availability*].

6.3 Security Assurance Requirements

This section defines the assurance requirements for the TOE. Assurance requirements are taken from the CC Part 3 and are EAL2+ augmented with ALC_FLR.2. Table 11 summarizes these requirements.

Table 11 – Assurance Requirements

Assurance Requirements	
Class ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
Class ALC: Life Cycle Support	ALC_CMC.2 Use of a CM[21] system
	ALC_CMS.2 Parts of the TOE CM Coverage
	ALC_DEL.1 Delivery Procedures
	ALC_FLR.2 Flaw Reporting Procedures

[21] CM – Configuration Management

Assurance Requirements	
Class ADV: Development	ADV_ARC.1 Security Architecture Description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
Class AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
Class ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing – Sample
Class AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

7. TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

7.1 TOE Security Functionality

Each of the security requirements and the associated descriptions correspond to a security functionality. Hence, each security functionality is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functionality and rationalize that the security functionality satisfies the necessary requirements. Table 12 lists the security functionality and their associated SFRs.

Table 12 – Mapping of TOE Security Functionality to Security Functional Requirements

TOE Security Functionality	SFR ID[22]	Description
Security Audit	FAU_GEN.1	Audit data generation
User Data Protection	FDP_ACC.1	Subset access control
	FDP_ACF.1	Security attribute based access control
	FDP_ROL.1	Basic rollback
	FDP_SDI.2	Stored data integrity monitoring and action
Identification and Authentication	FIA_UID.2	User identification before any action
Security Management	FMT_MSA.1	Management of security attributes
	FMT_MSA.3	Static attribute initialization
	FMT_MTD.1	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
Protection of TOE Security Functionality	FPT_FLS.1(a)	Failure with preservation of secure state (for disks)
	FPT_FLS.1(b)	Failure with preservation of secure state (for NVMe cards)
	FPT_STM.1	Reliable time stamps
Resource Utilization	FRU_FLT.1(a)	Degraded fault tolerance for disks
	FRU_FLT.1(b)	Degraded fault tolerance for cards
	FRU_PRS.2	Full priority of service
Extended High Availability Functionality	EXT_FHA_TST.1	Component testing

[22] ID – Identification

7.1.1 Security Audit

The TOE generates audit records for startup and shutdown events for the TOE as well as drive failures, volume failures, node failures, NVMe PCIe flash card failures, node reboots, node shutdown events, and nodes failing to answer heartbeat messages. These audit records can be viewed in the Quick Diagnostic Log, which is viewable through the vCenter Plugin. The Quick Diagnostic Log audit records contain the following information:

Table 13 – Quick Diagnostic Log Audit Record Contents

Field	Content
Timestamp	Contains the date and time the event occurred.
Scope	Identifies the areas of the TOE affected by the event.
Severity	Identifies the degree of event severity to be Major, Minor, and Informational.
Boot#	Identifies the boot that the event occurred in with a number.
Sub System	Identifies the sub system that detected or originated the event.
Summary	Contains detailed information about the event as well as the outcome of the event.

In addition, the audit records can also be viewed from the Task Console, which records all management operations performed via the vCenter Plugin. The details recorded in the Task Console include the following information:

Table 14 – Task Console Contents

Field	Content
Task Name	Identifies the task that is being performed.
Target	Identifies the IP address of the system that the task applies to.
Status	Identifies the current status of the task.
Details	Provides more details about the task if applicable.
Initiator	Identifies the user that started the task.
Queued For	Records the time the task waited for the system to perform it.
Start Time	Records the timestamp for when the command was started.
Completion Time	Records the timestamp for when the command was completed.
Execution Time	Records the time it took the system to complete the task.
Server	Identifies the IP address of the server that the task originated from.

TOE Security Functional Requirements Satisfied: FAU_GEN.1.

7.1.2 User Data Protection

The TOE provides iSCSI-based access controls that permit operations between the ESXi host and logical volumes. Access control lists (ACLs) are based on the host access group's comprising initiator and target identifiers, and mutual CHAP authentication secrets. Host access groups are stored in the TOE's configuration and require the client (ESXi) iSCSI host IQN to have a valid access entry for the storage volume for access to be permitted. The access may be set to "None" or "Read/Write". Additionally, the client must pass a valid CHAP secret that can be

validated by the server, and the server must also pass a valid CHAP secret that can be validated by the client. If either secret fails validation, access is not permitted.

The TOE uses SEC techniques to provide data resiliency for the vPG. This includes monitoring for hardware failures and data loss. Part of the SEC technique includes striping data across the entire vPG, providing high availability services for user data. This provides the TOE with the capability to rebuild lost data if the simultaneous data loss does not exceed three failed disks or one failed node plus one failed disk.

The TOE provides rollback functionality for modified or deleted files, folders, volumes, or VMs that are stored within a vPG. This is limited to the snapshots created using the snapshot feature. Once a snapshot has been captured, it can be mounted as a cloned volume of the original. This allows administrators to rollback modified or deleted files, folders, volumes, or VMs by restoring them from the cloned volume to the active volume.

TOE Security Functional Requirements Satisfied: FDP_ACC.1, FDP_ACF.1, FDP_ROL.1, FDP_SDI.2.

7.1.3 Identification and Authentication

The users of the TOE are authenticated by the underlying vCenter Server before access to the TOE is allowed. After the vCenter Plugin is invoked, it uses the user's vCenter user account ID (vCenter username) and role for identification and access control purposes. The TOE only permits access to functionality if a user has been successfully identified and they carry the required role permission.

TOE Security Functional Requirements Satisfied: FIA_UID.2.

7.1.4 Security Management

The TOE is managed through the vSphere Web Client via the vCenter Plugin. This plugin provides the only administrative interface for the TOE. Management activities that can be performed on this interface include the following:

- Creating and modifying vPGs
- Creating and expanding volumes
- Creating and expanding datastores
- Creating and mounting snapshots
- Restoring or rolling back a deleted or modified file, folders, volumes, or VMs from a snapshot
- Setting QoS parameters
- Setting access control lists
- Configuring CHAP secrets
- Performing Acuity Software Platform updates

The vCenter Plugin also provides a series of screens and interfaces that allow administrators to check the health, status, and utilization of the vPG. Three roles are available for managing the TOE: User, SuperUser Administrator, and Read Only. The SuperUser Administrator role has unrestricted access to all management activities for the entire vPG. The User role can access functionality only for vPGs and functionality that has been explicitly granted to the user by a SuperUser Administrator. The Read Only role provides the ability to query the TOE configuration

but cannot perform any modifications. SuperUser and Read Only roles are fixed roles. The User role is based on an association with specific vPGs set by an administrator.

By default, all hosts have no access to provisioned storage in the global storage array until configured otherwise by an authorized user.

TOE Security Functional Requirements Satisfied: FMT_MSA.1, FMT_MSA.3, FMT_MTD.1, FMT_SMF.1, FMT_SMR.1.

7.1.5 Protection of the TSF

The TOE will continue to operate without any affect to its security configuration when a simultaneous failure of up to three disks or one node plus one disk occurs. The system uses SEC to detect hardware (disk and node) failures and can rebuild lost data without any interruption to the availability of the global storage array. The TOE will also continue to operate without any affect to its security configuration when one of the active NVMe PCIe flash card fails. The environment contains two active NVMe PCIe flash cards, and when one of them fails, the TOE will control the I/O requests and route them to the remaining card.

The TOE relies on the ESXi host to synchronize time during the boot process and once per hour. Once received, the time is propagated to all nodes in the vPG. In order to simplify deployments with multiple ESXi hosts, one of the ESXi instances is designated the “timekeeper” and used by all nodes to synchronize time across the vPG.

TOE Security Functional Requirements Satisfied: FPT_FLS.1(a), FPT_FLS.1(b), FPT_STM.1.

7.1.6 Resource Utilization

When a simultaneous failure of up to three disks or one node plus one disk occurs, the TOE will rebuild the lost data using SEC. During the data rebuilding process, VMs are still able to access their data, and once the data is rebuilt, the TOE will repeat the process once the failed hardware is repaired or replaced. When one of the active NVMe PCIe flash cards fail, the TOE will route all I/O requests to the remaining card to ensure uninterrupted operation of the TOE continues.

The TOE implements five default QoS performance policies. When a new volume is created, one of the QoS performance policies must be selected to control the priority that is assigned to that volume. These policies are used by the TOE to ensure volumes are given the correct priority to all of the system resources stored in the environment.

TOE Security Functional Requirements Satisfied: FRU_FLT.1(a), FRU_FLT.1(b), FRU_PRS.2.

7.1.7 Extended High Availability Functionality

The TOE uses a heartbeat protocol to determine if a node or card failure has occurred. If a node fails to respond to the heartbeat protocol within a certain time interval, the node is considered to have failed and the data contained on the node is rebuilt across the remaining nodes in the vPG. The TOE also performs metadata consistency checks on a periodic basis in order to detect data corruption or disk failures. The TOE performs read

scrubbing[23] to determine if data can be read. If the read fails, then the TOE attempts to restore the lost data from SEC parity data. If a card fails and a host receives failures from I/O requests, the TOE will reissue the failed requests to the surviving accelerator node. Ownership of the volumes will be transferred to the surviving accelerator node until the failed accelerator node is back online.

TOE Security Functional Requirements Satisfied: EXT_FHA_TST.1.

[23] Read scrubbing consists of reading from each location in storage, correcting bit errors (if any), and writing the corrected data back to the same location.

8. Rationale

8.1 Conformance Claims Rationale

This Security Target extends Part 2 and conforms to Part 3 of the *Common Criteria for Information Technology Security Evaluation*, Version 3.1 Release 5.

8.2 Security Objectives Rationale

This section provides a rationale for the existence of each threat, policy statement, and assumption that compose the Security Target. Sections 8.2.1, 8.2.2, and 8.2.3 demonstrate the mappings between the threats, policies, and assumptions to the security objectives are complete. The following discussion provides detailed evidence of coverage for each threat, policy, and assumption.

8.2.1 Security Objectives Rationale Relating to Threats

Table 15 provides a mapping of the objectives to the threats they counter.

Table 15 – Threats: Objectives Mapping

Threats	Objectives	Rationale
T.DATA_CORRUPTION Data could become corrupted or TOE security compromised due to hardware failure.	O.TSF_PROTECT The TOE must protect its functions and TSF data to ensure its SFRs are enforced and capabilities are intact when drive or node failures occur. It also must provide for the ability to check that its nodes are operating correctly.	O.TSF_PROTECT mitigates this threat by providing mechanisms to protect the TSF data from unauthorized modification.
	O.USER_DATA_PROTECT The TOE must ensure the integrity of stored user data by monitoring for errors and automatically correcting them or providing functionality to restore user data from a snapshot.	O.USER_DATA_PROTECT mitigates this threat by monitoring user data for errors and providing a recovery feature.
	OE.PROTECT The TOE environment must protect itself and the TOE from external interference or tampering.	OE.PROTECT mitigates this threat by ensuring that the TOE is protected from external interference or tampering.
T.UNAUTH An unprivileged user may gain access to TSF data on the TOE, even though the user is not authorized in accordance with the TOE security policy.	O.ADMIN The TOE must include a set of functions that allow efficient management of its functions, security attributes, and TSF data, ensuring that only authorized TOE administrators (in defined roles) may exercise such control.	O.ADMIN mitigates this threat by ensuring that access to TOE security data is limited to those users with access to the management functions of the TOE.

Threats	Objectives	Rationale
	O.IDENTIFICATION The TOE must be able to identify administrators using the authentication mechanisms in the TOE environment prior to allowing any access to TOE administrative functions and TSF data. An administrator's username must be associated with every management action.	O.IDENTIFY mitigates this threat by ensuring that users are identified prior to gaining access to TOE security data.
	O.TSF_PROTECT The TOE must protect its functions and TSF data to ensure its security policies are enforced and capabilities are intact when drive or node failures occur. It also must provide for the ability to check that its nodes are operating correctly.	O.TSF_PROTECT mitigates this threat by ensuring continued operation of the TOE in a secure state in the event of hardware failures.
	OE.AUTH The TOE environment must provide a secure repository of user accounts used to manage the TOE.	OE.ADMIN_AUTH mitigates this threat by ensuring that the TOE environment provides a secure repository of users authorized to manage the TOE.
T.UNINTENDED_ACCESS An attacker and/or user of the TOE functionality could access VM storage they are not authorized to access.	O.ACCESS The TOE must implement rules to govern access to stored user data and system resources.	O.ACCESS mitigates this threat by ensuring only authorized hosts may obtain access to stored user data and system resources.
	O.ADMIN The TOE must include a set of functions that allow efficient management of its functions, security attributes, and TSF data, ensuring that only authorized TOE administrators (in defined roles) may exercise such control.	O.ADMIN mitigates this threat by ensuring that only authorized users have access to TOE security data and management functionality.
	O.AUDIT The TOE must record security relevant events and (if applicable) associate each event with the identity of the administrator that caused the event.	O.AUDIT mitigates this threat by ensuring that security relevant events that may indicate attempts to tamper with the TOE are recorded.
	O.TSF_PROTECT The TOE must protect its functions and TSF data to ensure its security policies are enforced and capabilities are intact when drive or node failures occur. It also must provide for the ability to check that its nodes are operating correctly.	O.TSF_PROTECT mitigates this threat by ensuring continued operation of TOE in a secure state in the event of hardware failures.
	OE.ACCESS The TOE environment must implement access control mechanisms to protect the confidentiality of TOE user data accessed by users.	OE.ACCESS mitigates this threat by placing additional access control measures on the hypervisor and virtual machines that provide end-users with access to TOE user data.

Every threat is mapped to one or more objectives in the table above. This complete mapping demonstrates that the defined security objectives counter all defined threats.

8.2.2 Security Objectives Rationale Relating to Policies

Table 16 below gives a mapping of policies and the objectives that support them.

Table 16 – Policies: Objectives Mapping

Policies	Objectives	Rationale
P.ACCESS Organizations deploying the TOE must ensure that a defense-in-depth strategy is used to further control user access to TOE resources by applying the appropriate management and user access control policies for the TOE environment.	O.ACCESS The TOE must implement rules to govern access to stored user data and system resources.	O.ACCESS upholds this policy by ensuring that the TOE enforces access control and CHAP authentication to protect the confidentiality of TOE user data.
	OE.ACCESS The TOE environment must implement access control mechanisms to protect the confidentiality of TOE data accessed by users.	OE.ACCESS upholds this policy by ensuring that the organization has taken the appropriate measures to protect the confidentiality of TOE user data by applying access control rules on the hypervisor storage and VMs utilizing it.

Every policy is mapped to one or more objectives in the table above. This complete mapping demonstrates that the defined security objectives enforce all defined policies.

8.2.3 Security Objectives Rationale Relating to Assumptions

Table 17 gives a mapping of assumptions and the environmental objectives that uphold them.

Table 17 – Assumptions: Objectives Mapping

Assumptions	Objectives	Rationale
A.ADMIN_AUTH The TOE environment provides a secure repository of users that are authorized to manage the TOE.	OE.AUTH The TOE environment must provide a secure repository of user accounts used to manage the TOE.	OE.AUTH upholds this assumption by ensuring that the vCenter Server provides a repository of TOE users.
A.ADMIN_PROTECT The TOE environment provides the workstation used to manage the TOE that is free of malicious software.	OE.ADMIN_PROTECT The administrator workstation must be protected from any external interference or tampering.	OE.ADMIN_PROTECT upholds this assumption by ensuring that the administrator workstation is protected from external interference or tampering.
A.LOCATE The physical environment must be suitable for supporting a computing device in a secure setting.	NOE.PHYSICAL The TOE and the environment upon which it relies is located within a controlled access facility.	NOE.PHYSICAL upholds this assumption by ensuring that physical security is provided within the TOE environment that provides appropriate protection to the system and network resources.
A.MANAGE There are one or more competent individuals assigned to manage the TOE and the security of the information it contains.	NOE.MANAGE Sites deploying the TOE will provide competent, non-hostile TOE administrators who are appropriately trained and follow all administrator guidance. TOE administrators will ensure the system is used securely.	NOE.MANAGE upholds this assumption by ensuring that competent individuals are assigned to manage the TOE and the TSF.

Assumptions	Objectives	Rationale
A.NETWORK The TOE environment provides the network infrastructure required for its operation that is appropriately secured and protected from interference or tampering.	OE.NETWORK The TOE environment must be implemented such that the TOE is appropriately secured and protected from interference or tampering.	OE.NETWORK upholds this assumption by ensuring that the TOE environment will provide the appropriate connectivity to allow the TOE to perform its function in a secure manner.
A.NOEVIL The administrators who manage the TOE are non-hostile, appropriately trained, and follow all guidance.	NOE.MANAGE Sites deploying the TOE will provide competent, non-hostile TOE administrators who are appropriately trained and follow all administrator guidance. TOE administrators will ensure the system is used securely.	NOE.MANAGE upholds this assumption by ensuring that the users who manage the TOE are non-hostile, appropriately trained, and follow all guidance.
A.PLATFORM The TOE environment provides the hardware and hypervisor necessary for the operation of the TOE.	OE.PLATFORM The TOE hardware and hypervisor must support all required TOE functions.	OE.PLATFORM upholds this assumption by ensuring that the TOE hardware provides the necessary compute, memory, storage, hypervisor, and other resources necessary to support the TOE's operation.
A.PROTECT The TOE software will be protected from unauthorized modification.	OE.PROTECT The TOE environment must protect itself and the TOE from external interference or tampering.	OE.PROTECT upholds this assumption by ensuring that the TOE environment provides protection from external interference and tampering.

Every assumption is mapped to one or more objectives in the table above. This complete mapping demonstrates that the defined security objectives uphold all defined assumptions.

8.3 Rationale for Extended Security Functional Requirements

A family of EXT_FHA requirements was created to specifically address the high availability self-tests performed by the TOE. The purpose of this family of requirements is to call out high availability functionality provided by the TOE. These requirements have no dependencies since the stated requirements embody all the necessary security functions. These requirements exhibit functionality that can be easily documented in the ADV assurance evidence and thus do not require any additional Assurance Documentation.

8.4 Rationale for Extended TOE Security Assurance Requirements

There are no extended SARs defined for this ST.

8.5 Security Requirements Rationale

The following discussion provides detailed evidence of coverage for each security objective.

8.5.1 Rationale for Security Functional Requirements of the TOE Objectives

Table 18 below shows a mapping of the objectives and the SFRs that support them.

Table 18 – Objectives: SFRs Mapping

Objective	Requirements Addressing the Objective	Rationale
O.ACCESS The TOE must implement rules to govern access to stored user data and system resources.	FDP_ACC.1 Subset access control	This requirement satisfies O.ACCESS by ensuring that the Virtual Storage Access Control SFP is applied to all storage connection attempts by iSCSI clients.
	FDP_ACF.1 Security attribute based access control	This requirement satisfies O.ACCESS by ensuring that the TOE enforces the Virtual Storage Access Control SFP on all storage connection attempts iSCSI clients.
	FMT_MSA.1 Management of security attributes	This requirement satisfies O.ACCESS by ensuring that the TOE identifies users prior to allowing access to administrative functions to ensure that only those trusted users may manage the security behavior of the TOE.
	FMT_MSA.3 Static attribute initialization	This requirement satisfies O.ACCESS by ensuring that the TOE identifies users prior to allowing access to administrative functions to ensure that only those trusted users may manage the security behavior of the TOE.
	FRU_PRS.2 Full priority of service	This requirement satisfies O.ACCESS by ensuring that the TOE assigns policies about using system resources. This ensures that each volume is assigned a priority for using system resources.
O.ADMIN The TOE must include a set of functions that allow efficient management of its functions, security attributes, and TSF data, ensuring that only authorized TOE administrators (in defined roles) may exercise such control.	FMT_MSA.1 Management of security attributes	This requirement satisfies O.ADMIN by ensuring that the TOE restricts management of security attributes to only those users with the appropriate privileges.
	FMT_MSA.3 Static attribute initialization	This requirement satisfies O.ADMIN by ensuring that the TOE restricts administrative functions to only those users with the appropriate privileges.
	FMT_MTD.1 Management of TSF data	This requirement satisfies O.ADMIN by ensuring that the TOE restricts access to TSF data based on the user's role.
	FMT_SMF.1 Specification of Management Functions	This requirement satisfies O.ADMIN by ensuring that the TOE includes administrative functions to facilitate the management of the TSF.

Objective	Requirements Addressing the Objective	Rationale
	FMT_SMR.1 Security roles	This requirement satisfies O.ADMIN by ensuring that the TOE associates users with roles to provide access to TSF management functions, security attributes, and TSF data.
O.AUDIT The TOE must record security relevant events and (if applicable) associate each event with the identity of the administrator that caused the event.	FAU_GEN.1 Audit data generation	This requirement satisfies O.AUDIT by ensuring that the TOE maintains a record of defined security related events, including relevant details about the event.
	FPT_STM.1 Reliable time stamps	This requirement satisfies O.AUDIT by providing reliable time stamps for audit records, preserving the order of events.
O.IDENTIFICATION The TOE must be able to identify administrators using the authentication mechanisms in the TOE environment prior to allowing any access to TOE administrative functions and TSF data. An administrator's username must be associated with every management action.	FIA_UID.2 User identification before any action	This requirement satisfies O.IDENTIFICATION by ensuring that each user is identified before being allowed access to any TSF functionality.
O.TSF_PROTECT The TOE must protect its functions and TSF data to ensure its security policies are enforced and capabilities are intact when drive or node failures occur. It also must provide for the ability to check that its nodes are operating correctly.	FPT_FLS.1(a) Failure with preservation of secure state (for disks)	This requirement satisfies O.TSF_PROTECT by ensuring the TOE preserves a secure state upon defined drive or node hardware failures.
	FPT_FLS.1(b) Failure with preservation of secure state (for NVMe cards)	This requirement satisfies O.TSF_PROTECT by ensuring the TOE preserves a secure state upon NVMe PCIe flash card failures.
	FRU_FLT.1(a) Degraded fault tolerance for disks	This requirement satisfies O.TSF_PROTECT by ensuring the continued operation of the TOE in a degraded state from drive or node hardware failures.
	FRU_FLT.1(b) Degraded fault tolerance for cards	This requirement satisfies O.TSF_PROTECT by ensuring the continued operation of the TOE in a degraded state from NVMe PCIe flash card failures.
	EXT_FHA_TST.1 Component testing	This requirement satisfies O.TSF_PROTECT by ensuring the TOE performs self-tests of the node hardware and accelerator node availability to detect failures.
O.USER_DATA_PROTECT The TOE must ensure the integrity of stored user data by monitoring for errors and automatically correcting them or provide functionality to restore user data from a snapshot.	FDP_ROL.1 Basic rollback	This requirement satisfies O.USER_DATA_PROTECT by permitting rollbacks of modified or deleted files from within a storage volume based on the created snapshots.
	FDP_SDI.2 Stored data integrity monitoring and action	This requirement satisfies O.USER_DATA_PROTECT by ensuring user data is monitored for integrity errors.

8.5.2 Security Assurance Requirements Rationale

EAL2+ was chosen to provide a low to moderate level of assurance that is consistent with good commercial practices. As such, minimal additional tasks are placed upon the vendor assuming the vendor follows reasonable software engineering practices and can provide support to the evaluation for design and testing efforts. The chosen assurance level is appropriate with the threats defined for the environment. The TOE is expected to be in a non-hostile position and embedded in or protected by other products designed to address threats that correspond with the intended environment. At EAL2+, the system will have incurred a search for obvious flaws to support its introduction into the non-hostile environment.

The augmentation of ALC_FLR.2 was chosen to give greater assurance of the developer's on-going flaw remediation processes.

8.5.3 Dependency Rationale

The SFRs in this ST satisfy all of the required dependencies listed in the Common Criteria, applicable PPs, and SFRs explicitly stated in this ST. Table 19 lists each requirement to which the TOE claims conformance and indicates whether the dependent requirements are included. As the table indicates, all dependencies have been met.

Table 19 – Functional Requirements Dependencies

SFR ID	Dependencies	Dependency Met	Rationale
FAU_GEN.1	FPT_STM.1	✓	
FDP_ACC.1	FDP_ACF.1	✓	
FDP_ACF.1	FDP_ACC.1	✓	
	FMT_MSA.3	✓	
FDP_ROL.1	FDP_ACC.1	✓	
FDP_SDI.2	No dependencies		
FIA_UID.2	No dependencies		
FMT_MSA.1	FDP_ACC.1	✓	
	FMT_SMF.1	✓	
	FMT_SMR.1	✓	
FMT_MSA.3	FMT_MSA.1	✓	
	FMT_SMR.1	✓	
FMT_MTD.1	FMT_SMF.1	✓	
	FMT_SMR.1	✓	
FMT_SMF.1	No dependencies		
FMT_SMR.1	FIA_UID.1	✓	Although FIA_UID.1 is not included, FIA_UID.2, which is hierarchical to FIA_UID.1 is included. This satisfies this dependency.
FPT_FLS.1(a)	No dependencies		
FPT_FLS.1(b)	No dependencies		

SFR ID	Dependencies	Dependency Met	Rationale
FPT_STM.1	No dependencies		
FRU_FLT.1(a)	FPT_FLS.1(a)	✓	
FRU_FLT.1(b)	FPT_FLS.1(b)	✓	
FRU_PRS.2	No dependencies		
EXT_FHA_TST.1	No dependencies	✓	

9. Acronyms

Table 20 defines the acronyms used throughout this document.

Table 20 – Acronyms

Acronym	Definition
ACL	Access Control List
API	Application Programming Interface
CC	Common Criteria
CEM	Common Evaluation Methodology
CHAP	Challenge Handshake Authentication Protocol
CM	Configuration Management
COTS	Commercial off-the-Shelf
DHCI	Dynamic Hyperconverged Infrastructure
EAL	Evaluation Assurance Level
F/O	Failover
Gb	Gigabit
GUI	Graphical User Interface
HDD	Hard Disk Drive
I/O	Input/Output
ID	Identification
IQN	iSCSI Qualified Name
iSCSI	Internet Small Computer System Interface
IT	Information Technology
LUN	Logical Unit Number
MB	Megabyte
NVMe	Non-Volatile Memory Express
OSP	Organizational Security Policy
PCIe	Peripheral Component Interconnect Express
PDF	Portable Document Format
PP	Protection Profile
QoS	Quality of Service
RAID	Redundant Array of Independent Disks
RAM	Random Access Memory
SAN	Storage Area Network

Acronym	Definition
SAR	Security Assurance Requirement
SDK	Software Development Kit
SEC	Scalar Erasure Coding
SFP	Security Functional Policy
SFR	Security Functional Requirement
SSO	Single Sign-On
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
USB	Universal Serial Bus
VM	Virtual Machine
vPG	Virtual Performance Group

Prepared by:

Corsec Security, Inc.

13921 Park Center Road, Suite 460
Herndon, VA 20171
United States of America

Phone: +1 703 267 6050
Email: info@corsec.com
http://www.corsec.com