Dell EMC™ Elastic Cloud Storage™ v3.2

Security Target

Evaluation Assurance Level (EAL): EAL2+

Doc No: 2032-000-D102
Version: 1.0
6 April 2018

EMC Corporation
176 South Street
Hopkinton, MA, USA
01748

Prepared by:
EWA-Canada
1223 Michael Street North, Suite 200
Ottawa, Ontario, Canada
K1J7T2

CONTENTS

LIST OF TABLES

LIST OF FIGURES

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 and Packages. This ST does not conform to a Protection Profile.

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.

1.2 SECURITY TARGET REFERENCE

ST Title:	Dell EMC™ Elastic Cloud Storage™ v3.2 Security Target
ST Version:	1.0
ST Date:	6 April 2018

1.3 TOE REFERENCE

TOE Identification: Dell EMC™ Elastic Cloud Storage™ v3.2 Build 531

TOE Developer: Dell EMC

TOE Type: Data Storage (Other Devices and Systems)

1.4 TOE OVERVIEW

Elastic Cloud Storage (ECS) is a software-defined cloud storage platform that supports the storage, manipulation, and analysis of unstructured data on commodity hardware. ECS is specifically designed to support mobile, cloud, big data, and social networking applications. It is deployed as a turnkey storage appliance using qualified commodity servers and disks. ECS provides object and file user access to stored data.

At a high level ECS is composed of the following main components:

- ECS Portal and Provisioning Services – provides a Web-based portal that allows self-service, automation, reporting and management of ECS nodes. It also handles licensing, authentication, multi-tenancy, and provisioning services.
- Data Services – provides services, tools and Application Programming Interfaces (APIs) to support Object, and Hadoop Distributed File System (HDFS) and Network File System (NFS) version 3. HDFS access is not included in the evaluated configuration.
- Storage Engine – responsible for storing and retrieving data, managing transactions, and protecting and replicating data.
- Fabric – provides clustering, health, software and configuration management as well as upgrade capabilities and alerting.
- Infrastructure – uses SUSE Linux Enterprise Server (SLES) 12 SP2 as the base operating system.
- Hardware – the hardware is provided as a turnkey appliance made up of industry standard hardware components.

1.4.1 DEPLOYMENT

ECS can be deployed as a single site or in a multi-site configuration. The evaluated configuration is a single site deployment. The building blocks of an ECS deployment include:

- Virtual Data Center (VDC) – a VDC is a geographical location defined as a single ECS deployment within a site.
- Storage Pool – a storage pool can be thought of as a subset of nodes and its associated storage belonging to a VDC. An ECS node can belong to only one storage pool; a storage pool can have any number of nodes, the

minimum being five. A storage pool can be used as a tool for physically separating data belonging to different applications.

- Replication Group – replication groups define where storage pool content is protected and locations from which data can be read or written. Local replication groups protect objects within the same VDC against disk or node failures.
- Namespace - a namespace, which is conceptually the same as a “tenant,” is a logical construct. The key characteristic of a namespace is that users from one namespace cannot access objects belonging to another namespace. Namespaces can represent a department within an organization or a group within a department.
- Buckets – buckets are containers for object data. Buckets are created in a namespace to give applications access to data stored within ECS. In S3, these containers are called “buckets” and this term has been adopted by ECS. In Atmos, the equivalent of a bucket is a “subtenant”; in Swift, the equivalent of a bucket is a “container”, and for Content Addressed Storage (CAS), a bucket is a “CAS pool”. Buckets are global resources in ECS.

In a single site, storage pools are defined, then the VDC is created with namespaces and buckets.

The TOE is a hardware and software TOE.

1.4.2 Security Features

ECS provides the following security features:

- Auditing and Alert capabilities
- Controlled access to data resources
- Protection against loss of data
- Secure management of security features
- Data at Rest Encryption (D@RE) functionality

1.5 TOE DESCRIPTION

1.5.1 Physical Scope

ECS is deployed as a turnkey storage appliance using qualified commodity servers and disks.

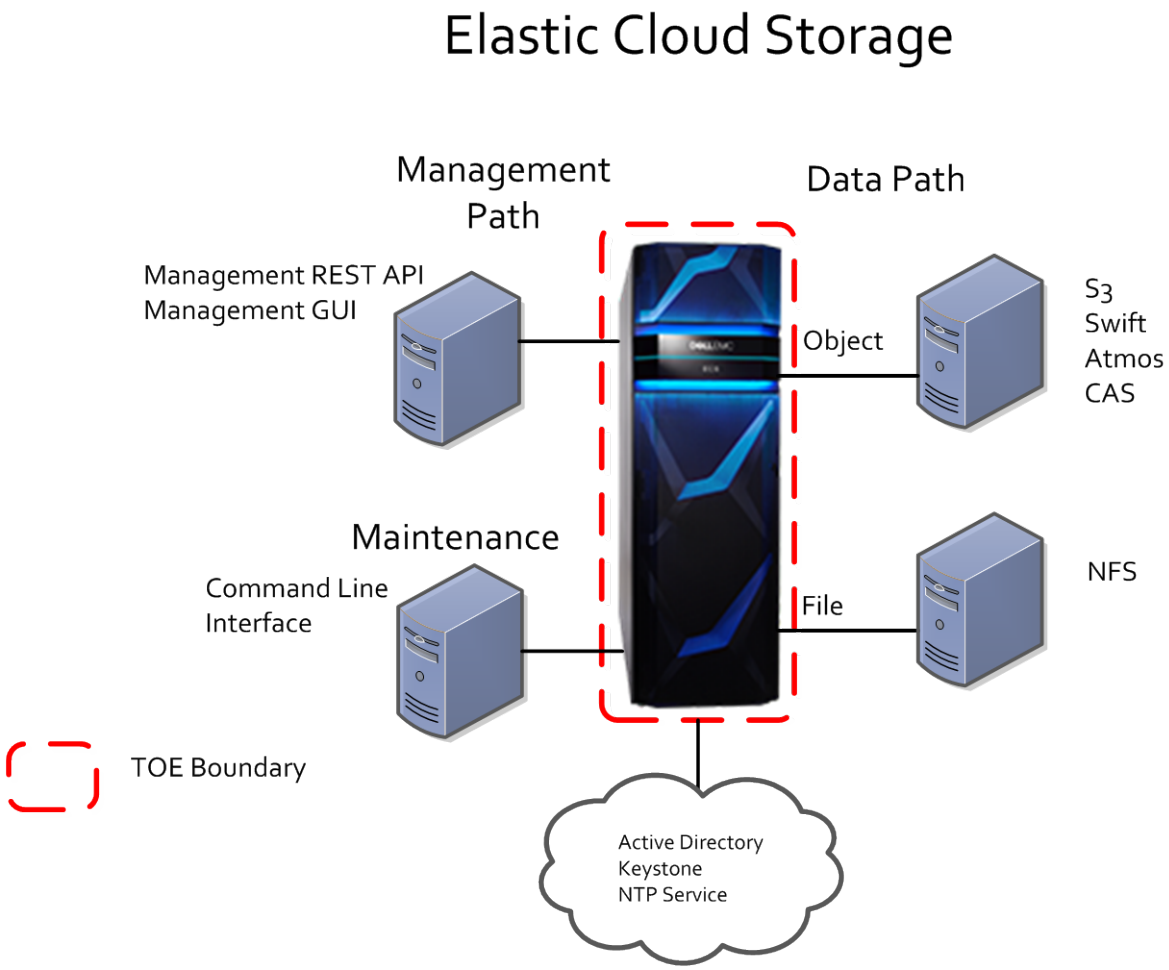

Figure 1 – Elastic Cloud Storage TOE Diagram

1.5.2 TOE Components

The following hardware and software make up the TOE.

TOE Component	Description
Hardware	ECS Gen 2, 5 node, 30 disk hardware
Software	ECS Version 3.2 build 531[1]

Table 1 – TOE Hardware and Software

[1] Viewed in the UI, the version appears as Version 3.2.0.0-101423.d3b297f

1.5.3 TOE Environment

The following operating system, hardware and network components are required for operation of the TOE in the evaluated configuration.

Component	Supporting Software and Operating System	Supporting Hardware
Management Path	Windows 10 This machine supports the applications required to support the Management path options: • Management REST API – PuTTy • Management Graphical User Interface (GUI) – Chrome Browser	General Purpose Computer Hardware
Maintenance	Windows 10 This machine supports an SSH Client to access the Command Line Interface (CLI) for TOE installation	General Purpose Computer Hardware
Data Path Access	SLES 12 SP2 This machine supports all of the applications required to exercise the Data path options: • S3 – S3Curl • Swift – Curl • Atmos – AtmosCurl • CAS – CenteraExerciser	General Purpose Computer Hardware
Domain Controller	Windows Server 2012 R2 with Active Directory This server supports LDAP authentication for Management path users	General Purpose Computer Hardware
Keystone Server	SLES 12 SP2 This server supports Lightweight Directory Access Protocol (LDAP) for OpenStack to provide	General Purpose Computer Hardware

Component	Supporting Software and Operating System	Supporting Hardware
	authentication services for the Swift object Data path	

Table 2 – Non-TOE Hardware and Software

1.5.4 TOE Guidance

The TOE includes the following guidance documentation:

- Elastic Cloud Storage (ECS) Version 3.2 Security Configuration Guide (302-004-495 01 Published March 2018)
- Elastic Cloud Storage (ECS) Version 3.2 Administrator's Guide (302-004-490 01 Published March 2018)
- EMC Elastic Cloud Storage (ECS) CLI Quick Reference (302-001-998 Rev 01 dated June 2015)
- EMC® Centera®, SDK Version 3.3, Programmers Guide (069001127 RevA13 dated July 2012)
- EMC® Centera®, SDK Version 3.3, API Reference Guide (069001185 REV A09 dated July 2012)
- EMC® Atmos™, Version 2.4, Programmer's Guide (302-002-655 Rev01 dated March 2016)
- Elastic Cloud Storage (ECS), version 3.2, Data Access Guide (302-004-491 01 Published March 2018)

1.5.5 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
Security Audit	Audit entries are generated for security related events. The audit logs may be filtered, and reviewed by authorized administrators. Failure events are monitored, and an alert is presented to the administrator where human intervention may be required.

Functional Classes	Description
Cryptographic Support	Cryptographic key generation and key destruction functionality and cryptographic operation support Data at Rest (D@RE) encryption of stored user information. ECS uses the RSA BSAFE® Crypto-J JSAFE and JCE Software Module (Software Version: 6.1) module, Cryptographic Module Validation Program (CMVP) certificate number 2057. The vendor affirms that no source code changes were made to the cryptographic module prior to recompilation into the TOE software.
User Data Protection	User authentication and access control list (ACL) information associated with data is evaluated to determine if a user is permitted to access requested data. The TOE ensures the integrity of stored data. A retention period can be associated with data objects. Data may not be deleted until the retention period expires.
Identification and Authentication	Both Data path users and Management path/maintenance users must identify and authenticate prior to TOE access.
Security Management	The TOE provides management capabilities via a REST API and a web-based GUI. Management functions allow the administrators to review audit records, configure storage options, and configure users and roles.
Protection of the TSF	The TOE protects user data against disk and node failure. Reliable time stamps are provided for audit records.
Resource Utilization	Read and write access is maintained in the case of limited failures.
TOE Access	Management path users may initiate logout, or will be logged out automatically after a configurable period of inactivity.

Table 3 – Logical Scope of the TOE

1.5.6 Functionality Excluded from the Evaluated Configuration

The following features are excluded from this evaluation:

- Hadoop Distributed File System (HDFS) user access to stored data
- Transfer of audit records to a syslog server
- Advanced Retention Management (ARM)
- Geo-Federation and Geo-Replication (Multi-site deployment)

The CLI is used only during installation in the evaluated configuration.

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 ASSURANCE PACKAGE CLAIM

This Security Target claims conformance to Evaluation Assurance Level 2 augmented with ALC_FLR.2 Flaw Reporting Procedures.

2.3 PROTECTION PROFILE CONFORMANCE CLAIM

This ST does not claim conformance of the TOE with any Protection Profile (PP).

3 SECURITY PROBLEM DEFINITION

Table 4 lists the threats addressed by the TOE. Mitigation to the threats is through the objectives identified in Section 4.1, Security Objectives for the TOE.

3.1 THREATS

Potential threat agents are authorized TOE users, and unauthorized persons. The level of expertise of both types of attacker is assumed to be unsophisticated. TOE users are assumed to have access to the TOE, extensive knowledge of TOE operations and to possess a level of skill commensurate with their responsibilities. They have moderate resources to alter TOE parameters, but are assumed not to be wilfully hostile. Unauthorized persons have little knowledge of TOE operations, a low level of skill, limited resources to alter TOE parameters and no physical access to the TOE.

Threat	Description
T.ACCOUNT	An authorized user of the TOE could gain unauthorized access to TOE configuration information, or perform operations for which no access rights have been granted, via user error, system error, or other actions.
T.LOSS	A network or hardware failure could result in temporary or permanent loss of user data.
T.UNAUTH	A hostile/unauthorized user could gain access to stored data by bypassing the protection mechanisms of the TOE.
T.UNDETECT	Authorized or unauthorized users may be able to access TOE data or modify TOE behavior without a record of those actions in order to circumvent TOE security functionality.

Table 4 – Threats

3.2 ORGANIZATIONAL SECURITY POLICIES

Organizational Security Policies (OSPs) are security rules, procedures, or guidelines imposed on the operational environment. Table 5 lists the OSPs that are presumed to be imposed upon the TOE or its operational environment by an organization that implements the TOE in the Common Criteria evaluated configuration.

Organizational Security Policy	Description
P.CRYPTO	The TOE shall incorporate cryptographic mechanisms to protect against potential disclosure of the data it has been entrusted to store.
P.RETAIN	The TOE shall provide a means to identify a retention period before which data is not to be deleted, and prevent data from

Organizational Security Policy	Description
	being deleted prior to the expiry of the retention period.

Table 5 – Organizational Security Policies

3.3 ASSUMPTIONS

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

Assumptions	Description
A.AUTH	The operational environment provides authentication services to the TOE in support of access control decisions.
A.LOCATE	The TOE will be located within controlled access facilities, which will prevent unauthorized physical access.
A.NETWORK	The operational environment will provide secure network communications to protect data that is sent to and received from the TOE.
A.NOEVIL	The authorized administrators are not careless, wilfully negligent, or hostile, are appropriately trained and will follow the instructions provided by the TOE documentation.

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.

Security Objective	Description
O.ADMIN	The TOE must provide functionality that enables an authorized administrator to manage TOE security functions, and must ensure that only authorized administrators are able to access such functionality.
O.AUDIT	The TOE must provide a means of logging security related events, and a means of filtering and viewing those events. The TOE must alert administrators to failure events that could compromise the availability of data to users.
O.CRYPTO	The TOE must provide cryptographic functions to support encryption of data at rest.
O.IDAUTH	The TOE must ensure that users are identified and authenticated prior to allowing access to data or administrative functions.
O.INTEGRITY	The TOE must protect the data that it has been entrusted to store from integrity errors due to node or disk failure.
O.PROTECT	The TOE must protect the data that it has been entrusted to store from unauthorized access.
O.RETAIN	The TOE must prevent the deletion of data prior to expiry of the assigned retention period.
O.TIME	The TOE must provide reliable time stamps.

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 domain or by non-technical or procedural means.

Security Objective	Description
OE.ADMIN	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security. There are an appropriate number of authorized administrators trained to maintain the TOE, including its security policies and practices. Authorized administrators are non-hostile and follow all administrator guidance.
OE.COMMS	Communications with the TOE are appropriately protected by the operational environment through the use of physical and network security.
OE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to the enforcement of security are protected from interference, tampering, and physical attack.
OE.SERVICE	The operational environment shall provide an Active Directory server and a Keystone server to provide authentication services.

Table 8 – Security Objectives for the Operational Environment

4.3 SECURITY OBJECTIVES RATIONALE

The following Table maps the security objectives to the assumptions and threats identified for the TOE.

	T.ACCOUNT	T.LOSS	T.UNAUTH	T.UNDETECT	P.CRYPTO	P.RETAIN	A.AUTH	A.LOCATE	A.NETWORK	A.NOEVIL
O.ADMIN	X		X	X						
O.AUDIT		X		X						
O.CRYPTO					X					
O.IDAUTH	X		X	X						
O.INTEGRITY		X								

	T.ACCOUNT	T.LOSS	T.UNAUTH	T.UNDETECT	P.CRYPTO	P.RETAIN	A.AUTH	A.LOCATE	A.NETWORK	A.NOEVIL
O.PROTECT			X							
O.RETAIN						X				
O.TIME				X		X				
OE.ADMIN										X
OE.COMMS									X	
OE.PHYSICAL								X		
OE.SERVICE							X			

Table 9 – Mapping Between Objectives, Threats, and Assumptions

4.3.1 Security Objectives Rationale Related to Threats

The security objectives rationale related to threats traces the security objectives for the TOE and the Operational Environment back to the threats addressed by the TOE.

Threat: **T.ACCOUNT**	An authorized user of the TOE could gain unauthorized access to TOE configuration information, or perform operations for which no access rights have been granted, via user error, system error, or other actions.	
Objectives:	O.ADMIN	The TOE must provide functionality that enables an authorized administrator to manage TOE security functions, and must ensure that only authorized administrators are able to access such functionality.
	O.IDAUTH	The TOE must ensure that users are identified and authenticated prior to allowing access to data or administrative functions.
Rationale:	O.ADMIN mitigates this threat by ensuring that access to the security management functions of the TOE are restricted to authorized administrators. O.IDAUTH mitigates this threat by ensuring that all authorized administrators are identified and authenticated prior to gaining access to the TOE security management functions, and that all	

	users are identified and authenticated prior to being granted access to data.

Threat: **T.LOSS**	A network or hardware failure could result in temporary or permanent loss of user data.	
Objectives:	O.AUDIT	The TOE must provide a means of logging security related events, and a means of filtering and viewing those events. The TOE must alert administrators to failure events that could compromise the availability of data to users.
	O.INTEGRITY	The TOE must protect the data that it has been entrusted to store from integrity errors due to node or disk failure.
Rationale:	O.AUDIT mitigates this threat by alerting administrators to failure events that require human intervention for correction. O.INTEGRITY mitigates this threat by ensuring that the TOE provides the ability to protect data in the case of node or disk failure.	

Threat: **T.UNAUTH**	A hostile/unauthorized user could gain access to stored data by bypassing the protection mechanisms of the TOE.	
Objectives:	O.ADMIN	The TOE must provide functionality that enables an authorized administrator to manage TOE security functions, and must ensure that only authorized administrators are able to access such functionality.
	O.IDAUTH	The TOE must ensure that users are identified and authenticated prior to allowing access to data or administrative functions.
	O.PROTECT	The TOE must protect the data that it has been entrusted to store from unauthorized access.
Rationale:	O.ADMIN mitigates this threat by providing authorized administrators the ability to manage TOE security functions. O.IDAUTH mitigates this threat by ensuring that all users are identified and authenticated prior to gaining access to the TOE security management functions and data. O.PROTECT mitigates this threat by ensuring that only authorized users have access to stored data.	

Threat: **T.UNDETECT**	Authorized or unauthorized users may be able to access TOE data or modify TOE behavior without a record of those actions in order to circumvent TOE security functionality.	
Objectives:	O.ADMIN	The TOE must provide functionality that enables an authorized administrator to manage TOE security functions, and must ensure that only authorized administrators are able to access such functionality.
	O.AUDIT	The TOE must provide a means of logging security related events, and a means of filtering and viewing those events. The TOE must alert administrators to failure events that could compromise the availability of data to users.
	O.IDAUTH	The TOE must be able to identify and authenticate users prior to allowing access to the administrative functions and TSF data.
	O.TIME	The TOE must provide reliable time stamps.
Rationale:	O.ADMIN mitigates this threat by ensuring that access to the security functions of the TOE are restricted to authorized administrators. O.AUDIT counters this threat by ensuring that the TOE tracks all management actions taken against the TOE, and by providing a means to review these records. O.IDAUTH mitigates this threat by ensuring that all authorized administrators are identified and authenticated prior to gaining access to the TOE security management functions. O.TIME supports this policy by providing reliable time stamps in support of audit records.	

4.3.2 Security Objectives Rationale Related to OSPs

The security objectives rationale related to the OSPs traces the security objectives for the TOE back to the OSPs applicable to the TOE.

Policy: **P.CRYPTO**	The TOE shall incorporate cryptographic mechanisms to protect against potential disclosure of the data it has been entrusted to store.	
Objectives:	O.CRYPTO	The TOE must provide cryptographic functions to support encryption of data at rest.
Rationale:	O.CRYPTO supports this policy by providing the capability to encrypt stored data.	

Policy: **P.RETAIN**	The TOE shall provide a means to identify a retention period before which data is not to be deleted, and prevent data from being deleted prior to the expiry of the retention period.	
Objectives:	O.RETAIN	The TOE must prevent the deletion of data prior to expiry of the assigned retention period.
	O.TIME	The TOE must provide reliable time stamps.
Rationale:	O.RETAIN supports this policy by ensuring that the TOE prevents deletion of data prior to expiry of the assigned retention period. O.TIME supports this policy by providing reliable time stamps in support the functions that determine whether or not the retention period has expired.	

4.3.3 Security Objectives Rationale Related to Assumptions

The security objectives rationale related to assumptions traces the security objectives for the operational environment back to the assumptions for the TOE's operational environment.

Assumption: **A.AUTH**	The operational environment provides authentication services to the TOE in support of access control decisions.	
Objectives:	OE.SERVICE	The operational environment shall provide an Active Directory server and a Keystone server to provide authentication services.
Rationale:	OE.SERVICE supports this assumption by providing Active Directory and Keystone authentication services for use in access control decisions.	

Assumption: **A.LOCATE**	The TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	
Objectives:	OE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to the enforcement of security are protected from interference, tampering, and physical attack.
	OE.COMMS	Communications with the TOE are appropriately protected by the operational environment through the use of physical and network security.
Rationale:	OE.PHYSICAL supports this assumption by protecting the physical resources of the TOE from attack.	

	OE.COMMS supports this assumption by providing for physical security in support of communications with the TOE.

Assumption: A.NETWORK	The operational environment will provide secure network communications to protect data that is sent to and received from the TOE.	
Objectives:	OE.COMMS	Communications with the TOE are appropriately protected by the operational environment through the use of physical and network security.
Rationale:	OE.COMMS supports this assumption by ensuring that communications with the TOE are protected by the network in the operational environment.	

Assumption: A.NOEVIL	The authorized administrators are not careless, wilfully negligent, or hostile, are appropriately trained and will follow the instructions provided by the TOE documentation.	
Objectives:	OE.ADMIN	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security. There are an appropriate number of authorized administrators trained to maintain the TOE, including its security policies and practices. Authorized administrators are non-hostile and follow all administrator guidance.
Rationale:	OE.ADMIN supports this assumption by ensuring that the administrators managing the TOE have been specifically chosen to be careful, attentive and non-hostile.	

5 EXTENDED COMPONENTS DEFINITION

This section specifies the extended Security Functional Requirement (SFR) used in this ST. An extended SFR, Retention of data (FDP_RET_EXT.1) has been created to address additional security features of the TOE.

5.1 CLASS FDP: USER DATA PROTECTION

A new family has been added with one SFR. FDP_RET_EXT Retention of data addresses the requirement to retain data. FDP_RET_EXT.1 Retention of data addresses retention requirements for stored data, and is modelled after FDP_SDI.1 Stored data integrity monitoring and FPT_RCV.1 Manual recovery.

5.1.1 FDP_RET_EXT

Family Behaviour

This family provides requirements that address retention of user data while it is stored within containers controlled by the TOE Security Functionality (TSF).

Component Levelling

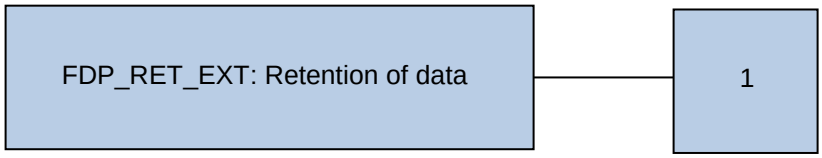

Figure 2 – FDP_RET: Data Retention Component Levelling

Management: FDP_RET_EXT.1

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

a) Setting the retention period.

Audit: FDP_RET_EXT.1

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

a) Minimal: changes to the retention period.

5.1.1.1 FDP_RET_EXT.1 Retention of data

Hierarchical to: No other components.

Dependencies: FPT_STM.1

FDP_RET_EXT.1.1 The TSF shall allow a retention period to be assigned to user data.

FDP_RET_EXT.1.2 Where a retention period has been assigned to data, the TSF shall deny requests to delete the data until the retention period has expired, or has been removed.

5.2 SECURITY ASSURANCE REQUIREMENTS

This ST does not include extended Security Assurance Requirements.

6 SECURITY REQUIREMENTS

Section 6 provides security functional and assurance requirements that must be satisfied by a compliant TOE. These requirements consist of functional components from Part 2 of the CC, an extended requirement, and an Evaluation Assurance Level (EAL) that contains assurance components from Part 3 of the CC.

6.1 CONVENTIONS

The CC permits four types of operations to be performed on functional requirements: selection, assignment, refinement, and iteration. These operations, when performed on requirements that derive from CC Part 2, are identified in this ST in the following manner:

- Selection: Indicated by surrounding brackets, e.g., [selected item].
- Assignment: Indicated by surrounding brackets and italics, e.g., [*assigned item*].
- Refinement: Refined components are identified by using **bold** for additional information, or ~~strikeout~~ for deleted text.
- Iteration: Indicated by assigning a number in parenthesis to the end of the functional component identifier as well as by modifying the functional component title to distinguish between iterations, e.g., 'FDP_ACC.1(1), Subset access control (administrators)' and 'FDP_ACC.1(2) Subset access control (devices)'.

6.2 TOE SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for this ST consist of the following components from Part 2 of the CC and extended components defined in Section 5, summarized in Table 10 - Summary of Security Functional Requirements.

Class	Identifier	Name
Security Audit (FAU)	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
Cryptographic Support (FCS)	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.4	Cryptographic key Destruction
	FCS_COP.1	Cryptographic operation

Class	Identifier	Name
User Data Protection (FDP)	FDP_ACC.1	Subset access control
	FDP_ACF.1	Security attribute based access control
	FDP_SDI.2	Stored data integrity monitoring and action
	FDP_RET_EXT.1	Retention of data
Identification and Authentication (FIA)	FIA_UAU.2	User authentication before any action
	FIA_UID.2	User identification before any action
Security Management (FMT)	FMT_MSA.1	Management of security attributes
	FMT_MSA.3	Static attribute initialisation
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.1	Security roles
Protection of the TSF (FPT)	FPT_FLS.1	Failure with preservation of secure state
	FPT_STM.1	Reliable time stamps
Resource Utilization (FRU)	FRU_FLT.1	Degraded fault tolerance
TOE Access (FTA)	FTA_SSL.3	TSF-initiated termination
	FTA_SSL.4	User-initiated termination

Table 10 – Summary of Security Functional Requirements

6.2.1 Security Audit (FAU)

6.2.1.1 FAU_ARP.1 Security alarms

Hierarchical to: No other components.

Dependencies: FAU_SAA.1 Potential violation analysis

FAU_ARP.1.1 The TSF shall ~~take~~ [*indicate an alert in the GUI*] upon detection of a potential security violation.

6.2.1.2 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) [*no other specifically defined auditable events*].

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*].

6.2.1.3 FAU_SAA.1 Potential violation analysis

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [*failure events*] known to indicate a potential security violation;

b) [*no other rules*].

6.2.1.4 FAU_SAR.1 Audit review

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.1.1 The TSF shall provide [*authorised management users*] with the capability to read [*all audit information*] from the audit records.

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

6.2.1.5 FAU_SAR.3 Selectable audit review

Hierarchical to: No other components.

Dependencies: FAU_SAR.1 Audit review

FAU_SAR.3.1 The TSF shall provide the ability to apply [*filtering*] of audit data based on [*date time range, and namespace*].

6.2.2 Cryptographic Support (FCS)

6.2.2.1 FCS_CKM.1 Cryptographic key generation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*Deterministic Random Bit Generator*] and specified cryptographic key sizes [*256 bits*] that meet the following: [*SP800-90A*].

6.2.2.2 FCS_CKM.4 Cryptographic key destruction

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [*zeroization*] that meets the following: [*FIPS 140-2*].

6.2.2.3 FCS_COP.1 Cryptographic operation

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES*] and cryptographic key sizes [*256 bits*] that meet the following: [*FIPS 197*].

6.2.3 User Data Protection (FDP)

6.2.3.1 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 [*Object Storage Access Control SFP*] on [*Subjects: Users accessing storage*
Objects: Storage objects
Operations: *Read, Write, Execute*].

6.2.3.2 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 initialisation

FDP_ACF.1.1 The TSF shall enforce the [*Object Storage Access Control SFP*] to objects based on the following:
[*Subjects: Users accessing storage*

Security Attributes:

- *Username*
- *Authentication status (success or failure)*

Objects: Storage objects

Security Attributes:

- *ACLs for each object*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [*A successfully authenticated subject of the TOE is allowed to perform an operation if the content of the Access Control List (containing permissions) for the object authorizes the Subject to perform the desired operation*].

FDP_ACF.1.3 The TSF shall explicitly authorise 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 following additional rules: [*no additional rules*].

6.2.3.3 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 [*integrity errors*] on all objects, based on the following attributes: [*checksum*].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall [*rebuild the data*].

6.2.3.4 FDP_RET_EXT.1 Retention of data

Hierarchical to: No other components.

Dependencies: FPT_STM.1

FDP_RET_EXT.1.1 The TSF shall allow a retention period to be assigned to user data.

FDP_RET_EXT.1.2 Where a retention period has been assigned to data, the TSF shall deny requests to delete the data until the retention period has expired, or has been removed.

6.2.4 Identification and Authentication (FIA)

6.2.4.1 FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

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

6.2.4.2 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.

6.2.5 Security Management (FMT)

6.2.5.1 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_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 The TSF shall enforce the [*Object Storage Access Control SFP*] to restrict the ability to [query, modify, delete] the security attributes [*username, object ACLs*] to [*authorized Management path users*].

6.2.5.2 FMT_MSA.3 Static attribute initialisation

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 [*Object Storage Access Control SFP*] to provide [permissive] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [*authorized Management path users*] to specify alternative initial values to override the default values when an object or information is created.

6.2.5.3 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: [*storage configuration, viewing of audit records, management of Management path users, management of Data path users and Data path user access, enabling/disabling of D@RE, management of retention periods and policies*].

6.2.5.4 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 [*System Admin, System Monitor, Namespace Admin*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

6.2.6 Protection of the TSF (FPT)

6.2.6.1 FPT_FLS.1 Failure with preservation of secure state

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [*disk failures, node failures*].

6.2.6.2 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.

6.2.7 Resource Utilization (FRU)

6.2.7.1 FRU_FLT.1 Degraded fault tolerance

Hierarchical to: No other components.

Dependencies: FPT_FLS.1 Failure with preservation of secure state

FRU_FLT.1.1 The TSF shall ensure the operation of [*read and write operations*] when the following failures occur: [*loss of one node and one disk*].

6.2.8 TOE Access (FTA)

6.2.8.1 FTA_SSL.3 TSF-initiated termination

Hierarchical to: No other components.

Dependencies: No dependencies.

FTA_SSL.3.1 The TSF shall terminate an interactive session after ~~a~~ [*2 hours of user inactivity*].

6.2.8.2 FTA_SSL.4 User-initiated termination

Hierarchical to: No other components.

Dependencies: No dependencies.

FTA_SSL.4.1 The TSF shall allow user-initiated termination of the user's own interactive session.

6.3 SECURITY FUNCTIONAL REQUIREMENTS RATIONALE

The following Table provides a mapping between the SFRs and Security Objectives.

	O.ADMIN	O.AUDIT	O.CRYPTO	O.IDAUTH	O.INTEGRITY	O.PROTECT	O.RETAIN	O.TIME
FAU_ARP.1		X						
FAU_GEN.1		X						
FAU_SAA.1		X						
FAU_SAR.1		X						
FAU_SAR.3		X						
FCS_CKM.1			X					
FCS_CKM.4			X					
FCS_COP.1			X					
FDP_ACC.1						X		
FDP_ACF.1						X		
FDP_SDI.2					X			
FDP_RET_EXT.1							X	
FIA_UAU.2	X			X				
FIA_UID.2	X			X				
FMT_MSA.1	X							
FMT_MSA.3	X							
FMT_SMF.1	X							
FMT_SMR.1	X							
FPT_FLS.1					X			
FPT_STM.1								X
FRU_FLT.1					X			
FTA_SSL.3	X							
FTA_SSL.4	X							

Table 11 – Mapping of SFRs to Security Objectives

6.3.1 SFR Rationale Related to Security Objectives

The following rationale traces each SFR back to the Security Objectives for the TOE.

Objective: O.ADMIN	The TOE must provide functionality that enables an authorized administrator to manage TOE security functions, and must ensure that only authorized administrators are able to access such functionality.	
Security Functional Requirements:	FIA_UAU.2	User authorization before any action
	FIA_UID.2	User identification before any action
	FMT_MSA.1	Management of security attributes
	FMT_MSA.3	Static attribute initialisation
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.1	Security roles
	FTA_SSL.3	TSF-initiated termination
	FTA_SSL.4	User-initiated termination
Rationale:	FIA_UID.2 and FIA_UAU.2 ensure that users are identified and authenticated prior to being allowed access to manage TOE security functions. FMT_MSA.1 ensures that access to the security attributes supporting access control functions is restricted to authorized Management path users. FMT_MSA.3 ensures that default values for the security attributes that make up that TSF data are permissive. FMT_SMF.1 provides security management functionality to support storage configuration, viewing of audit records, user management, access control, enabling/disabling D@RE and retention period and policy management. FMT_SMR.1 provides the security roles for Management path users. FTA_SSL.3 and FTA_SSL.4 protects the security management functionality from unauthorized access by allowing a user to terminate the user's own interactive GUI session, or by terminating the session after a configurable period of inactivity.	

Objective: O.AUDIT	The TOE must provide a means of logging security related events, and a means of filtering and viewing those events. The TOE must alert administrators to failure events that could compromise the availability of data to users.

Security Functional Requirements:	FAU_ARP.1	Security alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potential violation analysis
	FAU_SAR.1	Audit review
	FAU_SAR.3	Selectable audit review
Rationale:	FAU_GEN.1 outlines what data must be included in audit records and what events must be audited. FAU_SAR.1 provides functionality to review audit records. FAU_SAR.3 allows the Management path user to filter those records for more convenient viewing. FAU_SAA.1 provides the functionality to monitor the audited events for failures that may require human intervention. FAU_ARP.1 indicates an alert when such an event is detected.	

Objective: O.CRYPTO	The TOE must provide cryptographic functions to support encryption of data at rest.	
Security Functional Requirements:	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.4	Cryptographic key Destruction
	FCS_COP.1	Cryptographic operation
Rationale:	FCS_CKM.1 provides the key generation, FCS_CKM.4 provides the key destruction and FCS_COP.1 provides the cryptographic operation that supports the data at rest encryption functionality provided by the TOE.	

Objective: O.IDAUTH	The TOE must ensure that users are identified and authenticated prior to allowing access to data or administrative functions.	
Security Functional Requirements:	FIA_UAU.2	User authentication before any action
	FIA_UID.2	User identification before any action
Rationale:	FIA_UID.2 ensures that the TOE verifies that the user has been identified before being allowed to access data or security management functions. FIA_UAU.2 ensures that the TOE verifies that the user has been authenticated before being allowed to access data or security management functions.	

Objective: O.INTEGRITY	The TOE must protect the data that it has been entrusted to store from integrity errors due to node or disk failure.	
Security Functional Requirements:	FDP_SDI.2	Stored data integrity monitoring and action
	FPT_FLS.1	Failure with preservation of secure state
	FRU_FLT.1	Degraded fault tolerance
Rationale:	FDP_SDI.2 monitors the stored data for integrity errors and rebuilds the data if an error is detected. FPT_FLS.1 preserves the integrity of the data in the case of disk or node failure. FRU_FLT.1 ensures that read and write operations continue to be processed in the case of disk and node failure.	

Objective: O.PROTECT	The TOE must protect the data that it has been entrusted to store from unauthorized access.	
Security Functional Requirements:	FDP_ACC.1	Subset access control
	FDP_ACF.1	Security attribute based access control
Rationale:	FDP_ACC.1 and FDP_ACF.1 details the Object Storage Access Control SFP which ensures that only authorized users are able to access data resources protected by the TOE.	

Objective: O.RETAIN	The TOE must prevent the deletion of data prior to expiry of the assigned retention period.	
Security Functional Requirements:	FDP_RET_EXT.1	Retention of data
Rationale:	FDP_RET_EXT.1 ensures that data is not deleted prior to the expiry of the retention period.	

Objective: O.TIME	The TOE must provide reliable time stamps.	
Security Functional Requirements:	FPT_STM.1	Reliable time stamps
Rationale:	FPT_STM.1 provides reliable time stamps for use on audit records	

	and the enforcement of retention periods.

6.4 DEPENDENCY RATIONALE

Table 12 identifies the Security Functional Requirements from Part 2 of the CC and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency.

SFR	Dependency	Dependency Satisfied	Rationale
FAU_ARP.1	FAU_SAA.1	✓	
FAU_GEN.1	FPT_STM.1	✓	
FAU_SAA.1	FAU_GEN.1	✓	
FAU_SAR.1	FAU_GEN.1	✓	
FAU_SAR.3	FAU_SAR.1	✓	
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1	✓	Satisfied by FCS_COP.1
	FCS_CKM.4	✓	
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Satisfied by FCS_CKM.1
FCS_COP.1	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	✓	Satisfied by FCS_CKM.1
	FCS_CKM.4	✓	
FDP_ACC.1	FDP_ACF.1	✓	
FDP_ACF.1	FDP_ACC.1	✓	
	FMT_MSA.3	✓	
FDP_SDI.2	None	N/A	
FDP_RET_EXT.1	FPT_STM.1	✓	
FIA_UAU.2	FIA_UID.1	✓	FIA_UID.2 is hierarchical to FIA_UID.1; this dependency has been satisfied.
FIA_UID.2	None	N/A	
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1	✓	Satisfied by FDP_ACC.1

SFR	Dependency	Dependency Satisfied	Rationale
	FMT_SMR.1	✓	
	FMT_SMF.1	✓	
FMT_MSA.3	FMT_MSA.1	✓	
	FMT_SMR.1	✓	
FMT_SMF.1	None	N/A	
FMT_SMR.1	FIA_UID.1	✓	
FPT_FLS.1	None	N/A	
FPT_STM.1	None	N/A	
FRU_FLT.1	FPT_FLS.1	✓	
FTA_SSL.3	None	N/A	
FTA_SSL.4	None	N/A	

Table 12 – Functional Requirement Dependencies

6.5 TOE SECURITY ASSURANCE REQUIREMENTS

The TOE assurance requirements for this ST consist of the requirements corresponding to the EAL 2 level of assurance, as defined in the CC Part 3, augmented by the inclusion of Flaw reporting procedures (ALC_FLR.2). EAL 2 was chosen for competitive reasons. The developer is claiming the ALC_FLR.2 augmentation since there are a number of areas where current practices and procedures exceed the minimum requirements for EAL 2.

The assurance requirements are summarized in Table 13.

Assurance Class	Assurance Components	
	Identifier	**Name**
Development (ADV)	ADV_ARC.1	Security architecture description
	ADV_FSP.2	Security-enforcing functional specification
	ADV_TDS.1	Basic design
Guidance Documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life-Cycle Support (ALC)	ALC_CMC.2	Use of a CM system
	ALC_CMS.2	Parts of the TOE CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_FLR.2	Flaw reporting procedures
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_COV.1	Evidence of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment (AVA)	AVA_VAN.2	Vulnerability analysis

Table 13 – Security Assurance Requirements

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 SECURITY AUDIT

Audit logs capture security management functions resulting from use of the Management path interfaces. The management GUI allows only authorized management users to read audit records. The audit records may be filtered based on a time and date range and the namespace for which they were generated.

When events that could indicate errors that require human intervention to correct are detected, an alert is generated. This alert appears in a widget in the management GUI, and would also be seen in the management REST API by a user that polls for alerts.

TOE Security Functional Requirements addressed: FAU_ARP.1, FAU_GEN.1, FAU_SAA.1, FAU_SAR.1 FAU_SAR.3.

7.2 CRYPTOGRAPHIC SUPPORT

ECS uses the RSA BSAFE® Crypto-J JSAFE and JCE Software Module (Software Version: 6.1) module (CMVP certificate number 2057) in support of the Data at Rest Encryption (D@RE) functionality.

AES 256 is the only supported algorithm. The module employs a FIPS-approved HMAC Deterministic Random Bit Generator (HMAC DRBG SP 800-90A) for generating symmetric keys used in AES. Key destruction is performed on keys held in memory after their use using the BSAFE clearSensitiveData function. Stored keys are not destroyed. New keys are created for each object, and are encrypted using a Key Encryption Key (KEK). KEKs are encrypted with a master key, which itself is stored encrypted. The KEKs and master key are never stored on the same drive.

TOE Security Functional Requirements addressed: FCS_CKM.1, FCS_CKM.4, FCS_COP.1.

7.3 USER DATA PROTECTION

7.3.1 Object Storage Access Control SFP

Each object within the storage space has an associated Access Control List (ACL), which is enforced in accordance with the rules of the protocol. The ACLs associated with an object are defined by the owner. Users must be authenticated before the ACL will be evaluated. Authentication occurs locally, or ECS verifies that the user has been authenticated, depending upon the data access protocol being used.

7.3.1.1 Amazon S3 Authentication

For Amazon S3, each user is assigned a secret key. The user presents the ECS user ID (which maps to the AWS Access key ID in Amazon S3 terminology) and a signature created using the secret key. Once successful authentication is confirmed, ECS verifies that the user has permission based on the contents of the associated ACL. Then, the user will be granted access to the object.

7.3.1.2 OpenStack Swift Authentication

ECS enables Data path users to authenticate with the ECS Swift service and obtain a token that can be used when making subsequent API calls to the ECS Swift service.

For v1 and v2 protocols, access to the ECS object store using the OpenStack Swift protocol requires an ECS object user account and a Swift password.

For OpenStack Swift version 3 protocol, users must be assigned to projects and roles outside of ECS using a Keystone V3 service. Within ECS, an administrator must add the Swift group and Swift password to the user account. ECS provides support for Keystone V3 by validating authentication tokens provided by OpenStack Swift users. For Keystone V3, users are created outside of ECS using a Keystone V3 service. ECS does not perform authentication, but validates the authentication token with the Keystone V3 service.

7.3.1.3 Atmos Authentication

Atmos users are authenticated locally by username and password.

7.3.1.4 Content Addressable Storage (CAS) Authentication

CAS features are assigned to the user profile to allow object access via the CAS protocol. Information added through the ECS management interface is used to make up the elements of a CAS profile. This creates a PEA (Pool Entry Authorization) file for use in CAS applications. For CAS, ACLs allowing access to data objects are also configured through the ECS management interface.

7.3.1.5 NFSv3 Authentication

File based storage may be accessed using an NFS client and local authentication.

7.3.2 Data Integrity

Data integrity is a key security function of ECS. The integrity of data is protected in three ways: Triple-mirroring, Erasure Coding and Checksums. Checksums are used to verify integrity. Erasure coding ensures that the data can be rebuilt in the case of disk or node loss. Triple-mirroring protects data before erasure coding is complete.

7.3.2.1 Checksums

During write operations, the checksum is calculated in memory and then written to disk. On reads, data is read along with the checksum, and then the checksum is calculated in memory from the data read and compared with the checksum

stored in disk to determine data integrity. The storage engine runs a consistency checker in the background which performs checksum verification over the entire data set.

7.3.2.2 Triple-mirroring

All types of information relating to objects, such as data, metadata, and index are written to chunks. At ingest, the chunks are triple mirrored to three different nodes within the ECS system. This technique of triple mirroring allows for data protection of the data in case of a node or disk failure. Data in chunks is triple-mirrored until the chunk is full. After that, the data is erasure coded to provide the same integrity using less storage space.

7.3.2.3 Erasure Coding

When erasure coding is applied, a chunk is broken into 12 data fragments and 4 coding (parity) fragments, with a Cold Storage option for 10+2 as well. The resulting 16 fragments are dispersed across the nodes. The storage engine can reconstruct a chunk from any 12 of the 16 fragments.

All data in ECS is erasure coded except the index and system metadata. The index provides location to objects and chunks and is frequently accessed; hence, it is always kept in triple-mirrored chunks for protection.

When a chunk is full (128MB), or after a set period of time, it is sealed and erasure-coded. Erasure coding is conducted as a background process. After erasure coding completes, the mirrored copies are discarded and a single erasure coded copy persists.

7.3.2.4 Data Reconstruction

If a disk or node fails, the data is reconstructed. If a request is made to read an object that has become unavailable, the object is reconstructed. If a disk or node fails, the hardware is replaced and the data that had been held by those resources may be reconstructed.

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

7.3.3 Data Retention

ECS implements policy based record retention to prevent data being modified or deleted within a specified retention period. Retention periods and retention policies can be defined in metadata associated with objects, and is checked each time a request to modify an object is made. There are two ways of defining retention: retention periods and retention policies.

7.3.3.1 Retention Periods

Retention periods are assigned at the object and/or bucket level. Each time an attempt is made to modify or delete an object, an expiration time is calculated, where the object expiration time is equal to the object creation time plus the retention period. Where a retention period is assigned to a bucket, the retention

period for the bucket is checked and the expiration time calculated based on the retention period set on the object and the value set on the bucket, whichever is the longest.

Applying a retention period to a bucket means that the retention period for all objects in a bucket can be changed at any time, and can override the value written to the object by an object client by setting it to a longer period. It is possible to specify that an object is retained indefinitely.

7.3.3.2 Retention Policies

A retention policy may be applied to a set of data objects within a namespace to apply a retention period on those objects. This allows flexibility to change the period associated with a policy and, in doing so, automatically change the retention period that applies to any objects that have been assigned that policy.

By applying a retention policy to a number of objects, rather than applying a retention period directly, a change to the retention policy will cause the retention period to be changed for the complete set of objects to which the policy has been applied. A request to modify an object that falls before the expiration of the retention period will be disallowed. For example, a retention policy can be created for email and this policy may have a one year retention period. All email is then assigned this policy. If it is decided that all email should be held for two years, the policy is changed. All email assigned to this policy now has a two year retention period.

TOE Security Functional Requirements addressed: FDP_RET_EXT.1.

7.4 IDENTIFICATION AND AUTHENTICATION

For the Management path interfaces, LDAP authentication using Active Directory is used in the evaluated configuration. Users must be identified and authenticated before any access to TOE functions is granted through the management REST API or management GUI.

Authentication for Data path access depends upon the protocol being used. In the evaluated configuration, local authentication is used for S3, Atmos and CAS access, and local and Keystone authentication is used for Swift access.

TOE Security Functional Requirements addressed: FIA_UAU.2, FIA_UID.2.

7.5 SECURITY MANAGEMENT

ECS provides a management REST API and a management GUI to administer the TOE. The Management GUI is a graphical representation of the functionality provided through the REST API.

Through the management REST API and Management GUI, authorized Management path users can:

- Configure storage
- View audit records
- Perform user and role administration of Management path users

- Perform user administration of Data path users
- Administer authentication and access for Data path users
- Enable and disable D@RE
- Manage retention periods and policies

Access to data is controlled based on username, authentication status and the ACLs for the requested object. Username and ACLs (in some cases) may be managed through the management REST API and GUI. Default values for username and object ACLs are permissive, in that they may be set outside of the TOE.

The roles and privileges available in ECS are shown in Table 14.

Role	Privileges
System Admin	Management users in the System Admin role can configure storage, perform namespace administration, and configure user permissions.
System Monitor	Users in the System Monitor role can view all ECS Portal data, but cannot make any changes.
Namespace Admin	Users in the Namespace Admin role can assign local users as object users for the namespace and create and manage buckets within the namespace.

Table 14 – Roles and Privileges

A root user account, which is assigned to the System Admin role, is provided for initial access. Note that this root account is not related to node-level Linux accounts. As indicated in the administrative guidance, users are directed to change the password for all pre-provisioned accounts on initial access. One or more System Admin accounts should be created and the root account should no longer be used.

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

7.6 PROTECTION OF THE TSF

The ability to rebuild objects from erasure coded data ensures that data is not lost due to the failure of disks or nodes. The number of disks and nodes that can be lost without loss of data is dependent upon the number of nodes implemented. In the evaluated configuration, there are 5 nodes. This configuration will maintain a secure state (i.e. no loss of data) in the case of losses up to and including the simultaneous loss of one node and one disk.

ECS requires that a Network Time Protocol (NTP) service be available. Time from the NTP service is maintained by the TOE and provided as reliable time stamps on audit records, and is used when determining retention expiry dates.

TOE Security Functional Requirements addressed: FPT_FLS.1, FPT_STM.1.

7.7 RESOURCE UTILIZATION

The number of disks and nodes that can be lost while still maintaining read and write functionality is dependent upon the number of nodes implemented. In the evaluated configuration, there are five nodes. This configuration can tolerate the simultaneous loss of one node and one disk.

TOE Security Functional Requirements addressed: FRU_FLT.1.

7.8 TOE ACCESS

Interactive sessions with the Management GUI or the management REST API are terminated after a period of inactivity. Once authenticated, users are provided with a token. This token is presented with subsequent requests. After two hours of inactivity, the token expires and the user must log in again. This period of inactivity is configurable. Users may log out of the GUI at any time.

TOE Security Functional Requirements addressed: FTA_SSL.3, FTA_SSL.4.

8 TERMINOLOGY AND ACRONYMS

8.1 TERMINOLOGY

The following terminology is used in this ST:

Term	Description
Bucket	Buckets are object containers and can be used to control access to objects and to set properties that define attributes for all contained objects.
Data path user	A Data path user accesses stored data resources.
Namespace	Namespace is conceptually the same as a 'tenant'. It may represent a department within an organization, or a group within a department.
Management path user	A Management path user is assigned a role and performs security management functions through the management REST API or management GUI.

Table 15 – Terminology

8.2 ACRONYMS

The following acronyms are used in this ST:

Acronym	Definition
ACL	Access Control List
AES	Advanced Encryption Standard
API	Application Programming Interface
ARM	Advanced Retention Management
CAS	Content Addressed Storage
CC	Common Criteria
CLI	Command Line Interface
CM	Configuration Management
CMVP	Cryptographic Module Validation Program
D@RE	Data at Rest Encryption
DRBG	Deterministic Random Bit Generator
EAL	Evaluation Assurance Level
ECS	Elastic Cloud Storage

Acronym	Definition
FIPS	Federal Information Processing Standards
GUI	Graphical User Interface
HDFS	Hadoop Distributed File System
IT	Information Technology
KEK	Key Encryption Key
LDAP	Lightweight Directory Access Protocol
N/A	Not applicable
NFS	Network File System
NTP	Network Time Protocol
OSP	Organizational Security Policy
PEA	Pool Entry Authorization
PP	Protection Profile
REST	Representational State Transfer
SFP	Security Function Policy
SFR	Security Functional Requirement
SLES	SUSE Linux Enterprise Server
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
VDC	Virtual Data Center

Table 16 – Acronyms