CyberArk

Privileged Access Manager – Linux Components

Including Privileged Session Manager SSH (PSMP) v14.0

Security Target

Version 1.9

Jun 2024

Document prepared by

www.lightshipsec.com

Document History

Version	Date	Author	Description
0.1	10 Sept 2021	Marina Ibrishimova	First draft.
0.2	26 Oct 2021	Marina Ibrishimova	Migrated ST to APP PP v1.4
0.3	13 Mar 2022	Marina Ibrishimova	Addressed vendor's comments, applied latest technical decisions.
0.4	19 Sep 2022	Marina Ibrishimova	Added latest TDs.
0.5	5 Jan 2023	Marina Ibrishimova	Addressed evaluator's observations.
0.6	21 Feb 2023	Marina Ibrishimova	Addressed evaluator's observations.
0.7	16 Apr 2023	Marina Ibrishimova	Addressed evaluator's observations.
0.8	25 Apr 2023	Marina Ibrishimova	Addressed ALC observations.
0.9	11 Oct 2023	Marina Ibrishimova	Added latest TDs.
1.0	16 Oct 2023	Marina Ibrishimova	Addressed vendor comments.
1.1	29 Nov 2023	Oshrit IM	Update TOE description
1.2	13 Dec 2023	Marina Ibrishimova	Addressed ASE observations.
1.3	24 Jan 2024	Enav Coresh	Addressed ASE observations.
1.4	30 Jan 2024	Enav Coresh	Addressed ASE observations.
1.5	11 Feb 2024	Enav Coresh	Addressed ASE observations.
1.6	8 Apr 2024	Marina Ibrishimova	Addressed ASE comments.
1.7	9 May 2024	Marina Ibrishimova	Addressed ASE comments.
1.8	16 May 2024	Marina Ibrishimova	Addressed ASE comments.
1.9	13 Jun 2024	Marina Ibrishimova	Addressed ASE comments.

Table of Contents

List of Tables

1 ST Introduction

1 This Security Target (ST) defines the CyberArk Privileged Access Manager – Linux Components Target of Evaluation (TOE) for the purposes of Common Criteria (CC) evaluation.

2 The TOE is a software-based solution that runs on Linux and is a part of CyberArk's Privileged Access Manager (PAM) Solution. PAM enables organizations to secure, provision, control, and monitor all activities associated with privileged identities used in enterprise systems and applications.

3 The TOE is an application composed of the PAM component Privileged Session Manager SSH Proxy (PSMP).

4 PSMP enables organizations to secure, control, and monitor privileged access to *NIX systems.

1.1 ST and TOE References

Table 1: Evaluation identifiers

ST Title	CyberArk Software Ltd. Privileged Access Manager – Linux Components including Privileged Session Manager SSH (PSMP) v14.0 Security Target
ST version	Version 1.9
ST Author	Lightship Security
ST Publication Date	Jun 13, 2024
TOE Reference	Cyberark Privileged Access Manager – Linux Components including PSMP v14.0.0.14

1.2 TOE Overview

1.2.1 Type

5 The TOE is a software application that runs on the Linux Operating System (OS), and it is a part of the CyberArk Privileged Access Manager (PAM) Solution suite.

6 The TOE is composed of the PAM component PSMP, which provides the functionality to establish SSH connection to remote devices.

7 In its evaluated configuration, the TOE is a part of CyberArk's PAM Solution. PSMP is installed on a single instance of Red Hat Enterprise Linux (RHEL) 8. RHEL contains the required OpenSSL and OpenSSH Server packages that are required to secure communications with clients. PSMP contains the required OpenSSH Client package that is installed on the RHEL machine for communicating with remote targets.

1.2.2 Usage

8 A user connects to PSMP by providing a target device, target user, Vault user, and Vault password, which are then relayed to the Digital Vault Server for verification.

Once the user is verified, PSMP retrieves the target user's credentials to connect the user to the target device. While a user is connected to a target device and is performing activities in the privileged session, PSMP actively records all the activities and uploads them to the Digital Vault Server. PSMP is used to establish SSH connection to a remote target. PSMP separates the users from remote targets and stores the remote target's password in the Digital Vault. When a user connects to a remote target, PSMP retrieves the remote target's password from the Digital Vault using TLS through port 443, so PSMP enables connections to privileged devices without having to divulge the passwords to the user. PSMP logs user's activities that are performed in the privileged session and uploads the logs to the Digital Vault Server, where they are accessed by authorized users.

9 Communication between the TOE and the Digital Vault Server is over TLS as shown in Figure 1.

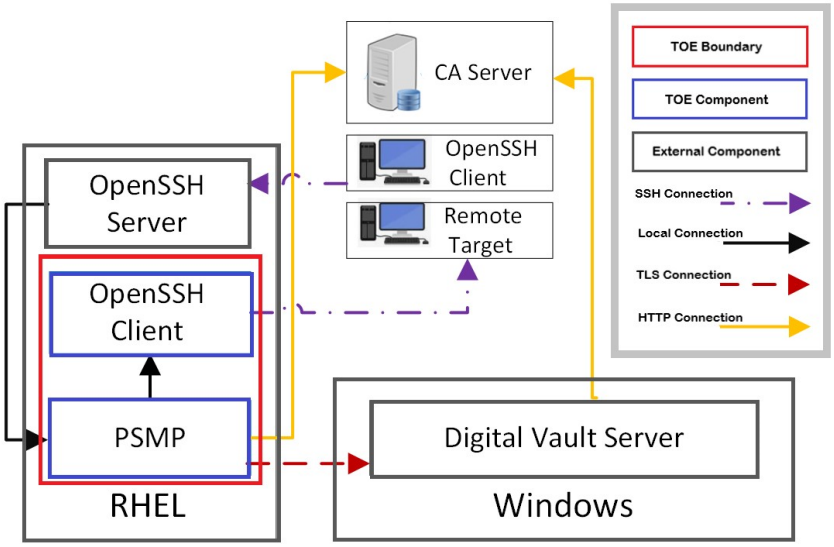

Figure 1: Example TOE deployment

1.2.2.1 Environment

10 It is assumed that there will be no untrusted users, administrators, or software on the TOE server component. Access to the server's OS must be limited to authorized personnel and secured with an authentication method. In addition, the TOE server component is intended to be deployed in a physically secured cabinet, room, or data centre with the appropriate level of physical access control and physical protection (e.g., badge access, fire control, locks, alarms, etc.).

1.2.3 Security Functions

11 The TOE provides the following security functions:

a) **Cryptographic Support.** The TOE uses a CAVP-validated cryptographic algorithm provided by its OpenSSL FIPS Object Module with the CyberArk library. The library is used to support the establishment of trusted channels to protect data in transit. In the evaluated configuration, the TOE's cryptographic library is used by the OpenSSH Client to remote targets and the TLS client connection to the Digital Vault Server. The TOE provides the cryptographic functionality listed in Table 11.

b) **User Data Protection.** The TOE stores sensitive information in the form of encrypted passwords in non-volatile memory. The TOE limits its access to only network connectivity when accessing the platform's hardware resources. The network connection is used for communications from the TOE to the Digital Vault Server, the TOE to the target devices, and the user to the TOE. The TOE also accesses the Digital Vault Server's sensitive information repository (Safes) when it needs to authenticate users or request root credentials.

c) **Identification and Authentication.** To validate the Digital Vault Server's certificate during the TLS handshake, the TOE implements functionality to validate X.509 certificates. The TOE uses a CRL to check certificate revocation status and does not establish a connection to the Digital Vault Server when the CRL is unavailable.

d) **Security Management.** The TOE is configured with default file permissions already in place and does not provide default credentials for user authentication. The TOE relies on the platform for storing and setting configuration options within its config files. Administrators are able to view the status of the TOE in addition to being able to start, stop, and restart it.

e) **Privacy.** The TOE does not store or transmit any Personally Identifiable Identification (PII).

f) **Protection of the TSF.** The TOE protects against exploitation by implementing address space layout randomization (ASLR) except for vesical and not allocating memory with both writing and execution. The TOE is also compatible with SELinux and is compiled with stack-based buffer overflow protection. It also stores user-modifiable files to directories that do not contain executable files. The TOE uses standard platform APIs and includes only the third-party libraries that it needs to perform its functionality. The TOE version can be checked using commands provided by the platform. Checking for updates to the TOE is reliant on the platform's functionally. Any update downloaded for the TOE must be installed using the platform's package manager. An administrator will install a public key from CyberArk that is used by the package manager to verify the integrity of any updates to the TOE.

g) **Trusted Path.** The TOE provides a trusted channel between itself and target devices over SSH using OpenSSH Client. The SSH software used by the TOE follows the Extended Package for Secure Shell. A trusted TLS channel is used between itself and the Digital Vault Server.

1.2.4 Non-TOE Components

12 The TOE operates with the following non-TOE components in the environment:

a) Digital Vault Server – Digital Vault, v14.0.0.40.

Digital Vault Server provides secure storage and access to privileged account files, and to the administrator and session activity files.

b) RHEL – A Red Hat Enterprise Linux server v8.0

RHEL v8.0 is the OS, on which the TOE runs.

c) PSM (optional) – Privileged Session Management v14.0.0.40 through an RDP

1.3 TOE Description

1.3.1 Physical Scope

13 The physical scope of the TOE is the Privileged Access Manager – Linux Components Linux application. The TOE version is v14.0.0.14 and the TOE is delivered through the CyberArk's online customer portal, which uses AWS Marketplace. The TOE delivery format is *.elf. The customer portal can be accessed after customers register to the portal https://cyberark.my.site.com/s/login.

1.3.1.1 Guidance Parts

14 The TOE includes the following guidance documents, which are delivered to customers through a download link that becomes available to them after they purchase the TOE and sign in for the CyberArk Privileged Access Manager - Self-Hosted customer portal:

a) Privileged Access Manager – Linux Components Common Criteria Guide, v1.6 (PDF), May 2024

b) PAM Self-Hosted v14.0, 25-Jan-2024, No. A8474D5E4B6532ED3402D38B46F7DB15F650CA75EBD0372BB891F3ECDC7089CE, as follow:

   Download the PAM Self hosted document described above >go to Cyberark Portal cyberark.my.site.com/mplace/s/#software > choose Privileged Access Manager Self-Hosted > go to Components > choose Documentation > download Production-PublicHelp-PAS - 14.0.zip and Extract > choose OnlineHelp.htm > choose Install and Harden components

   a) Install: Installation > install PAM Self-Hosted
   b) Upgrade: Installation > Upgrade
   c) Admin: Administrator > Components

1.3.1.2 Configuration List

15 The evaluation package consists of the following:

a) Privileged Access Manager – Linux Components (TOE)

b) Privileged Access Manager – Linux Components Security Target, v1.9

c) Privileged Access Manager – Linux Components Common Criteria Guide, v1.6

d) Privileged Access Manager – Linux Components Entropy Description, v0.4

e) PAM Self-Hosted v14.0, 25-Jan-2024, No. A8474D5E4B6532ED3402D38B46F7DB15F650CA75EBD0372BB891F3ECDC7089CE

1.3.1.3 Out-of-Scope Functionality

16 All out of scope functionalities are disabled by default in the evaluated configuration of the TOE.

17 The out-of-scope functionalities are as follows:

a) Subnet accounts

b) ADBridge

c) Ticketing

d) MFA caching

e) plink

1.3.2 Logical Scope

18 The logical scope of the TOE comprises the security functions defined in section Security Functions.

1.4 Terminology

Table 2: Terminology

Term	Definition
CA	Certificate Authority
CC	Common Criteria
CPM	CyberArk Central Policy Manager
CRL	Certificate Revocation List
CDP	CRL distribution point
DRBG	Deterministic Random Bit Generator
EAL	Evaluation Assurance Level
IIS	Internet information Services
LDAP	Lightweight Directory Access Protocol
NIAP	National Information Assurance Partnership
PP	Protection Profile
PAM	CyberArk Privileged Access Manager
PSM	CyberArk Privileged Session Manager
PSMP	CyberArk Privileged Session Manager SSH (Secure Shell) Proxy
PVWA	CyberArk Password Vault Web Access
SRP	Secure Remote Password
TOE	Target of Evaluation
TSF	TOE Security Functionality

2 Conformance Claims

19 The following conformance claims are made:

1) CC version 3.1 Revision 5, April 2017
2) CC Part 2 extended, CCMB-2017-04-002, April 2017
3) CC Part 3 extended, CCMB-2017-04-003, April 2017
4) NIAP Protection Profile for Application Software, v1.4 (PP_APP), 2021-10-07
5) NIAP Functional Package for Transport Layer Security, v1.1, 2019-03-01, Conformant.
6) NIAP Functional Package for Secure Shell, v1.0, 2021-05-13, Conformant.
7) NIAP Technical Decisions per Table 3

Table 3: NIAP Technical Decisions

TD Type	TD #	Name	Rationale if N/A
PP_APP	TD0628	Addition of Container Image to Package Format	
PP_APP	TD0650	Conformance claim sections updated to allow for MOD_VPNC_V2.3 and 2.4	
PP_APP	TD0664	Testing activity for FPT_TUD_EXT.2.2	
PP_APP	TD0717	Format changes for PP_APP_V1.4	
PP_APP	TD0719	ECD for PP APP V1.3 and 1.4	
PP_APP	TD0736	Number of elements for iterations of FCS_HTTPS_EXT.1	N/A. The TOE does not claim FCS_HTTPS_EXT.1/Server
PP_APP	TD0743	FTP_DIT_EXT.1.1 Selection exclusivity	
PP_APP	TD0747	Configuration Storage Option for Android	N/A. the TOE is not an Android app
PP_APP	TD0756	Update for platform-provided full disk encryption	
PP_APP	TD0780	FIA_X509_EXT.1 Test 4 Clarification	
PP_APP	TD0798	Static Memory Mapping Exceptions	
PP_APP	TD0815	Addition of Conditional TSS Activity for FPT_AEX_EXT.1.5	
PP_APP	TD0822	Correction to Windows Manifest File for FDP_DEC_EXT.1	
PP_APP	TD0823	Update to Microsoft Windows Exploit Protection link in FPT_AEX_EXT.1.3	

TD Type	TD #	Name	Rationale if N/A
PKG_TLS_1.1	DT0779	Updated Session Resumption Support in TLS package V1.1	
PKG_TLS_1.1	TD0770	TLSS.2 connection with no client cert	NA, this SFR is not claimed
PKG_TLS_1.1	TD0739	PKG_TLS_V1.1 has 2 different publication dates	
PKG_TLS_1.1	TD0726	Corrections to (D)TLSS SFRs in TLS 1.1 FP	
PKG_TLS_1.1	TD0513	CA Certificate loading	
PKG_TLS_1.1	TD0499	Testing with pinned certificates	
PKG_TLS_1.1	TD0469	Modification of test activity for FCS_TLSS_EXT.1.1 test 4.1	
PKG_TLS_1.1	TD0442	Updated TLS Ciphersuites for TLS Package	
PKG_SSH_1.0	TD0777	Clarification to Selections for Auditable Events for FCS_SSH_EXT.1	
PKG_SSH_1.0	TD0732	FCS_SSHS_EXT.1.3 Test 2 Update	Not applicable as TOE is not an SSH server
PKG_SSH_1.0	TD0695	Choice of 128 or 256 bit size in AES-CTR in SSH Functional Package.	
PKG_SSH_1.0	TD0682	Addressing Ambiguity in FCS_SSHS_EXT.1 Tests	Not applicable as TOE is not an SSH server

3 Security Problem Definition

20 The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any organizational security policies that the TOE is expected to enforce.

3.1 Threats

Table 4: Threats

Identifier	Description
T.NETWORK_ATTACK	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may engage in communications with the application software or alter communications between the application software and other endpoints in order to compromise it.
T.NETWORK_EAVESDROP	An attacker is positioned on a communications channel or elsewhere on the network infrastructure. Attackers may monitor and gain access to data exchanged between the application and other endpoints.
T.LOCAL_ATTACK	An attacker can act through unprivileged software on the same computing platform on which the application runs. Attackers may provide maliciously formatted input to the application in the form of files or other local communications.
T.PHYSICAL_ACCESS	An attacker may try to access sensitive data at rest.

3.2 Assumptions

Table 5: Assumptions

Identifier	Description
A.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying platform and whatever runtime environment it provides to the TOE.
A.PROPER_USER	The user of the application software is not wilfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.
A.PROPER_ADMIN	The administrator of the application software is not careless, wilfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.

3.3 Organizational Security Policies

21 There are no organizational security policies for the application.

4 Security Objectives

4.1 Objectives for the TOE

Table 6: Security Objectives

Identifier	Description
O.INTEGRITY	Conformant TOEs ensure the integrity of their installation and update packages, and also leverage execution environment-based mitigations. Software is seldom, if ever, shipped without errors. The ability to deploy patches and updates to fielded software with integrity is critical to enterprise network security. Processor manufacturers, compiler developers, execution environment vendors, and operating system vendors have developed execution environment-based mitigations that increase the cost to attackers by adding complexity to the task of compromising systems. Application software can often take advantage of these mechanisms by using APIs provided by the runtime environment or by enabling the mechanism through compiler or linker options.
O.QUALITY	To ensure quality of implementation, conformant TOEs leverage services and APIs provided by the runtime environment rather than implementing their own versions of these services and APIs. This is especially important for cryptographic services and other complex operations such as file and media parsing. Leveraging this platform behavior relies upon using only documented and supported APIs.
O.MANAGEMENT	To facilitate management by users and the enterprise, conformant TOEs provide consistent and supported interfaces for their security-relevant configuration and maintenance. This includes the deployment of applications and application updates through the use of platform-supported deployment mechanisms and formats, as well as providing mechanisms for configuration. This also includes providing control to the user regarding disclosure of any PII.
O.PROTECTED_STORAGE	To address the issue of loss of confidentiality of user data in the event of loss of physical control of the storage medium, conformant TOEs use data-at-rest protection. This involves encrypting data and keys stored by the TOE in order to prevent unauthorized access to this data. This also includes unnecessary network communications whose consequence may be the loss of data.
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application, which should not be exposed outside of the application.

4.2 Objectives for the Operational Environment

Table 7: Operational environment objectives

Identifier	Description
OE.PLATFORM	The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and any discrete execution environment provided to the TOE.
OE.PROPER_USER	The user of the application software is not wilfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.
OE.PROPER_ADMIN	The administrator of the application software is not careless, wilfully negligent or hostile, and administers the software in compliance with the applied enterprise security policy.

5 Security Requirements

5.1 Conventions

22 This document uses the following font conventions to identify the operations defined by the CC:

a) **Assignment.** Indicated with italicized text in square brackets.

b) **Refinement.** Indicated with bold text and strikethroughs in square brackets.

c) **Selection.** Indicated with underlined text in square brackets.

d) **Assignment within a selection.** Indicated with italicized and underlined text in square brackets.

e) **Iteration.** Indicated by adding a slash and a name, e.g., “FCS_COP.1/Hash”.

5.2 Extended Components Definition

23 All extended components (identified by EXT) are reproduced directly from the claimed Protection Profile and therefore no further definition is provided in this document.

5.3 Functional Requirements

Table 8: Summary of SFRs

Requirement	Title	Type
FCS_CKM.1/AK	Cryptographic Asymmetric Key Generation	Selection
FCS_CKM_EXT.1	Cryptographic Key Generation Services	Mandatory
FCS_CKM.2	Cryptographic Key Establishment	Selection
FCS_COP.1/SKC	Cryptographic Operation – Encryption/Decryption	Selection
FCS_COP.1/Hash	Cryptographic Operation – Hashing	Selection
FCS_COP.1/Sig	Cryptographic Operation – Signing	Selection
FCS_COP.1/KeyedHash	Cryptographic Operation – Keyed-Hash Message Authentication	Selection
FCS_RBG_EXT.1	Random Bit Generation Services	Mandatory
FCS_SSH_EXT.1	SSH Protocol	Selection
FCS_SSHC_EXT.1	SSH Protocol - Client	Selection
FCS_STO_EXT.1	Storage of Credentials	Mandatory
FCS_TLS_EXT.1	TLS Protocol	Mandatory

Requirement	Title	Type
FCS_TLSC_EXT.1	TLS Client Protocol	Selection
FCS_TLSC_EXT.5	TLS Client Support for Supported Groups Extension	Selection
FDP_DEC_EXT.1	Access to Platform Resources	Mandatory
FDP_NET_EXT.1	Network Communications	Mandatory
FDP_DAR_EXT.1	Encryption Of Sensitive Application Data	Mandatory
FIA_X509_EXT.1	X.509 Certificate Validation	Selection
FIA_X509_EXT.2	X.509 Certificate Authentication	Selection
FMT_MEC_EXT.1	Supported Configuration Mechanism	Mandatory
FMT_CFG_EXT.1	Secure by Default Configuration	Mandatory
FMT_SMF.1	Specification of Management Functions	Mandatory
FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Information	Mandatory
FPT_API_EXT.1	Use of Supported Services and APIs	Mandatory
FPT_AEX_EXT.1	Anti-Exploitation Capabilities	Mandatory
FPT_TUD_EXT.1	Integrity for Installation and Update	Mandatory
FPT_TUD_EXT.2	Integrity for Installation and Update	Selection
FPT_LIB_EXT.1	Use of Third-Party Libraries	Mandatory
FPT_IDV_EXT.1	Software Identification and Versions	Mandatory
FTP_DIT_EXT.1	Protection of Data in Transit	Mandatory

5.3.1 Cryptographic Support (FCS)

FCS_CKM.1/AK Cryptographic Asymmetric Key Generation

FCS_CKM.1.1/AK The application shall [implement functionality] **to** generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm [

- [**ECC schemes**] using **["NIST curves" P-384 and [P-256]]** that meet the following: [**FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4**],

].

Application note: This SFR was altered by TD0717.

FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1 The application shall [implement asymmetric key generation].

Application note: This SFR was altered by TD0717.

FCS_CKM.2 Cryptographic Key Establishment

FCS_CKM.2.1 The application shall [implement functionality] to perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- [Elliptic curve-based key establishment schemes] that meets the following: [NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"],
- [FFC Schemes using "safe-prime" groups] that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 7919]

].

FCS_COP.1/SKC Cryptographic Operation – Encryption/Decryption

FCS_COP.1.1/SKC The application shall perform [*encryption/decryption*] in accordance with a specified cryptographic algorithm [

- AES-CBC (as defined in NIST SP 800-38A) mode,
- AES-CTR (as defined in NIST SP 800-38D) mode,
- AES-GCM (as defined in NIST SP 800-38D) mode,

] and cryptographic key sizes [128-bit, 256-bit].

Application note: This SFR was altered by TD0717.

FCS_COP.1/Hash Cryptographic Operation – Hashing

FCS_COP.1.1/Hash The application shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm [

- SHA-256,
- SHA-384
- SHA-512

] and **message digest** sizes [

- 256,

- 384
- 512

] **bits** that meet the following: [FIPS Pub 180-4].

Application note: This SFR was altered by TD0717.

FCS_COP.1/Sig Cryptographic Operation – Signing

FCS_COP.1.1/Sig The **application** shall perform [*cryptographic signature services (generation and verification)*] in accordance with a specified cryptographic algorithm [

- **RSA schemes** using cryptographic key sizes of [2048-bit or greater] that meet the following: [FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5]

].

Application note: This SFR was altered by TD0717.

FCS_COP.1/KeyedHash Cryptographic Operation – Keyed-Hash Message Authentication

FCS_COP.1.1/KeyedHash The **application** shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm [

- HMAC-SHA-256
- HMAC-SHA-384
- HMAC-SHA-512

] and [

- no other algorithms

] **with** key sizes [*256, 384, 512*] **and message digest sizes** [256, 384, 512] **and** [no other size] **bits** that meet the following: [*FIPS Pub 198-1, 'The Keyed-Hash Message Authentication Code' and FIPS Pub 180-4 'Secure Hash Standard'*].

Application note: This SFR was altered by TD0717.

FCS_RBG_EXT.1 Random Bit-Generation Services

FCS_RBG_EXT.1.1 The application shall [invoke platform-provided DRBG functionality] for its cryptographic operations.

FCS_SSH_EXT.1 SSH Protocol

FCS_SSH_EXT.1.1 The TOE shall implement SSH acting as a [client] in accordance with that complies with RFCs 4251, 4252, 4253, 4254, [4256, 4344, 5656, 6668] and [no other standard].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods: [

- "keyboard-interactive" (RFC 4256),
- "publickey" (RFC 4252): [
  - ecdsa-sha2-nistp256 (RFC 5656),
  - ecdsa-sha2-nistp384 (RFC 5656)]

] and no other methods.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*262144 bytes*] in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall protect data in transit from unauthorized disclosure using the following mechanisms: [

- aes128-ctr (RFC 4344),
- aes256-ctr (RFC 4344),
- aes128-cbc (RFC 4253),
- aes256-cbc (RFC 4253]

and no other mechanisms.

FCS_SSH_EXT.1.5 The TSF shall protect data in transit from modification, deletion, and insertion using: [

- hmac-sha2-256 (RFC 6668)
- hmac-sha2-512 (RFC 6668)]

and no other mechanisms.

FCS_SSH_EXT.1.6 The TSF shall establish a shared secret with its peer using:

- ecdh-sha2-nistp256 (RFC 5656)
- ecdh-sha2-nistp384 (RFC 5656)]

and no other mechanisms.

FCS_SSH_EXT.1.7 The TSF shall use SSH KDF as defined in [

- RFC 4253 (Section 7.2),
- RFC 5656 (Section 4)

] to derive the following cryptographic keys from a shared secret: *session keys*.

FCS_SSH_EXT.1.8 The TSF shall ensure that [

- a rekey of the session keys

] occurs when any of the following thresholds are met:

- one hour connection time
- no more than one gigabyte of transmitted data, or

- no more than one gigabyte of received data.

FCS_SSHC_EXT.1 SSH Protocol - Client

FCS_SSHC_EXT.1.1 The TSF shall authenticate its peer (SSH server) using: [

- using a local database by associating each host name with a public key corresponding to the following list: [
  - ecdsa-sha2-nistp256 (RFC 5656),
  - ecdsa-sha2-nistp384 (RFC 5656)

  ]

] as described in RFC 4251 section 4.1.

FCS_STO_EXT.1 Storage of Credentials

FCS_STO_EXT.1.1 The application shall [

- invoke the functionality provided by the platform to securely store [*Certificates*],
- implement functionality to securely store [*file keys*] according to [FCS_COP.1/SKC]

] to non-volatile memory.

FCS_TLS_EXT.1 TLS Protocol

FCS_TLS_EXT.1.1 The product shall implement [

- TLS as a client

].

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The product shall implement TLS 1.2 (RFC 5246) and [no earlier TLS versions] as a client that supports the cipher suites [

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

] and also supports functionality for [none].

Application Note: This SFR was altered by TD0442.

FCS_TLSC_EXT.1.2 The product shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3 The product shall not establish a trusted channel if the server certificate is invalid [

- with no exceptions

].

FCS_TLSC_EXT.5 TLS Client Support for Supported Groups Extension

FCS_TLSC_EXT.5.1 The product shall present the Supported Groups Extension in the Client Hello with the supported groups [

- secp256r1
- secp384r1

].

5.3.2 User Data Protection (FDP)

FDP_DEC_EXT.1 Access to Platform Resources

FDP_DEC_EXT.1.1 The application shall restrict its access to [

- network connectivity

].

FDP_DEC_EXT.1.2 The application shall restrict its access to [

- *[PSMP application logs]*

].

FDP_NET_EXT.1 Network Communications

FDP_NET_EXT.1.1 The application shall restrict network communication to [

- user-initiated communication for *[SSH to target machines], [TLS connections to the Digital Vault Server]*

].

FDP_DAR_EXT.1 Encryption of Sensitive Application Data

FDP_DAR_EXT.1.1 The application shall [

- leverage platform-provided functionality to encrypt sensitive data,

- protect sensitive data in accordance with FCS_STO_EXT.1

] in non-volatile memory.

5.3.3 Identification and Authentication (FIA)

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1 The application shall [invoke platform-provided functionality, implement functionality] to validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The application shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates, and that any path constraints are met.
- The application shall validate that any CA certificate includes caSigning purpose in the key usage field
- The application shall validate the revocation status of the certificate using [CRL as specified in RFC 5280 Section 6.3].
- The application shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing Purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the EKU field.
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the EKU field.
    - S/MIME certificates presented for email encryption and signature shall have the Email Protection purpose (id-kp 4 with OID 1.3.6.1.5.5.7.3.4) in the EKU field.
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the EKU field.
    - Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the EKU field.

FIA_X509_EXT.1.2 The application shall treat a certificate as a CA certificate only if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1 The application shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [TLS].

FIA_X509_EXT.2.2 When the application cannot establish a connection to determine the validity of a certificate, the application shall [not accept the certificate].

5.3.4 Security Management (FMT)

FMT_MEC_EXT.1 Supported Configuration Mechanism

FMT_MEC_EXT.1.1 The application shall [invoke the mechanisms recommended by the platform vendor for storing and setting configuration options].

FMT_CFG_EXT.1 Secure by Default Configuration

FMT_CFG_EXT.1.1 The application shall provide only enough functionality to set new credentials when configured with default credentials or no credentials.

FMT_CFG_EXT.1.2 The application shall be configured by default with file permissions which protect the application binaries and data files from modification by normal unprivileged users.

FMT_SMF.1 Specification of Management Functions

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

- [*start/stop service, uploading of certificates*]

].

5.3.5 Privacy (FPR)

FPR_ANO_EXT.1 User Consent for Transmission of Personally Identifiable Information

FPR_ANO_EXT.1.1 The application shall [

- not transmit PII over a network,

] .

5.3.6 Protection of the TSF (FPT)

FPT_API_EXT.1 Use of Supported Services and APIs

FPT_API_EXT.1.1 The application shall use only documented platform APIs.

FPT_AEX_EXT.1 Anti-Exploitation Capabilities

FPT_AEX_EXT.1.1 The application shall not request to map memory at an explicit address except for [

- *vsyscall*

].

FPT_AEX_EXT.1.2 The application shall [not allocate any memory region with both write and execute permissions].

FPT_AEX_EXT.1.3 The application shall be compatible with security features provided by the platform vendor.

FPT_AEX_EXT.1.4 The application shall not write user-modifiable files to directories that contain executable files unless explicitly directed by the user to do so.

FPT_AEX_EXT.1.5 The application shall be built with stack-based buffer overflow protection enabled.

FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1 The application shall [leverage the platform] to check for updates and patches to the application software**.**

FPT_TUD_EXT.1.2 The application shall [provide the ability] to query the current version of the application software.

FPT_TUD_EXT.1.3 The application shall not download, modify, replace or update its own binary code.

FPT_TUD_EXT.1.4 Application updates shall be digitally signed such that the application platform can cryptographically verify them prior to installation.

FPT_TUD_EXT.1.5 The application is distributed [as an additional software package to the platform OS].

FPT_TUD_EXT.2 Integrity for Installation and Update

FPT_TUD_EXT.2.1 The application shall be distributed using [the format of the platform-supported package manager].

FPT_TUD_EXT.2.2 The application shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events.

FPT_TUD_EXT.2.3 The application installation package shall be digitally signed such that its platform can cryptographically verify them prior to installation.

Application note: This SFR was altered by TD0628.

FPT_LIB_EXT.1 Use of Third-Party Libraries

FPT_LIB_EXT.1.1 The application shall be packaged with only [*no third-party libraries*].

FPT_IDV_EXT.1 Software Identification and Versions

FPT_IDV_EXT.1.1 The application shall be versioned with [[*version number*]].

5.3.7 Trusted Path/Channel (FTP)

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1 The application shall [

- encrypt all transmitted [sensitive data] with [TLS as a client as defined in the Functional Package for TLS, SSH as defined in the Functional Package for Secure Shell]

] between itself and another trusted IT product.

5.4 Assurance Requirements

24 The TOE security assurance requirements are summarized in Table 9.

Table 9: Assurance Requirements

Assurance Class	Components	Description
Security Target Evaluation	ASE_CCL.1	Conformance Claims
	ASE_ECD.1	Extended Components Definition
	ASE_INT.1	ST Introduction
	ASE_OBJ.1	Security Objectives for the Operational Environment
	ASE_REQ.1	Stated Security Requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE Summary Specification
Development	ADV_FSP.1	Basic Functional Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM Coverage
	ALC_TSU_EXT.1	Timely Security Updates (as defined in PP_APP)
Tests	ATE_IND.1	Independent Testing – Conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability Analysis

6 TOE Summary Specification

6.1 Timely Security Updates

25 CyberArk endeavors to remediate critical and high severity publicly disclosed vulnerabilities in its TOEs, in accordance with their severity as implemented in the TOE, and subject to patches made available by their respective vendors (if applicable). The security updated can be provided as quickly as 4 weeks.

26 CyberArk will report a vulnerability to its customers when customers are required to take action to apply the remediation. Reporting of vulnerability-related issues may be via a security bulletin, release notes, knowledge base article, in-product notification or any other appropriate notification method. For the protection of CyberArk's customers, reporting of a vulnerability (including disclosure to any individual customer) will only be made once a remediation is made generally available by CyberArk, unless otherwise required by applicable law or regulation. In addition, the level of detail regarding a vulnerability in any reporting will be limited only to the minimum necessary.

27 If a security bulletin is issued, notification is sent via email to our technical subscribers (defined per customer upon request) and also published on the CyberArk website - Product Security | CyberArk, leading to a password-protected technical community - Login (site.com). First time users are asked to register prior to login.

6.2 SFR Fulfilment

28 Table 10 describes how the TOE fulfils the SFRs.

Table 10: SFR Fulfilment / TOE Summary Specification

SFR	Fulfilment
FCS_CKM.1/AK	Table 11 below lists all the key sizes used for ECC asymmetric key generation schemes and the usage of each key. Table 11 also lists the key establishment and key exchange schemes used by the TOE. The TOE uses key generation and establishment/exchange with the TLS and SSH protocols. The use of asymmetric encryption is needed for the TLS and SSH protocols used by the TOE. The key generation methods follow the requirements within FIPS PUB 186-4. The key establishment methods follow the requirements within NIST Special Publication 800-56A.
FCS_CKM_EXT.1	
FCS_CKM.2	
FCS_COP.1/SKC	Table 11 lists all the key sizes used for AES encryption and decryption within the TOE. Encryption and decryption operations are limited to being used in TLS, SSH, and protecting passwords in credential files. The TOE uses AES-CBC and AES-GCM in its TLS connections. AES is included in the TOE's cryptographic libraries that are statically linked to their components. The cryptographic algorithm follows NIST SP 800-38A (CBC and CTR) and NIST SP 800-38D (GCM). The cryptographic key sizes are 128-bit and 256-bit for all modes.
FCS_COP.1/Hash	Table 11 lists all the key sizes used for SHA hashing and message digests within the TOE. Usage of SHA is limited to TLS and SSH

SFR	Fulfilment
	connections. The TOE's implementations of SHA follow the requirements within FIPS Pub 180-4.
FCS_COP.1/Sig	Table 11 lists all the key sizes used for signature generation and verification within the TOE. Signature generation is used in TLS and SSH connections. Signature verification is used in TLS and SSH connections. The TOE's implementations of signature generation and verification follow the requirements within FIPS PUB 186-4.
FCS_COP.1/KeyedHash	Table 11 lists all the key sizes used for HMAC message authentication within the TOE. Usage of HMAC is limited to TLS and SSH connections. The TOE's implementations of HMAC follows the requirements within FIPS Pub 180-4.
FCS_RBG_EXT.1	The TOE uses OpenSSL in FIPS mode for all cryptographic operations. OpenSSL's FIPS mode DRBG conforms to the NIST Special Publication 800-90A requirements. OpenSSL's FIPS mode implementation of CTR_DRBG is AES-256. The DRBG is seeded by an entropy source that accumulates entropy from a platform-based RNG and no other noise source with a minimum of 256 bits of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and hashes that it will generate. When the DRBG requires entropy bits, it uses the **RAND_seed()** function to fetch entropic bits from the blocking /dev/random device driver. The number of bits requested is compared to the entropy counter value. • If there is less entropy in the Output Entropy Pool than requested by **RAND_seed**, then the DRBG attempts to transfer bits from the Input Entropy Pool to the Output Entropy Pool. • If the Input Entropy Pool has sufficient entropy, then entropy is extracted from the Input Entropy Pool and input to the Output Entropy Pool. • If there is not enough entropy available in the Input Entropy Pool to provide to the Output Entropy Pool, then the DRBG waits until sufficient entropy is available. The SP 800-90A CTR_DRBG requires a minimum entropy input length of 256 bits for instantiation and reseeding. The 256 bits of entropy are requested during the DRBG's instantiation using the **get_entropy** function. Entropy is only extracted from /dev/random during the instantiation and reseed operations for the DRBG. Subsequent random number requests continue to use the output of the properly seeded and instantiated DRBG. The DRBG does require a reseed at an interval, which, as specified in SP 800-90A, must be <= 2^32 or 2^48 (for higher strength algorithms). Therefore, even at the lower value of 2^32, 4.2 billion DRBG generate operations may occur before a reseed is required. The amount of time between the required reseed operations provides more than ample time for enough entropy to be gathered for the next **/dev/random** call.

SFR	Fulfilment
FCS_SSH_EXT.1 FCS_SSHC_EXT.1	The OpenSSH Client component implements public key-based authentication as described in RFC 4252, and keyboard-interactive-based authentication as described in RFC 4256. The OpenSSH Client uses the TOE's cryptographic library for ECDSA-SHA2-NISTP256 and ECDSA-SHA2-NISTP384 as its public key algorithms and rejects all other public key algorithms. The TOE's encryption algorithms used by the OpenSSH Client for SSH transport include AES128-CTR, AES256-CTR, AES128-CBC, and AES256-CBC. All other encryption algorithms are rejected. No other optional characteristics are used for the encryption and public key algorithms. The OpenSSH Client uses the TOE's HMAC-SHA2-256, and HMAC-SHA2-512 as its data integrity MAC algorithms used in SSH transport. All other MAC algorithms are rejected. The key exchange algorithms used by the OpenSSH Client include ECDH-SHA2-NISTP256 and ECDH-SHA2-NISTP384. No other key exchange methods can be used for the SSH protocol. • The OpenSSH Client component ensures that packets greater than 262144 bytes are dropped from the SSH transport connection. • 262144 bytes include **packet_length**, **padding_length**, **payload**, **random padding**, and **mac**. The OpenSSH Client also ensures that the SSH connection is rekeyed after 1 gigabyte of data has been transmitted or 1 hour of connection time has passed, whichever happens first. The OpenSSH Client component ensures that the identity of the SSH server is authenticated using a local database that associates each host name with its corresponding public key.
FCS_STO_EXT.1	The TOE securely stores credentials in non-volatile memory for the following accounts: • **PSMPappuser** – This is the Digital Vault Server account that runs the PSMP application. It is located in the /etc/opt/CARKpsmp/Vault/ folder in the **psmpappuser.cred** file. • **PSMPgwuser** – This is the Digital Vault Server account that runs the PSMP gateway. It is located in the /etc/opt/CARKpsmp/Vault/ folder in the psmpgwuser.cred file. The TOE encrypts file keys using 256-bit AES-GCM (as defined in NIST SP 800-38D).
FCS_TLS_EXT.1 FCS_TLSC_EXT.1	The TOE implements a TLS v1.2 client according to RFC 5246 using its CyberArk PAM TLS Library for Linux. This functionality is only used to communicate to the Digital Vault Server over TLS. Only the following cipher suites are suggested by the TOE for communications to the Digital Vault Server: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

SFR	Fulfilment
	TLS_ECDHE_ECDSA_WITH_AES_128-GCM-SHA256 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 From these, the Vault server does not support ECDSA certificates. From the remaining four suites, the vault server will always select the strongest one available, which is TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384. The TOE components communicate only with the Vault Server. Therefore, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 is the only ciphersuite used. As part of establishing a TLS connection as a client, the TOE verifies that the presented identifier in the peer certificate is a valid reference identifier according to RFC 6125. The reference identifier is established by the TOE. The acceptable reference identifiers for the TOE are a Common Name or IP address for the Subject Name field. The Common Name field of the Digital Vault Server's certificate may contain its IP address because the Digital Vault Server is on a hardened machine that is not necessarily accessible via DNS43. The use of wildcards in the Subject Name is not supported. Certificate pinning is also not supported. The TOE will only establish a TLS connection if the peer certificate is valid.
FCS_TLSC_EXT.5	The TOE implements TLS v1.2 with support for EC algorithms. It supports the use of secp256r1 and secp384r1 EC Extensions to protect communications. The support of these curves is enabled by default for the TLS connection to the Digital Vault Server and configured during hardening of the server.
FDP_DEC_EXT.1 FDP_NET_EXT.1	The TOE limits its access to only network connectivity when accessing the platform's hardware resources. The TOE requires network access to the SSH client, Digital Vault Server, and target devices. 1. The user initiates the connection from their SSH client to the PSMP Client TSFI over port 22 when they want to SSH to a target machine, requiring the TOE to use the OpenSSH Server to use network resources. 2. The TOE connects to the Digital Vault Server and sends the authentication information over TLS on port 443 for verification. 3. If the authentication passes, PSMP then connects to the target device with the supplied credentials using OpenSSH Client over an SSH connection on port 22. The TOE uses port 80 for HTTP connections to the CA server for certification revocation checks from each component. The TOE also

SFR	Fulfilment
	connects to the Digital Vault Server periodically over TLS on port 443 to download ACLs and configuration information. The TOE limits its access to the sensitive information repository, which includes the SELinux logs. The SELinux logs are accessed when the TOE needs to store related event data.
FDP_DAR_EXT.1	Sensitive data within the TOE is limited to the credentials for service accounts. The TOE limits the storing of sensitive data in non-volatile memory to only passwords for the **psmpappuser** and **psmpgwuser** service accounts. Each account's password is encrypted using AES256-CBC before it is saved. Other credentials that are used to authenticate with the Digital Vault Server are never stored by the TOE and transferred to and from the TOE over secure connections. PSMP recordings may contain sensitive information if the user chooses to connect to a remote target which may contain sensitive information, therefore, we recommend the user to enable LUKS.
FIA_X509_EXT.1	The TOE provides its own implementation of TLS to perform certificate validation. The TOE is a client to the Digital Vault Server and each validates the Digital Vault Server's X.509v3 certificate during TLS authentication. The TOE ensures that the X.509v3 certificate adheres to RFC 5280 (certificate validation and certificate path validation) and that the certificate path terminates with a trusted CA certificate. The TOE treats a certificate as a CA certificate when the certificate includes the **basicConstraints** extension and verifies that the CA flag is set to TRUE for all CA certificates. The TOE validates the revocation status of the Digital Vault Server's TLS certificate according to RFC 5759 using a CRL when establishing the TLS connection. The CRL is downloaded from the CA server in the operating environment. The path to the CRL is read from the certificate's CRL Distribution Point (CDP) field. The TOE checks the Digital Vault Server certificate against the downloaded CRL and automatically rejects the certificate if it is found to be invalid. When a TLS v1.2 connection cannot be established because the validity check of a certificate fails, the connection is aborted. The TOE validates that the Digital Vault Server's server certificate presented for TLS has the Server Authentication purpose in the extended key usage field. The TOE does not accept S/MIME, OCSP or EST certificates. The TOE supports a maximum trust depth of two nodes.
FIA_X509_EXT.2	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication. Both components of the TOE read the location of their certificates from their vault.ini files using the **ClientCertificate** parameter.

SFR	Fulfilment
	The TOE validates X.509v3 certificates from the Digital Vault Server for TLS authentication. This functionality is enabled by default. The path to the CRL is read from the certificate's CRL Distribution Point (CDP) field. The TOE's implementation of TLS automatically rejects a certificate if it is found to be invalid according to the requirements in FIA_X509_EXT.1. It also rejects a certificate with an unknown revocation status due to the inability to establish a connection to the CDP. The connection from the TOE to the CDP is conducted over HTTP as per the RFC.
FMT_MEC_EXT.1	The TOE does not write or set any configuration options using local configuration files. All configuration settings are stored in a Safe on the Digital Vault Server and can be configured using the Password Vault Web Access interfaces. The TOE contains local configuration files that are created during installation, but the information is read-only and never written to by the TOE. Local configuration information related to the PSMP component is stored in the **/etc/opt/CARKpsmp/conf/basic_psmpserver.conf** file.
FMT_CFG_EXT.1	No default user credentials are provided by the TOE. Once the TOE is installed, it is connected to the Digital Vault Server for all authentication needs. The files created during installation are set with default permissions. The Linux permissions for **other** are set to **0** on all files except the ones used to launch the TOE, which have read and execute permissions, to protect the files from modifications made by unprivileged user.
FMT_SMF.1	The TOE does not provide any methods of configuration management through PSMP because this component reads its configurations from the Digital Vault Server. Other management functions are that the TOE provides are the following start/stop service, uploading of certificates.
FPR_ANO_EXT.1	The TOE does not collect PII for administrators or users. Therefore, there is no case in which the TOE will transmit this data over the network.
FPT_API_EXT.1	The TOE only uses supported platform APIs in order to function. The list in Annex A: List of Platform APIs includes all the platform APIs used by the PSMP component. To read more about each system call used by PSMP, refer to its reference in the Linux manual pages (also known as man pages).
FPT_AEX_EXT.1	The TOE does not request memory mappings at explicit addresses except for the usage of vsyscall. Vsyscall is a legacy memory segment that was added as a way to run specific system calls that do not need any real level of privilege to run,

SFR	Fulfilment
	such as gettimeofday. It has a fixed address ffffffffff600000 and only has read and execute permissions. When the TOE is being complied, it uses the -**fPIC** flag to enable ASLR. The TOE does not allocate any memory region with both write and execute permissions. No just-in-time compilations are performed by the TOE. The TOE is also complied using the -**fstackprotector**=all flag to enable stack-based buffer overflow protection. The TOE is compatible with SELinux and use SELinux profile files that are applied during installation. The TOE does not write user-modifiable files to directories that contain executable files. User modifiable files are written to the **/etc/opt/CARKpsmp/** and **/var/opt/CARKpsmp/**" folders. Executable files are stored in the **/opt/CARKpsmp/** folder.
FPT_TUD_EXT.1/ 2 FPT_IDV_EXT.1	The TOE is delivered through CyberArk's online customer portal, which uses AWS Marketplace. The TOE installation and configuration files are all packaged into a zip file that is digitally signed by CyberArk. To verify the integrity of the TOE installation file, do the following: 1. Run `rpm --import RPM-GPG-KEY-CyberArk.` 2. Run `rpm -K -v on the package.` Individual TOE files are signed using the Linux OS package manager MS21 Sign tool, and can be verified using `sha256sum.` The authorized signing source is CyberArk. To determine the currently installed version, the administrator can run the following command: **`rpm -q CARKpsmp`** Versioning naming convention: AA.B.C.DD (e.g: 14.0.0.32) - AA – Major Version Number – 14 - B – Minor Version Number – 0 - C – Patch Number – 0 - DD – Build Number - 32
FPT_LIB_EXT.1	The TOE is not packaged with any third-party libraries.
FTP_DIT_EXT.1	The TOE protects data in transit by providing trusted paths and channels using the cryptographic functions within the TOE's cryptographic libraries. The TOE provides a trusted channel between itself and target devices over SSH. The TOE uses OpenSSH Client to create this SSH connection, which provides support using AES-128-CTR, AES-256- CTR, AES-128-CBC, or AES-256-CBC for encrypting its traffic. The SSH connection used by the TOE follows the Extended Package for Secure Shell.

SFR	Fulfilment
	The TOE provides a trusted TLS channel between itself and the Digital Vault Server using OpenSSL. The TOE acts as a TLS client to connect to the Digital Vault Server over TLS when access Safes that are stored in Digital Vault Server.

Table 11: Cryptographic Algorithms

Operation	Usage	Algorithm	Key Size
Encryption/Decryption	Secure Storage	AES-CBC	256
	TLS	AES-GCM	128, 256
	SSH (client)	AES-CBC and AES-CTR	128, 256
Key Generation	Safe	AES CTR-DRBG	256
	SSH (client)	ECDSA	P256, P384
Signature Generation Signature Verification	TLS	RSA	2048, 3072, 4096
	SSH (client)	ECDSA	P256, P384
Key Exchange /Establishment	TLS	ECDHE, DHE	256, 384 3072
	SSH (client)	ECDH	256, 384
Message Digest	TLS	SHA-256, SHA-384, SHA-512	256, 384, 512
	SSH (client)	SHA-256, SHA-384	256, 384
Message Authentication	TLS	HMAC-SHA-256, HMAC-SHA-384	256, 384
	SSH (client)	HMAC-SHA-256, HMAC-SHA-512	256, 512
Random Number Generation	TOE DRBG	CTR DRBG (AES)	N/A

Table 12: HMAC

Hash Function	Block Size	Key Length	Output Digest
SHA256	512	256	256
SHA384	1024	384	384
SHA512	1024	512	512

7 Rationale

7.1 Conformance Claim Rationale

29 The following rationale is presented with regard to the PP conformance claims:

a) **TOE type.** As identified in section 1.2.1, the TOE is an application, consistent with the PP.

b) **Security problem definition.** As shown in section 3, the threats, OSPs and assumptions are reproduced in this ST.

c) **Security objectives.** As shown in section 4, the security objectives are reproduced in this ST.

d) **Security requirements.** As shown in section 5, the security requirements are reproduced from the PP. No additional requirements have been specified.

7.2 Security Objectives Rationale

30 All security objectives are drawn directly from the claimed PP.

Table 13: Security Objectives Rationale

Threat, Assumption, or OSP	Security Objectives	Rationale
T.NETWORK_ATTACK	O.PROTECTED_COMMS, O.INTEGRITY, O.MANAGEMENT	The threat T.NETWORK_ATTACK is countered by O.PROTECTED_COMMS as this provides for integrity of transmitted data. The threat T.NETWORK_ATTACK is countered by O.INTEGRITY because this provides for integrity of software that is installed onto the system from the network. The threat T.NETWORK_ATTACK is countered by O.MANAGEMENT because this provides for the ability to configure the application to defend against network attack.
T.NETWORK_EAVESDROP	O.PROTECTED_COMMS, O.QUALITY, O.MANAGEMENT	The threat T.NETWORK_EAVESDROP is countered by O.PROTECTED_COMMS because this provides for confidentiality of transmitted data. The objective O.QUALITY ensures use of mechanisms that provide protection against network-based attack.

Threat, Assumption, or OSP	Security Objectives	Rationale
		The threat T.NETWORK_EAVESDROP is countered by O.MANAGEMENT because this provides for the ability to configure the application to protect the confidentiality of its transmitted data.
T.LOCAL_ATTACK	O.QUALITY	The objective O.QUALITY protects against the use of mechanisms that weaken the TOE with regard to attack by other software on the platform.
T.PHYSICAL_ACCESS	O.PROTECTED_STORAGE	The objective O.PROTECTED_STORAGE protects against unauthorized attempts to access physical storage used by the TOE.
A.PLATFORM	OE.PLATFORM	The operational environment objective OE.PLATFORM is realized through A.PLATFORM.
A.PROPER_USER	OE.PROPER_USER	The operational environment objective OE.PROPER_USER is realized through A.PROPER_USER.
A.PROPER_ADMIN	OE.PROPER_ADMIN	The operational environment objective OE.PROPER_ADMIN is realized through A.PROPER_ADMIN.

7.3 Security Requirements Rationale

31 All security requirements are drawn directly from the claimed PP.

Annex A: List of Platform APIs

accept

accept4

access

bind

chdir

chmod

chown

clock_gettime

close

connect

dup

dup2

epoll_create

epoll_create1

epoll_ctl

epoll_wait

eventfd

execv

execve

exit

fchmod

fchown

fcntl

fork

ftruncate64

getcwd

getegid

geteuid

getgid

gethostname

getpagesize

getpeername

getpgid

getpid

getppid

getrusage
getsid
getsockname
getsockopt
gettimeofday
getuid
ioctl
kill
link
listen
lseek
mkdir
mmap
munmap
nanosleep
open
pause
pipe
poll
ppoll
read
readdir
readlink
readv
recv
recvfrom
recvmsg
rename
rmdir
sched_yield
select
semctl
semget
semop
send
sendmsg
sendto

setgroups
setresgid
setresuid
setsid
setsockopt
shutdown
sigaction
signal
signalfd
sigprocmask
socket
socketpair
statfs64
symlink
syscall
sysinfo
syslog
time
timerfd_create
timerfd_settime
truncate64
umask
unlink
utime
waitpid
write
writev