CyberArk Software Ltd.

Privileged Access Security – Windows Components

Including Privileged Session Manager (PSM) v10.4, Central Policy Manager (CPM) v10.4, and Password Vault Web Access (PVWA) v10.4

Security Target

Document Version: 0.15

Prepared for:

CYBERARK

CyberArk Software Ltd.
9 Hapsagot St. Park Ofer 2
P.O.B. 3143
Petach-Tikva 4951040
Israel

Phone: +1 888 808 9005
www.cyberark.com

Prepared by:

Corsec

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

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

Table of Contents

List of Figures

List of Tables

1. Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the organization of the ST. The TOE is the CyberArk Software Ltd. (CyberArk) Privileged Access Security – Windows Components, including Privileged Session Manager (PSM) v10.4, Central Policy Manager (CPM) v10.4, and Password Vault Web Access (PVWA) v10.4, and will hereafter be referred to as the TOE throughout this document. The TOE is a software-based solution that runs on Windows and is a component of CyberArk's Privileged Access Security (PAS) Solution. PAS enables organizations to secure, provision, control, and monitor all activities associated with privileged identities used in enterprise systems and applications. PSM is the part of PAS that enables organizations to secure, control, and monitor privileged access to network devices over RDP connections. CPM automatically enforces enterprise policies for password management. PVWA is the web interface of PAS that provides a single console for requesting, accessing, and managing privileged passwords throughout the environment.

1.1 Purpose

This ST is divided into 10 sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functionality and describes the physical and logical scope for the TOE as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any Common Criteria (CC), Protection Profile (PP), and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, organizational security policies, and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Assurance Requirements (Section 6) – Presents the SARs met by the TOE.
- Security Functional Requirements (Section 7) – Presents the SFRs met by the TOE.
- TOE Summary Specification (Section 8) – Describes the security functions provided by the TOE that satisfy the SFRs and objectives.
- Rationale (Section 9) – Presents the rationale for the SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms and Terms (Section 10) – Defines the acronyms and terminology used within this ST.

1.2 Security Target and TOE References

Table 1 below shows the ST and TOE references.

Table 1 – ST and TOE References

ST Title	*CyberArk Software Ltd. Privileged Access Security – Windows Components including Privileged Session Manager (PSM) v10.4, Central Policy Manager (CPM) v10.4, and Password Vault Web Access (PVWA) v10.4 Security Target*
ST Version	Version 0.15
ST Author	Corsec Security, Inc.
ST Publication Date	September 27, 2019
TOE Reference	CyberArk Privileged Access Security – Windows Components including PSM v10.04.100.25, CPM v10.04.10.7, and PVWA v10.04.10.4

1.3 Product Overview

The product is the CyberArk Privileged Access Security (PAS) Solution software suite, which enables organizations to secure, provision, control, and monitor all activities associated with the privileged identities used in enterprise systems.[1] PAS contains multiple applications that work together to provide the following functionality to: configure and administer PAS using a web-based interface; store, manage and control access to privileged accounts; establish connections to remote targets using the privileged account credentials; enforce password policy; control access to privileged commands; and record and securely store administrator and session activities.

The PAS software suite components PVWA, PSM, and CPM provide the following functionality: PVWA provides web-based administrator access to manage and configure PAS, PSM is used to establish an RDP connection to a remote target, and CPM enforces password policy. The other PAS applications provide the functionality to securely store and control access to each privileged account file and its unique key, establish SSH (Secure Shell) connections to remote target devices using the privileged account credentials, and to control access to privileged commands. The PAS applications described below interact with the PSM, PVWA, CPM components to provide the complete functionality of the PAS software suite, but they are not covered by the evaluation. The CyberArk Version Check tool is used to query the current version of PAS software installed on the host and to check if an update is available for the components of the TOE.

The Enterprise Password Vault (EPV) is the central component of the PAS software suite. EPV manages the secure storage and access to the privileged account files, and to the administrator and session activity files. The privileged account files are used to connect to target machines.

The Privileged Session Manager SSH Proxy (PSMP) allows users to obtain privileged account information from EPV, then log the user onto a target device over a secured SSH connection. PSMP records the activities that are performed in the privileged session and uploads the recording to EPV, where they are accessed and viewed by authorized users.

The On-Demand Privileges Manager (OPM) allows users to obtain privileged account permissions and privileged command access from their local Linux session without obtaining the root credentials or super user access. OPM enables users to granularly access and use privilege accounts according to the command permissions, which are created and managed in EPV.

[1] Note that the components of the PAS Solution were not evaluated as a distributed TOE but as standalone TOEs. This Security Target covers the evaluation of PVWA, PSM, and CPM.

1.4 TOE Overview

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

The TOE is the CyberArk Privileged Access Security – Windows Components, which contains the PSM, CPM, and PVWA software and CyberArk Version Check tool, all of which run on a Windows Server operating system (OS).

PSM acts as a gateway for RDP[2]-enabled devices by controlling access to privileged sessions and initiates RDP connections to remote devices on behalf of the user[3] without the need to disclose account credentials. PSM can record sessions in a compact format to provide detailed session audits and DVR[4]-like playback. The recorded sessions are stores in the Enterprise Password Vault (EPV) in the environment. CPM can change passwords automatically within EPV according to the organizational policy and store the new passwords in EPV without human intervention. CPM also enables organizations to verify passwords on remote machines and reconcile them when necessary. PVWA enables administrators[5] to define new privileged passwords and search to find privileged passwords or sensitive files with minimum effort. PVWA's dashboard enables administrators to see an overview of activity in the PAS Solution as well as statistics about all the activities that have taken place. The dashboard shows a graphic representation of the passwords that have been managed and links to specific information about accounts and passwords that require special attention. The CyberArk Version Check tool is used to query the current version of PAS software installed on the host and to check if an update is available for the components of the TOE.

All users and administrators must provide their credentials and be authenticated before accessing the TOE. The TOE will only retrieve privileged credentials for an authenticated and authorized user or administrator.

Sections 1.4.1, 1.4.2, 1.4.3, 1.4.4, and 1.5 respectively describe the software components of PSM, CPM, PVWA, CyberArk Version Check tool, and the components of the TOE environment in detail.

1.4.1 PSM

The PSM component allow users to obtain privileged account information through its PSM Client TSFI[6] to enable users to log onto remote devices over a secure RDP connection. PSM separates the users from targets, enabling connections to privileged devices without having to divulge the passwords to the user. PSM records the activities that are performed in the privileged session and uploads the recording to EPV, where they are accessed and viewed by authorized users.

PSM retrieves credentials from EPV by using TLS[7]. PSM enforces cryptographic support functionality of the TOE for securing the communication between the TOE and the EPV server. PSM uses a credential file to securely store its credentials to authenticate to EPV, and once authenticated, PSM communicates with EPV for retrieving the credentials for the remote targets. Communications between PSM and EPV are conducted over TLS through port

[2] RDP – Remote Desktop Protocol
[3] A user of the TOE is any entity with an account that uses the TOE for non-manage activities like connecting to PSM to remotely control a remote target.
[4] DVR – Digital Video Recorder/Recording
[5] An administrator of the TOE is any entity with an account that has permissions to manage the areas of the TOE listed in section 7.2.4.
[6] TSFI – TOE Security Function Interface
[7] TLS – Transport Layer Security

1858. PSM is compiled with the OpenSSL FIPS[8] Object Module[9] v2.0.14 that includes the CyberArk libraries, CyberArk PAS Cryptographic Library for Windows v1.0 and CyberArk PAS TLS Library for Windows v1.0, for its cryptographic functionality. PSM will also leverage the TLS connection of the RDP Client in the OE when connecting to a remote target. The RDP Client relies on the cryptographic functionality of the Windows cryptographic library to secure the TLS connection.

1.4.2 CPM

The CPM component is responsible for ensuring that secure passwords are used and created for all accounts within EPV. The PVWA Client TSFI must be used to configure the policies that CPM enforces as there is no direct access to CPM. Administrators can configure security and compliance policies for all accounts' passwords. The policies, which specify minimum password requirements such as length, expiration, complexity, and others, are stored in EPV. CPM enforces policies by automatically changing passwords and storing the passwords within EPV. All passwords that are monitored and generated by CPM conform to the Master Policy that was created by the administrator. Administrators will be notified via the PVWA Client TSFI when passwords are about to expire, are expired, or are in violation of the Master Policy criteria. Administrators can also implement a one-time password policy that requires a password to be changed after each login.

CPM uses a credential file to securely store its credentials to authenticate to EPV, and once authenticated, CPM communicates with EPV for retrieving and updating the passwords and password policies on the remote targets. The communication between CPM and EPV is conducted over TLS through port 1858. CPM is compiled with the OpenSSL FIPS Object Module v2.0.14 that includes the CyberArk libraries, CyberArk PAS Cryptographic Library for Windows v1.0 and CyberArk PAS TLS Library for Windows v1.0, for its cryptographic functionality. CPM records all password policy configuration activities in a local log file and sends the logs to EPV for storage.

1.4.3 PVWA

PVWA enables administrators to access and configure the PAS Solution remotely over a web browser by providing access to policy and platform management features. PVWA identifies the administrator during authentication by checking the submitted credentials against what is stored in EPV or by having EPV check the credentials against the external authentication server. PVWA allows administrators to define access control rules on credentials and platforms and to configure the Master Policies in EPV. PVWA uses a credential file to securely store its credentials to authenticate to EPV, and once authenticated, PVWA communicates with EPV for authenticating accounts used to access its interfaces and uploading any configuration changes to EPV.

Administrators that access the PVWA Client TSFI must have a valid account and the management workstation must be located within the same IP[10] subnet. The communication between PVWA and EPV conducted over TLS through port 1858. PVWA is compiled with the OpenSSL FIPS Object Module v2.0.14 that includes the CyberArk libraries, CyberArk PAS Cryptographic Library for Windows v1.0 and CyberArk PAS TLS Library for Windows v1.0, for its cryptographic functionality.

[8] FIPS – Federal Information Processing Standard
[9] Note that the OpenSSL FIPS Object Module is the name of the component created by OpenSSL and used with CyberArk's CAVP-validated cryptographic libraries. It is not meant to imply that this product had completed the CMVP validation.
[10] IP – Internet Protocol

PVWA enables administrators to access and manage privileged accounts on EPV. As administrators access, manage, and configure EPV, their activities are tracked and time-stamped. The activities can be filtered according to identity, target system, specified time interval, and a variety of other criteria.

PVWA also includes the PVWA RESTful[11] API[12] that allows administrators to create, list, modify, and delete entities in the PAS Solution. The PVWA RESTful API allows PVWA Client TSFI tasks to be automated and scripted. Any RESTful API client can access this interface over an HTTPS[13] connection. All commands on the PVWA RESTful API are authenticated. PVWA relies on IIS in the OE to provide the TLS connection for the PVWA Client TSFI and PVWA RESTful API. The IIS service relies on the Windows cryptographic library for its cryptographic functionality to secure the HTTPS protocol.

1.4.4 CyberArk Version Check Tool

The CyberArk Version Check tool is a script used to query the current versions of the installed PAS components and checks if an update is available for the found components. The tool relies on the EPV server to store the file it will use to check for the latest version.

1.5 TOE Environment

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 center with the appropriate level of physical access control and physical protection (e.g., badge access, fire control, locks, alarms, etc.).

In the evaluated configuration, the TOE is a part of CyberArk's PAS Solution. PSM, CPM, and PVWA are installed on a single instance of Microsoft Windows Server 2012 R2[14]. IIS[15] in the operating environment (OE) is used by PVWA for serving its web interface. PSM requires RDP services and RDP client in the OE for communicating with users and targets. An instance of EPV is part of the OE and is installed on a standalone Windows 2012 R2 server for access control usage by PSM, CPM, and PVWA. The CyberArk Version Check tool is also downloaded to the host from the CyberArk support vault and relies on the EPV server for storing the file that contains the latest version information.

To interact with the EPV server, all TOE components require access to the same network in which the EPV server is installed. Note that the platform that the TOE is installed on will not be allowed to have access to the internet and is intended for only intranet use. PSM will be used to connect to remote targets in the OE using secure RDP to allow users to securely interact with devices on the network. CPM will be used to connect to the EPV server using TLS to allow password management for stored accounts. PVWA will be used to manage all components in the PAS solution by connecting to the EPV server using TLS and updating their configurations.

[11] RESTful – Representational State Transfer
[12] API – Application Program Interface
[13] HTTPS – Hypertext Transport Protocol Secure
[14] R2 – Release Two
[15] IIS – Internet Information Service

The TOE communicates with the EPV server in the OE, which is on the same isolated network as the TOE. EPV is installed on a hardened[16] version of Microsoft Windows Server 2012 R2. Access to and from the EPV server is available to only components of the PAS Solution via a secure channel over TLS v1.2. EPV is used by the TOE for authenticating users/administrators, updating credentials, storing TOE data, and storing monitored session information.

Access to PSM can be obtained by a user either through the PVWA Client TSFI or directly from an RDP client. When a user accesses PSM through PVWA, they must select the account to use to log onto the target system, and the native protocol to use for this connection. Then PVWA redirects the user to PSM that will allow access to the desired target system. A user accesses PSM from an RDP client by connecting to the PSM server using the RDP protocol over TLS. The RDP service on the PSM server receives the connection and PSM interoperates the request. PSM uses TLS to connect to the EPV server when it authenticates the user and fetches the required account credentials for accessing the remote target. If the credentials are successfully returned, PSM connects to the remote target using the supplied credentials and the OE's RDP client over a secure RDP connection. The activities that are performed in the privileged session are recorded by PSM and uploaded to EPV, where they can be accessed and viewed by authorized administrators.

Communications between PSM and EPV are conducted over TLS through port 1858. PSM is compiled with the OpenSSL FIPS Object Module v2.0.14 with the CyberArk libraries for its cryptographic functionality.

Users initiate a connection to the PSM server through port 3389 using the RDP client to the RDP server. PSM will then proxy the incoming connect to the remote targets using the RDP client in the OE and secure the RDP protocol using TLS. The RDP client on the PSM server relies on the cryptographic functionality of the Windows cryptographic library to secure the TLS connection to the remote target.

Access to CPM is provided by the PVWA Client TSFI as there is no direct access to CPM. Administrators connect to the PVWA Client TSFI using HTTPS for a secure connection. CPM uses TLS to connect to the EPV server to query for updated password policies and to update any credentials that it changes.

Access to PVWA can be obtained by an administrator either through the PVWA Client TSFI or PVWA RESTful API. Administrators connect to the PVWA Client TSFI or PVWA RESTful API using HTTPS for a secure connection. PVWA relies on the OE's IIS service to perform the HTTPS connection between the server and workstation. PVWA uses TLS to connect to the EPV server to authenticate administrators and to upload any configuration changes to the PAS components.

Table 2 describes the requirements for the components found in the TOE environment.

[16] Protected by a firewall that only allows remote communications from CyberArk applications via a secure channel.

Table 2 – Environmental Components

Component	Requirements
Windows Components Server	Operating System: • Microsoft Windows Server 2012 R2 Hardware: • Intel i7-6700 or Intel Xeon E5 family processor Required Platform Applications: • IIS 8.5 • .NET Framework 4.5.2 • Remote Desktop Services (RDS) Session Host • Remote Desktop Client
EPV Server	Operating System: • Microsoft Windows Server 2012 R2 Hardware: • Intel i7-6700 or Intel Xeon E5 family processor Required Platform Applications: • CyberArk EPV v10.4 • .NET Framework 4.5.2
Web Browser Workstation	At least one of the following browsers: • Internet Explorer (IE) 8, 9, 10 or 11 • Firefox (latest version) • Chrome (latest version) Jar signer for verifying installation packages. A RESTful API client, such as Postman, RESTClient, Advanced REST Client, etc. Windows RDP Client for use with PSM.
Remote Target	At least one of the following: • Any target with Windows RDP • Any target with Windows Remotely Anywhere

1.6 TOE Description

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

1.6.1 Physical Scope

Figure 1 illustrates the physical scope and the physical boundary of the overall solution and ties together all the components of the software-only TOE as well as the constituents of the TOE Environment.

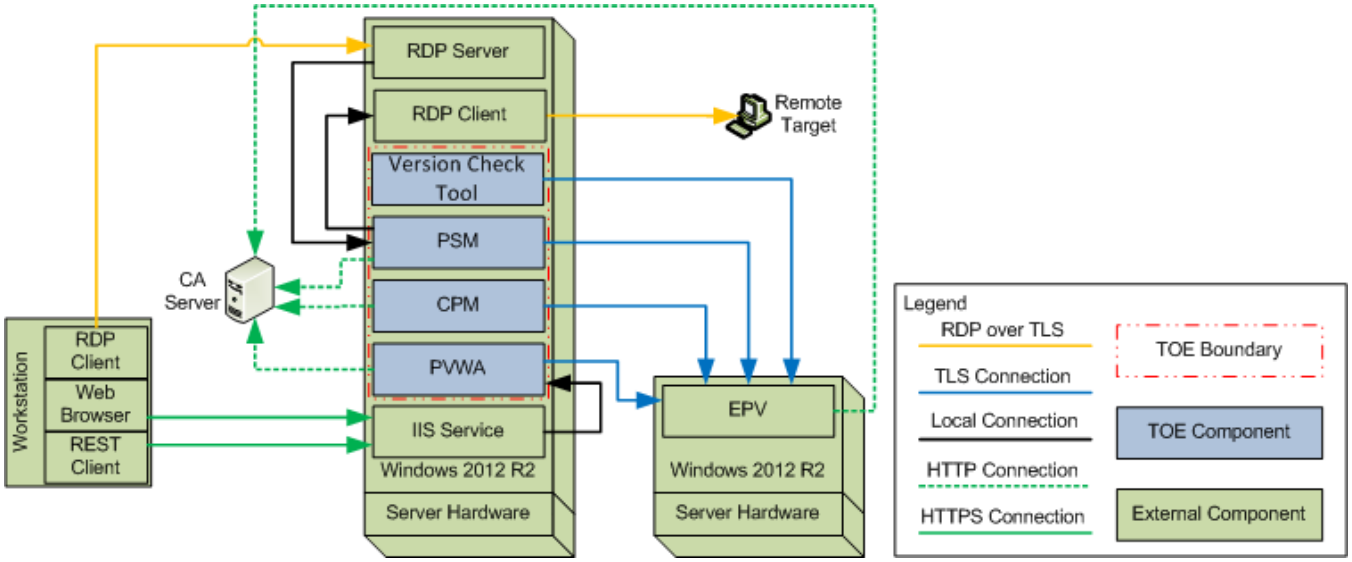

Figure 1 – TOE Boundary

The TOE Boundary includes all the CyberArk developed parts of PSM, CPM, PVWA, and the Version Check tool.

1.6.1.1 TOE Software

The TOE is a software-only TOE and is comprised of the PSM, CPM, PVWA, and Version Check tool software. For the evaluated configuration, the following TOE software must be installed on the Windows machine in the environment:

- CyberArk PSM v10.04.100.25
- CyberArk CPM v10.04.10.7
- CyberArk PVWA v10.04.10.4
- CyberArk Version Check tool v1.5

1.6.1.2 Guidance Documentation

Table 3 lists the TOE guidance documentation to install, configure, and maintain the TOE.

Table 3 – Guidance Documentation

Document Name	Description
CyberArk; Privileged Access Security Installation Guide; Version 10.4; PASINS-10-4-0-1	Includes steps for the basic initialization and setup of the TOE.
CyberArk; Privileged Access Security System Requirements; Version 10.4; PASSR-10-4-0-1	
CyberArk; Hardening the CyberArk CPM and PVWA Servers; Version 10.4; CAHEPV-0617	
CyberArk; Privileged Access Security End-user Guide; Version 10.4; PASEUG-10-4-0-1	Contains detailed steps for how to properly configure and maintain the TOE.
CyberArk; Privileged Access Security Reference Guide; Version 10.4; PASRG-10-4-0-1	
CyberArk; Privileged Access Security Implementation Guide; Version 10.4; PASIMPG-10-4-0-1	
CyberArk Software Ltd.; Privileged Access Security – Windows Components; Guidance Documentation Supplement; Document Version: 0.8	Contains information regarding specific configuration for the TOE evaluated configuration.

1.6.2 Logical Scope

The logical boundary of the TOE will be broken down into the following security classes, which are further described in sections 7 and 8 of this ST. The logical scope also provides the description of the security features of the TOE. The security functional requirements implemented by the TOE are usefully grouped under the following Security Function Classes.

1.6.2.1 Cryptographic Support

The TOE uses CAVP-validated cryptographic algorithm provided by its OpenSSL FIPS Object Module v2.0.14 with CyberArk libraries. The libraries are used to support the establishment of trusted channels and paths to protect data in transit. In the evaluated configuration, the TOE's cryptographic libraries are used by the TLS client connection to the EPV server from PSM, CPM, and PVWA. The TOE provides the cryptographic functionality listed in Table 15 below.

1.6.2.2 User Data Protection

The TOE stores sensitive information in the form of encrypted passwords in non-volatile memory. The TOE will limit its access to only network connectivity when accessing the platform's hardware resources. The network connection is used for communications between the TOE to the EPV server, the TOE to the target devices, and the user/administrator to the TOE.

1.6.2.3 Identification and Authentication

To validate the EPV server's certificate during the TLS handshake, the TOE implements functionality to validate X.509 certificates. The TOE uses a CRL[17] to check certificate revocation status and will not establish a connection to the EPV server when the CRL is unavailable. The same functionality is used by CPM when it connects to the EPV server to manage passwords.

1.6.2.4 Security Management

The TOE is configured with default file permissions already in place and does not provide default credentials for authentication. The TOE relies on PVWA for storing and setting configuration options for PSM and CPM. Administrators can manage various parts of the TOE's functionality using the PVWA interfaces. A list of manageable features is provided in section 8.1.4 below.

1.6.2.5 Privacy

The TOE does not store or transmit any personally identifiable information (PII).

1.6.2.6 Protection of the TOE Security Functionality (TSF)

The TOE protects against exploitation by implementing address space layout randomization (ASLR) except for the OpenSSL hash check and not allocating memory with both writing and execution. The TOE is also compatible with a hardened Windows environment 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 it needs to perform its functionality.

[17] CRL – Certificate Revocation List

The version of each TOE component can be checked using the platform's Programs and Features manager. PVWA also provides its version information in its help section. 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. The installation package for each TOE component is digitally signed using a public key from CyberArk that is used to verify the integrity of the TOE's files.

1.6.2.7 Trusted Path/Channels

The TOE relies on the IIS service in the OE to provide a trusted path for communications to the TOE using TLS. The TOE also relies on the RDP Client in the OE to provide a trusted channel for communications from the TOE to a remote target using TLS. The TOE provides its own trusted channel between each TOE component to the EPV server over TLS.

1.6.3 Product Physical/Logical Features and Functionality not included in the TOE

The following features and functionality are not part of the evaluated configuration of the TOE:

- All CPM password management except for managing CyberArk accounts stored in EPV.
- All PSM connection components except for the PSM-RDP connection component.
- Functions provided by PSMP.
- Functions provided by OMP.
- Functions provided by EPV.

1.6.4 Scope of Evaluation

The evaluation is limited in scope to the secure features described in the *Protection Profile for Application Software v1.2*; April 22, 2016 (AS PP) and detailed in section 1.6. The TOE is conformant to the AS PP and no interpretations apply to the claims made in this ST.

2. Conformance Claims

This section provides the identification for any CC, PP, and Technical Decisions (TD) conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for conformance claims can be found in section 9.1.

Table 4 – CC and PP Conformance

Common Criteria (CC) Identification and Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012; CC Part 2 extended; CC Part 3 extended. PP claim to the *Protection Profile for Application Software v1.2*; April 22, 2016 conformant.
PP Identification	Exact Conformance[18] to the *Protection Profile for Application Software v1.2;* April 22, 2016.
TD Conformance	Conformance to the following AS PP TDs is claimed: • 0434 – Windows Desktop Applications Test • 0392 – FCS_TLSC_EXT.1.2 Wildcard Checking • 0389 – Handling of SSH EP claim for platform • 0382 – Configuration Storage Options for Apps • 0359 – Buffer Protection • 0358 – Cipher Suites for TLS in SWApp v1.2 • 0327 – Default file permissions for FMT_CFG_EXT.1.2 • 0326 – RSA-based key establishment schemes • 0304 – Update to FCS_TLSC_EXT.1.2 • 0300 – Sensitive Data in FDP_DAR_EXT.1 • 0296 – Update to FCS_HTTPS_EXT.1.3 • 0295 – Update to FPT_AEX_EXT.1.3 Assurance Activities • 0293 – Update to FCS_CKM.1(1) • 0268 – FMT_MEC_EXT.1 Clarification • 0267 – TLSS testing – Empty Certificate Authorities list • 0244 – FCS_TLSC_EXT – TLS Client Curves Allowed • 0241 – Removal of Test 4.1 in FCS_TLSS_EXT.1.1 • 0238 – User-modifiable files FPT_AEX_EXT.1.4 • 0221 – FMT_SMF.1.1 – Assignments moved to Selections • 0217 – Compliance to RFC5759 and RFC5280 for using CRLs • 0215 – Update to FCS_HTTPS_EXT.1.2 • 0177 – FCS_TLSS_EXT.1 Application Note Update • 0174 – Optional Ciphersuites for TLS • 0163 – Update to FCS_TLSC_EXT.1.1 Test 5.4 and FCS_TLSS_EXT.1.1 Test • 0131 – Update to FCS_TLSS_EXT.1.1 Test 4.5 • 0119 – FCS_STO_EXT.1.1 in PP_APP_v1.2 • 0107 – FCS_CKM – ANSI[19] X9.31-1998, Section 4.1.for Cryptographic Key Generation

[18] Exact Conformance is a type of strict conformance such that the set of SFRs and the SPD/Objectives are exactly as presented within the accepted PP and Extended PP without changes.

[19] ANSI – American National Standards Institute

3. Security Problem Definition

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It provides the statements for the TOE security environment's threats, assumptions, and organizational security policies (OSPs) as identified in the AS PP.

3.1 Threats

Table 5 describes the threats that the TOE is expected to address as defined in the AS PP.

Table 5 – Threats

Threat	Description
T.LOCAL_ATTACK	An attacker can act through unprivileged software on the same computing platform on which the application executes. Attackers may provide maliciously formatted input to the application in the form of files or other local communications.
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.PHYSICAL_ACCESS	An attacker may try to access sensitive data at rest.

3.2 Assumptions

Table 6 describes the assumptions that are assumed to exist in the TOE's operating environment as defined in the AS PP.

Table 6 – Assumptions

Assumption	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_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.
A.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.

3.3 Organizational Security Policies

There are no OSPs defined in the AS PP.

4. Security Objectives

This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security Objectives for the TOE

Table 7 describes the security objectives that the TOE is required to meet as defined in the AS PP.

Table 7 – Security Objectives for the TOE

Objective	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, and 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. Addressed by: FDP_DEC_EXT.1, FMT_CFG_EXT.1, FPT_AEX_EXT.1, FPT_TUD_EXT.1
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. Addressed by: FMT_SMF.1, FPT_IDV_EXT.1, FPT_TUD_EXT.1.5, FPR_ANO_EXT.1
O.PROTECTED_COMMS	To address both passive (eavesdropping) and active (packet modification) network attack threats, conformant TOEs will use a trusted channel for sensitive data. Sensitive data includes cryptographic keys, passwords, and any other data specific to the application that should not be exposed outside of the application. Addressed by: FTP_DIT_EXT.1, FCS_TLSC_EXT.1(1), FCS_TLSC_EXT.1(2), FCS_DTLS_EXT.1, FCS_RBG_EXT.1
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 will 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. Addressed by: FDP_DAR_EXT.1, FDP_DAR_EXT.1, FCS_STO_EXT.1, FCS_RBG_EXT.1
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. Addressed by: FMT_MEC_EXT.1, FPT_API_EXT.1, FPT_LIB_EXT.1

4.2 Security Objectives for the Operational Environment

Table 8 describes the security objectives that the TOE's operating environment is required to meet as defined in the AS PP.

Table 8 – Security Objectives for the Operational Environment

Assumption	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_ADMIN	The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.
OE.PROPER_USER	The user of the application software is not willfully negligent or hostile, and uses the software within compliance of the applied enterprise security policy.

4.3 Security Objectives Rationale

Table 9 describes how the assumptions, threats, and organizational security policies map to the security objectives as defined in the AS PP.

Table 9 – Security Objectives Rationale Mapping

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

5. Extended Components

This section defines the extended SFRs and extended SARs met by the TOE.

5.1 Extended TOE Security Functional Components

Table 12 in section 7.2 below identifies the extended SFRs implemented by the TOE. These extended SFRs' definitions are not repeated in this ST, but they are taken directly from the AS PP.

5.2 Extended TOE Security Assurance Components

Table 10 identifies the extended SARs claimed for the TOE. The extended SARs' definitions are taken directly from the AS PP and are not repeated in this ST.

Table 10 – Extended TOE Security Assurance Components

Name	Description
ALC_TSU_EXT.1	Timely Security Updates

6. Security Assurance Requirements

The AS PP identifies the SARs to frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

This section lists the set of SARs that are required in evaluations against the AS PP. The AS PP is conformant to Parts 2 (extended) and 3 (extended) of CC V3.1, Revision 4.

The general model for evaluation of TOEs against STs written to conform to PPs is as follows: after the ST has been approved for evaluation, the ITSEF[20] will obtain the TOE, supporting environment (if required), and the guidance documentation for the TOE. The ITSEF is expected to perform actions mandated by the Common Evaluation Methodology (CEM) for the ASE and ALC SARs. The ITSEF also performs the Assurance Activities contained within the AS PP. The Assurance Activities that are captured in the AS PP also provide clarification as to what the developer needs to provide to demonstrate the TOE is compliant with the PP.

The TOE security assurance requirements are identified in Table 11.

Table 11 – Security Assurance Requirements

Assurance Requirements	
Security Target (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Stated security requirements (ASE_REQ.1)
	Security problem definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labeling of the TOE (ALC_CMC.1)
	TOE CM[21] coverage (ALC_CMS.1)
	Timely Security Updates (ALC_TSU_EXT.1)
Tests (ATE)	Independent testing – Conformance (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

[20] ITSEF – Information Technology Security Evaluation Facility
[21] CM – Configuration Management

7. Security Functional Requirements

The individual SFRs are specified in the sections below. SFRs in this section are mandatory SFRs that any conformant TOE must meet. Based on selections made in these SFRs, it will also be necessary to include some of the selection-based SFRs in Appendix B.

The Assurance Activities defined in AS PP describe actions that the evaluator will take in order to determine compliance of a particular TOE with the SFRs. The content of these Assurance Activities will therefore provide more insight into deliverables required from TOE Developers.

7.1 Conventions

The conventions used in descriptions of the SFRs are as follows:

- Refinement: Indicated with bold text (e.g., [**refinement**]).
- Selection: Indicated with underlined text surrounded by brackets (e.g., [selection]).
- Assignment: Indicated with italicized text surrounded by brackets (e.g., [*assignment*]).
- Assignment within a Selection: Indicated with italicized and underlined text surrounded by brackets (e.g., [*assignment within a selection*]).
- Refinement within a Selection: Indicated with bold and underlined text surrounded by brackets (e.g., [**assignment within a selection**]).
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/".
- Extended SFRs are identified by having a label 'EXT' at the end of the SFR name.

7.2 Security Functional Requirements

This section specifies the SFRs for the TOE and organizes the SFRs by CC class. Table 12 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement. Note that some column headers use the following abbreviations: S=Selection; A=Assignment; R=Refinement; I=Iteration.

Table 12 – TOE Security Functional Requirements

Name	Description	S	A	R	I
Required SFRs					
FCS_RBG_EXT.1	Random Bit Generation Services	✓			
FCS_STO_EXT.1	Storage of Credentials	✓	✓		
FDP_DAR_EXT.1	Encryption of Sensitive Application Data	✓			
FDP_DEC_EXT.1	Access to Platform Resources	✓	✓		
FDP_NET_EXT.1	Network Communications	✓	✓		
FMT_CFG_EXT.1	Secure by Default Configuration				
FMT_MEC_EXT.1	Supported Configuration Mechanism				

Name	Description	S	A	R	I
FMT_SMF.1	Specification of Management Functions	✓	✓		
FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Information	✓	✓		
FPT_AEX_EXT.1	Anti-Exploitation Capabilities	✓	✓		
FPT_API_EXT.1	Use of Supported Services and APIs				
FPT_LIB_EXT.1	User of Third Party Libraries		✓		
FPT_TUD_EXT.1	Integrity for Installation and Update	✓			
FTP_DIT_EXT.1	Protection of Data in Transit	✓			
Selection-based SFRs					
FCS_CKM.1(1)	Cryptographic Asymmetric Key Generation	✓		✓	✓
FCS_CKM.2	Cryptographic Key Establishment	✓		✓	
FCS_CKM_EXT.1	Cryptographic Key Generation Services	✓			
FCS_COP.1(1)	Cryptographic Operation – Encryption/Decryption	✓			✓
FCS_COP.1(2)	Cryptographic Operation – Hashing	✓			✓
FCS_COP.1(3)	Cryptographic Operation – Signing	✓		✓	✓
FCS_COP.1(4)	Cryptographic Operation – Keyed-Hash Message Authentication	✓	✓		✓
FCS_RBG_EXT.2	Random Bit Generation from Application	✓			
FCS_TLSC_EXT.1	TLS Client Protocol	✓			
FCS_TLSC_EXT.4	TLS Client Protocol	✓			
FIA_X509_EXT.1	X.509 Certificate Validation	✓			
FIA_X509_EXT.2	X.509 Certificate Authentication	✓			

7.2.1 Class FCS: Cryptographic Support

FCS_CKM.1(1) Cryptographic Asymmetric Key Generation

FCS_CKM.1.1(1)

The application shall [implement functionality] to generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm [

- [**ECC[22] schemes**] using [**"NIST[23] curves" P-256, P-384 and [no other curves]**]] that meet the following: [**FIPS PUB[24] 186-4, "Digital Signature Standard (DSS)", Appendix B.4**],

].

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: [

[22] ECC – Elliptic Curve Cryptography
[23] NIST – National Institute of Standards and Technology
[24] PUB – Publication

- [**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"**]

  ].

FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1

The application shall [implement asymmetric key generation].

FCS_COP.1(1) Cryptographic Operation – Encryption/Decryption

FCS_COP.1.1(1)

The application shall perform encryption/decryption in accordance with a specified cryptographic algorithm

- AES[25]-CBC[26] (as defined in NIST SP 800-38A) mode;
- and [AES-GCM[27] (as defined in NIST SP 800-38D)]

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

FCS_COP.1(2) Cryptographic Operation – Hashing

FCS_COP.1.1(2)

The application shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [SHA[28]-256, SHA-384] and message digest sizes [256, 384] bits that meet the following: FIPS Pub 180-4.

FCS_COP.1(3) Cryptographic Operation – Signing

FCS_COP.1.1(3)

The application shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

- [**RSA[29] schemes**] using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 4,
- [**ECDSA[30] schemes**] using "NIST curves" P-256, P-384 and [no other curves] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5

].

[25] AES – Advanced Encryption Standard
[26] CBC – Cipher Block Chaining
[27] GCM – Galois Counter Mode
[28] SHA – Secure Hash Algorithm
[29] RSA – Rivest, Shamir, Adleman
[30] ECDSA – Elliptic Curve Digital Signature Algorithm

FCS_COP.1(4) Cryptographic Operation – Keyed-Hash Message Authentication

FCS_COP.1.1(4)

The application shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm

- HMAC[31]-SHA-256
- and [SHA-384]

with key sizes [*256, 384*] and message digest sizes 256 and [384] bits that meet the following: FIPS Pub 198-1 *The Keyed-Hash Message Authentication Code* and FIPS Pub 180-4 *Secure Hash Standard*.

FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1

The application shall [implement DRBG[32] functionality] for its cryptographic operations.

FCS_RBG_EXT.2 Random Bit Generation from Application

FCS_RBG_EXT.2.1

The application shall perform all deterministic random bit generation (DRBG) services in accordance with NIST Special Publication 800-90A using [CTR_DRBG (AES)].

FCS_RBG_EXT.2.2

The deterministic RBG[33] shall be seeded by an entropy source that accumulates entropy from a platform-based DRBG and [a software-based 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.

FCS_STO_EXT.1 Storage of Credentials

FCS_STO_EXT.1.1

The application shall [implement functionality to securely store [*passwords to credentials for the pvwaappuser, pvwagwuser, passwordmanager, psmgw <machine name>, and psmapp <machine name> accounts*]] to non-volatile memory.

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1

The application shall [implement TLS 1.2 (RFC 5246)] supporting the following cipher suites: [

- TLS_ECDHE[34]_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289

[31] HMAC – Hash-based Message Authentication Code
[32] DRBG – Deterministic Random Bit Generator
[33] RBG – Random Bit Generation
[34] ECDHE – Elliptic Curve Diffie Hellman Ephemeral

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

].

FCS_TLSC_EXT.1.2

The application shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3

The application shall establish a trusted channel only if the peer certificate is valid.

FCS_TLSC_EXT.4 TLS Client Protocol

FCS_TLSC_EXT.4.1

The application shall present the supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [secp256r1, secp384r1].

7.2.2 Class FDP: User Data Protection

FDP_DAR_EXT.1 Encryption of Sensitive Application Data

FDP_DAR_EXT.1.1

The application shall [protect sensitive data in accordance with FCS_STO_EXT.1] in non-volatile memory.

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 [[*IIS logs and event logs*]].

FDP_NET.EXT.1 Network Communications

FDP_NET_EXT.1.1

The application shall restrict network communication to [user-initiated communication for [*RDP connections to PSM and HTTPS over TLS connections to PVWA*], [*application-initiated RDP over TLS connections to targets and TLS connections to the EPV server, HTTP connections to the CA server for certification revocation checks*]].

7.2.3 Class FIA: Identification and Authentication

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1

The application shall [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[35] 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.
- The application shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280]
- 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 [36] 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 extendedKeyUsage 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 extendedKeyUsage field.
  - S/MIME[37] 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 extendedKeyUsage 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 extendedKeyUsage field.
  - Server certificates presented for EST[38] shall have the CMC[39] Registration Authority (RA) purpose (id-kpcmcRA with OID 1.3.6.1.5.5.7.3.28) in the extendedKeyUsage 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].

[35] CA – Certificate Authority
[36] OID – Object Identifier
[37] MIME – Multipurpose Internet Mail Extensions
[38] EST – Enrollment over Secure Transport
[39] CMC – Certificate Management over Cryptographic Message Syntax

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].

7.2.4 Class FMT: Security Management

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's binaries and data files from modification by normal unprivileged user.

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_SMF.1 Specification of Management Functions

FMT_SMF.1.1

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

- [*Manage the target account platforms*
- *Manage the service account platforms*
- *Manage the discovery processes*
- *Manage the password change processes*]

].

7.2.5 Class FPR: Privacy

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].

7.2.6 Class FPT: Protection of the TSF

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 [*the OpenSSL hash check*].

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 compiled with stack-based buffer overflow protection enabled.

FPT_API_EXT.1 Use of Supported Services and APIs

FPT_API_EXT.1.1

The application shall use only documented platform APIs.

FPT_LIB_EXT.1 User of Third Party Libraries

FPT_LIB_EXT.1.1

The application shall be packaged with only [*the third-party libraries listed in Table 13*]

Table 13 – Third-Party Libraries

Components	Libraries	Version	Vender
PSM	MFC71.DLL	7.10.6041.0	Microsoft Corporation
	msvcp120.dll	12.00.21005.1	Microsoft Corporation
	msvcr71.dll	7.10.3052.4	Microsoft Corporation
	msvcr90.dll	9.00.21022.8	Microsoft Corporation
	msvcr120.dll	12.00.21005.1	Microsoft Corporation
	WebDriver.dll	3.8.0	Selenium Committers
	WebDriver.Support.dll	3.8.0	Selenium Committers
CPM	AWSSDK.dll	2.3.34.0	Amazon.com, Inc
	Calstle.Core.dll	3.2.0.2259	Castle Project
	Calstle.Core.dll	3.3.3.58	Castle Project
	Castle.Windsor.dll	3.2.0.2406	Castle Project
	Castle.Windsor.dll	3.3.0.51	Castle Project
	Common.Logging.Core.dll	3.0.0.0	Netcommon
	Common.Logging.dll	3.0.0.0	Netcommon
	DocumentFormat.OpenXml.dll	2.0.5022.0	Microsoft Corporation
	DummyDNA.dll	10.4.10.7	N/A
	FluentNHibernate.dll	0.0.0.0	N/A
	ICSharpCode.TextEditor.dll	3.2.1.6466	ic#code

Components	Libraries	Version	Vender
	iimConnector.dll	10.2.0.120	Ipswitch, Inc
	iimConnector64.dll	10.2.0.120	Ipswitch, Inc
	iimds.dll	10.0.1.13	Ipswitch, Inc
	iimFirefoxConnector.dll	2.0.1.12	Ipswitch, Inc
	iimInterface.dll	10.2.0.120	Ipswitch, Inc
	iimInterface64.dll	10.2.0.120	Ipswitch, Inc
	iimir.dll	6.3.0.1	Ipswitch, Inc
	iimIR1.dll	N/A	Ipswitch, Inc
	iimIR2.dll	N/A	Ipswitch, Inc
	iimIR3.dll	N/A	Ipswitch, Inc
	iimIRm.dll	2.0.1.33	Ipswitch, Inc
	iMacros.Core.dll	10.4.28.1074	Ipswitch, Inc
	iMacros.IO.Csv.dll	10.4.28.1074	Ipswitch, Inc
	iMacros.TabbedBrowser.dll	10.4.28.1074	Ipswitch, Inc
	iMacros.WinUI.ActionList.dll	10.4.28.1074	Ipswitch, Inc
	iMacros.WinUI.Common.dll	10.4.28.1074	Ipswitch, Inc
	iMacrosBHO.dll	10.4.28.1074	Ipswitch, Inc
	iMacrosScreenshot.dll	10.4.28.1074	Ipswitch, Inc
	imsys.dll	10.0.1.0	Ipswitch, Inc
	imtcp.dll	2.1.0.34	Ipswitch, Inc
	imtcp64.dll	2.1.0.34	Ipswitch, Inc
	Interop.SHDocVw.dll	1.1.0.0	N/A
	Interop.TaskScheduler.dll	1.1.0.0	N/A
	msvcr71.dll	7.10.6030.0	Microsoft Corporation
	msvcr90.dll	9.0.21022.8	Microsoft Corporation
	Newtonsoft.Json.dll	6.0.2.16931	Newtonsoft
	NHibernate.dll	4.1.1.4000	NHibernate.info
	PowerCollections.dll	N/A	N/A
	Quartz.dll	2.3.3.0	Quartz Scheduler
	Renci.SshNet.dll	2016.1.0.0	Renci
	rsaws.dll	8.1.1.0	EMC Corporation
	SystemWrapper.dll	0.5.2.0	N/A
	wcipsl.dll	1.0.0.1	Ipswitch, Inc
	xerces-c_2_5_0.dll	2.5.0.0	Apache Software Foundation
PVWA	msvcr71.dll	7.10.6030.0	Microsoft Corporation

Components	Libraries	Version	Vender
	msvcr90.dll	9.0.30729.1	Microsoft Corporation
	PowerCollections.dll	N/A	N/A
	Swashbuckle.Core.dll	1.0.0.0	N/A

FPT_TUD_EXT.1 Integrity for Installation and Update

FPT_TUD_EXT.1.1

The application shall [provide the ability] to check for updates and patches to the application software.

FPT_TUD_EXT.1.2

The application shall be distributed using the format of the platform-supported package manager.

FPT_TUD_EXT.1.3

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.1.4

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

FPT_TUD_EXT.1.5

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

FPT_TUD_EXT.1.6

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

7.2.7 Class FTP: Trusted Path/Channel

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1

The application shall [

- encrypt all transmitted sensitive data with [TLS],
- invoke platform-provided functionality to encrypt all transmitted sensitive data with [HTTPS, TLS]

] between itself and another trusted IT[40] product.

[40] IT – Information Technology

8. TOE Summary Specification

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

8.1 TOE Security Functionality

Each of the security requirements and the associated descriptions correspond to the security functions. Hence, each function is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions satisfy the necessary requirements.

Table 14 – Mapping of TOE Security Functionality to Security Functional Requirements

TOE Security Function	SFR ID[41]	Description
Cryptographic Support	FCS_CKM.1(1)	Cryptographic Asymmetric Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM_EXT.1	Cryptographic Key Generation Services
	FCS_COP.1(1)	Cryptographic Operation – Encryption/Decryption
	FCS_COP.1(2)	Cryptographic Operation – Hashing
	FCS_COP.1(3)	Cryptographic Operation – Signing
	FCS_COP.1(4)	Cryptographic Operation – Keyed-Hash Message
	FCS_RBG_EXT.1	Random Bit Generation Services
	FCS_RBG_EXT.2	Random Bit Generation from Application
	FCS_STO_EXT.1	Storage of Credentials
	FCS_TLSC_EXT.1	TLS Client Protocol
	FCS_TLSC_EXT.4	TLS Client Protocol
User Data Protection	FDP_DAR_EXT.1	Encryption of Sensitive Application Data
	FDP_DEC_EXT.1	Access to Platform Resources
	FDP_NET_EXT.1	Network Communications
Identification and Authentication	FIA_X509_EXT.1	Certificate Validation
	FIA_X509_EXT.2	Certificate Authentication
Security Management	FMT_CFG_EXT.1	Secure by Default Configuration
	FMT_MEC_EXT.1	Supported Configuration Mechanism
	FMT_SMF.1	Specification of Management Functions
Privacy	FPR_ANO_EXT.1	User Consent for Transmission of Personally Identifiable Information
Protection of the TSF	FPT_AEX_EXT.1	Anti-Exploitation Capabilities
	FPT_API_EXT.1	Use of Supported Services and APIs

[41] ID – Identification

TOE Security Function	SFR ID[41]	Description
	FPT_LIB_EXT.1	User of Third Party Libraries
	FPT_TUD_EXT.1	Integrity for Installation and Update
Trusted Path / Channels	FTP_DIT_EXT.1	Protection of Data in Transit

8.1.1 Cryptographic Support

The TOE implements the OpenSSL FIPS Object Module v2.0.14 with the CyberArk libraries to provide the required algorithms for the cryptographic operations used to establish trusted channels with PSM, CPM, and PVWA and for all cryptographic operations within PSM, CPM, and PVWA. Table 15 lists required information about the TOE's usage of cryptography.

Table 15 – Cryptographic Algorithm and Key Sizes for PSM, CPM, and PVWA

Cryptographic Operation	Algorithm	Key Sizes / Curves	Usage	Certificate
Encryption/Decryption	AES – CBC and GCM	128, 256	TLS (TOE)	CAVP 5487 and C1088
	AES – CBC	256	Inside credential files	
Signature Generation Signature Verification	RSA	2048, 3072	TLS (TOE)	CAVP 2948 and C1088
Key Pair Generation Public Key Verification Signature Generation Signature Verification	ECDSA – P256 and P384	256, 384	TLS (TOE)	CAVP 1473 and C1088
Key Exchange/Establishment	ECDHE	256, 384	TLS (TOE)	CAVP 1945 and C1088
Message Digest / Hashing	SHA-256, SHA-384	256, 384	TLS (TOE)	CAVP 4404 and C1088
Message Authentication	HMAC SHA-256, SHA-384	256, 384	TLS (TOE)	CAVP 3645 and C1088
Random Number Generation	CTR DRBG (with AES)	N/A[42]	TOE DRBG	CAVP 2162 and C1088

FCS_CKM.1(1)/FCS_CKM_EXT.1/FCS_CKM.2

Table 15 above lists all the key sizes used for RSA and ECC asymmetric key generation schemes and the usage of each scheme. Table 15 also lists the key establishment and key exchange schemes used by the TOE. Note that the TOE only implements functionality for ECC key generation and establishment. The TOE uses key generation and establishment with the TLS. The use of asymmetric encryption is needed for the TLS protocol used by the TOE. Key generation follows the requirements within FIPS PUB 186-4. Key establishment follows the requirements within NIST Special Publication 800-56A and NIST Special Publication 800-56B.

FCS_COP.1(1)

Table 15 above 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 and protecting passwords in credential files. The TOE uses AES-CBC and AES-GCM in its TLS connections and only AES-CBC when protecting passwords in a credential file.

[42] N/A – Not Applicable

The cryptographic algorithm follows NIST SP 800-38A (CBC) and NIST SP 800-38D (GCM). The cryptographic key sizes are 128-bit and 256-bit for all modes.

FCS_COP.1(2)
Table 15 above lists all the key sizes used for SHA hashing and message digests within the TOE. Usages of SHA is limited to TLS connections. The TOE's implementations of SHA follow the requirements within FIPS Pub 180-4.

FCS_COP.1(3)
Table 15 above lists all the key sizes used for signature generation and verification within the TOE. Signature generation is used in TLS connections. Signature verification is used in TLS connections and to verify the digital signature on TOE files. The TOE's implementations of signature generation and verification follow the requirements within FIPS PUB 186-4.

FCS_COP.1(4)
Table 15 above lists all the key sizes used for HMAC message authentication within the TOE. Usages of HMAC is limited to TLS connections. The TOE's implementations of HMAC follows the requirements within FIPS Pub 180-4.

FCS_RBG_EXT.1/FCS_RBG_EXT.2
The TOE's CTR_DRBG functionality is implemented within its statically linked CyberArk PAS Cryptographic Library for Windows. This implementation conforms to the NIST Special Publication 800-90A requirements. The TOE's implementation of CTR_DRBG is AES-256. When the TOE starts up, the DRBG is seeded with 256 bits of entropy from the Windows Entropy Pool by calling the OpenSSL RAND_seed function for the CryptGenRandom function and for Crypto API. The platform system time and tick count noise sources are added to the Windows Entropy Pool after initialization. On an ongoing basis, the TOE seeds the DRBG with 256 bits of entropy by calling the RAND_seed function for the BCryptGenRandom function and for the Crypto Next Generation API. More information about the entropy process is described in the proprietary Entropy Rationale document.

FCS_STO_EXT.1
The TOE securely stores passwords in non-volatile memory for the following accounts:

- PVWAappuser – This is the EPV account that will run the PVWA application. It is located in the "C:\CyberArk\CredFiles" folder in the appuser.ini file.
- PVWAgwuser – This is the EPV account that will run the PVWA gateway. It is located in the "C:\CyberArk\CredFiles" folder in the gwuser.ini file.
- PasswordManager – This is the EPV account that will run the CPM application. It is located in the "C:\Program Files (x86)\CyberArk\Password Manager\Vault" folder in the user.ini file.
- PSMApp_<machine_name> – This is the EPV account that will run the PSM application. It is located in the "C:\Program Files (x86)\CyberArk\PSM\Vault" folder in the psmapp.cred file.
- PSMGW_<machine_name> – This is the EPV account that will run the PSM gateway. It is located in the "C:\Program Files (x86)\CyberArk\PSM\Vault" folder in the psmgw.cred file.

The password for each of the accounts above is encrypted using AES256-CBC and saved in the noted file.

FCS_TLSC_EXT.1
The TOE implements a TLS v1.2 client according to RFC 5246 using its CyberArk PAS TLS Library for Windows. This functionality is only used to communicate to the EPV server over TLS. Only the following cipher suites are allowed by the TOE for communications to the EPV server:

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

As part of establishing a TLS connection as a client, the TOE will verify 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 EPV server's certificate may contain its IP address because the EPV server is on a hardened machine that is not necessarily accessible via DNS[43]. 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. The TLS library underwent component validation testing and was assigned the CAVP certificate 1947 and C1092.

FCS_TLSC_EXT.4
The TOE implements TLS v1.2 with support for EC[44] 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 EPV server and configured during hardening of the server.

TOE Security Functional Requirements Satisfied: FCS_CKM.1(1), FCS_CKM.2, FCS_CKM_EXT.1, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_RBG_EXT.1, FCS_RBG_EXT.2, FCS_STO_EXT.1, FCS_TLSC_EXT.1, FCS_TLSC_EXT.4

8.1.2 User Data Protection

FDP_DAR_EXT.1
Sensitive data within the TOE is limited to the passwords for service accounts. The TOE limits the storing of sensitive data in non-volatile memory to only passwords for the pvwaappuser, pvwagwuser, passwordmanager, psmgw_<machine_name>, and psmapp_<machine_name> service accounts. Each account's password is encrypted using AES256-CBC before it is saved. Other passwords that are used to authenticate with the EPV server are never stored in non-volatile memory by the TOE and transferred to and from the TOE over secure connections.

FDP_DEC_EXT.1 and FDP_NET_EXT.1
The TOE will limit its access to only network connectivity when accessing the platform's hardware resources. The TOE requires network access when workstations connect to the server over RDP with TLS and HTTPS or when the TOE server connects to the EPV server and remote targets over TLS. The user will initiate a connection from their RDP Client over port 3389 when they want to connect to the PSM Client TSFI to be routed to a target machine. The user or administrator will initiate a connection from their browser to IIS in the OE over port 443 using an HTTPS connection when they want to connect to the PVWA Client TSFI or PVWA RESTful API. This requires the

[43] DNS – Domain Name System
[44] EC – Elliptic Curve

TOE to use the IIS service to use network resources. The TOE will also use port 80 for HTTP connections to the CA server for certification revocation checks from each component.

When a user connects to the TOE for an RDP session, the TOE will connect to the EPV server to verify their authentication information over TLS on port 443. If the authentication passes, PSM then connects to the target device using the RDP Client over a TLS connection on port 3389. PVWA connects to the EPV server over TLS on port 443 when it verifies authentication information through the PVWA Client TSFI or PVWA RESTful API. PVWA also connects to the EPV server over TLS on port 443 when it needs to update configuration settings for the PAS components. CPM connects to the EPV server over TLS on port 443 when it queries for updated password policies or changes an account's password.

The TOE will limit its access to sensitive IIS and event logs that are stored on and maintained by the platform.

TOE Security Functional Requirements Satisfied: FDP_DAR_EXT.1, FDP_DEC_EXT.1, FDP_NET_EXT.1

8.1.3 Identification and Authentication

FIA_X509_EXT.1
The TOE provides its own implementation of TLS to perform certificate validation. The TOE's PSM, CPM, and PVWA components are clients to the EPV server and each validates the EPV server's X.509v3 certificate during TLS authentication. The components ensure 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 components treat 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. Each of the components validates the revocation status of the EPV'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 PSM, CPM, and PVWA components each check the EPV certificate against the downloaded CRL and automatically reject 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 PSM, CPM, and PVWA components each validate that the EPV'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. Each components of the TOE will read the location of their certificates from their vault.ini files using the ClientCertificate parameter.

The TOE validates X.509v3 certificates from the EPV 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 will automatically reject a certificate if it is found to be invalid according to the requirements in FIA_X509_EXT.1. A certificate with an unknown revocation status due to the inability to establish a connection to the CDP will be rejected. The connection from the TOE to the CDP is conducted over HTTP as per the RFC.

TOE Security Functional Requirements Satisfied: FIA_X509_EXT.1, FIA_X509_EXT.2

8.1.4 Security Management

FMT_CFG_EXT.1
No default credentials are provided by the TOE after the initial installation. Once the TOE is installed, it is connected to the EPV server for all authentication needs. The files created during installation are set with default permissions that do not allow the "Users" group to modify them. The files within the PSM's Oracle folder will be automatically configured to allow the "Users" group read and execute permissions. All other PSM files will not allow the "Users" group any access. The files within the CPM and PVWA folders will be automatically configured to allow the "Users" group read and execute permissions where needed.

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 EPV server and can be configured using the PVWA 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 PSM component is stored in "C:\Program Files (x86)\CyberArk\PSM\basic_psm.ini". Local configuration information related to the PVWA component is stored in the "C:\inetpub\wwwroot\PasswordVault\web.config" file. CPM does not provide a local configuration file

FMT_SMF.1
The TOE also provides the PVWA Client TSFI and PVWA RESTful API for managing the TOE and PAS components. Through these interfaces, administrators can perform the following:

- Manage the target/service account platforms – Target account platforms and service account platforms define the technical settings which determine how the system manages accounts on different platforms.
- Manage the discovery processes – Discovery processes scan predefined machines for new and modified accounts and their dependencies, and then display the discovered accounts so administrators can see which accounts should be onboarded into EPV.
- Manage the password change processes – CPM changes passwords according to the Password Change parameters that can be managed by an administrator.

TOE Security Functional Requirements Satisfied: FMT_CFG_EXT.1, FMT_MEC_EXT.1, FMT_SMF.1

8.1.5 Privacy

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.

TOE Security Functional Requirements Satisfied: FPR_ANO_EXT.1

8.1.6 Protection of the TSF

FPT_AEX_EXT.1

The TOE does not request memory mappings at explicit addresses except for checking the hash value of the OpenSSL library. During the self-check in FIPS mode, the calibeay64102k.dll and cassleay64102k.dll files are written to their respective memory addresses of 0x00000000FB000000 and 0x0000001800000000 to allow the module to compute the correct hash of the libraries for comparison to known values. This mapped memory area has only read and execute permissions but no write permissions. When the TOE is being complied, it uses the "RandomizedBaseAddress" 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 compiled using the "/NXCOMPAT" flag to enable Data Execution Protection (DEP) and the "/GS" flag to enable stack-based buffer overflow protection.

The TOE is installed on a hardened operating system based on Microsoft Bastion Host server recommendations. The TOE hardening is part of the installation and results in disablement of many operating system services. The hardening process also strips the permissions from existing and built-in Windows accounts (except the account that runs the installation). For more information about the hardening process, please refer to the "Harden the PSM server machine" section of the *CyberArk Installation Guide* and the *CyberArk Hardening the CyberArk CPM and PVWA Servers* document.

The TOE does not write user-modifiable files to directories that contain executable files. User-modifiable files are written to the "C:\CyberArk\Password Vault Web Access\Services\Logs\", "C:\CyberArk\Password Vault Web Access\Env\Log\", "C:\Program Files (x86)\CyberArk\PasswordManager\Logs", "C:\Program Files (x86)\CyberArk\PasswordManager\Scanner\Log", and "C:\Program Files (x86)\CyberArk\PSM\Logs" folders.

FPT_API_EXT.1

The TOE only uses supported platform APIs in order to function. The below list includes all the platform APIs used by the PSM component.

- advapi32.dll
- avifil32.dll
- comctl32.dll
- comdlg32.dll
- credui.dll
- crypt32.dll
- dnsapi.dll
- gdi32.dll
- imm32.dll
- iphlpapi.dll
- kernel32.dll
- mfc120.dll
- mpr.dll
- mscoree.dll
- msvcp120.dll
- msvcr120.dll
- msvcr90.dll
- msvfw32.dll
- netapi32.dll
- ole32.dll
- oleaut32.dll
- psapi.dll
- rpcrt4.dll
- secur32.dll
- shell32.dll
- shlwapi.dll
- user32.dll
- userenv.dll
- version.dll
- winmm.dll
- ws2_32.dll
- wtsapi32.dll

The below list includes all the platform APIs used by the CPM component.

- advapi32.lib
- comdlg32.lib
- gdi32.lib
- kernel32.lib
- odbc32.lib
- odbccp32.lib
- ole32.lib
- oleaut32.lib
- Psapi.lib
- shell32.lib
- shlwapi.lib
- user32.lib
- Userenv.lib
- uuid.lib
- version.lib
- winspool.lib
- ws2_32.lib
- Wtsapi32.lib

The below list includes all the platform APIs used by the PVWA component.

- Analyzers.dll
- BotDetect.dll
- Ionic.Zip.dll
- Log4Net.dll
- Microsot.CSharp.dll
- Newtonsoft.Json.dll
- PowerCollections.dll
- Quartz.dll
- Swashbuckle.core.dll
- Syste,Web.DynamicData.dll
- System.Configuration.dll
- System.Core.dll
- System.Data.dll
- System.Data.Linq.dll
- System.DirectoryServices.dll
- System.dll
- System.Drawing.dll
- System.Web.dll
- System.XML.dll
- VimService.dll
- WebChart.dll
- XmlDiffPatch.dll
- DocumentFormat.OpenXML.dll
- Microsoft.Web.UI.WebControls.dll
- System.ComponentModel.DataAnnotations.dll
- System.Data.DataSetExtensions.dll
- System.EnterpriseServices.dll
- System.Management.dll
- System.Net.Http.Formating.dll
- System.Runtime.Serializations.dll
- System.ServiceModel.Web.dll
- System.Web.Abstractions.dll
- System.Web.ApplicationServices.dll
- System.Web.Cors.dll
- System.Web.DynamicData.dll
- System.Web.Entity.dll
- System.Web.Extensions.Design.dll
- System.Web.Extensions.dll
- System.web.Http.Cors.dll
- System.Web.Http.dll
- System.Web.Http.WebHost.dll
- System.Windows.Forms.dll
- System.Xml.Linq.dll
- VimService.XmlSerializers.dll

FPT_LIB_EXT.1
The TOE is packaged with the third-party libraries listed in Table 13 above and requires these libraries in order to properly function.

FPT_TUD_EXT.1
The CyberArk Version Check tool is downloaded to the platform as part of the TOE and is used for checking updates to the TOE components. It relies on a file uploaded to the Vault server that contains all the current version information for the CyberArk PAS suite. The TOE administrator will need to upload this file once per version of PAS and can be used for all components of PAS. For local storage purposes, the TOE administrator will also need to upload the update packages to the vault to allow for an internal update repository. An email notification from CyberArk will be sent to the TOE administrator when a new version is available. The TOE administration that receives the email notification is responsible for uploading the files to the Vault server. The information in the email will contain the links to the appropriate download locations and the release notes related to the update. Since multiple components of the PAS solution check for updates against this central location, the administrator that uploads the files to the Vault server is responsible for maintaining accuracy of all component versions.

The TOE administration must run the Version Check tool whenever they need to check for a new version. This can be done periodically or when notified.

If an update is available for the TOE, the TOE administrator will download the latest version of the TOE software from the Safe on the Vault server. The package will contain the required executable (.exe) files for the TOE's platform. The current version of TOE software is returned after running the script. To determine the currently installed version without running the above script, the administrator can check for the TOE software in the

Programs and Features manager. The TOE will not automatically download or apply new packages that would replace or update its code.

The installation packages are digitally signed to protect them from alteration after publication. To verify the digital signature of a TOE package, the administrator must complete the following:

1. Download the TOE installation packages from CyberArk.
2. Download and install the Java Development Kit (JDK) from Oracle.
3. Download and install the JCE Unlimited Strength Jurisdiction Policy Files.
4. Run the following command without quotes, "%JDK_Home%\jarsigner.exe -verify -verbose -certs <filename>.zip". More information about the jarsigner's options can be found at https://docs.oracle.com/javase/7/docs/technotes/tools/windows/jarsigner.html#CCHFIDAB.

Individual TOE files are signed using the Windows OS package manager MS[45] Sign tool. To verify the integrity of the TOE installation file, complete the following:

1. Extract the files from the archive file.
2. Navigate to the *setup.exe* file.
3. Right-click the file, then Click on **Properties > Digitals Signatures**.
4. The "**CyberArk Software Ltd.**" signer should be selected. Click on **Details**, and then verify the signature details.

The TOE relies on the platform's package manager to make changes to the binary code. Installation of the updates is performed by an administrator while using the executable file (.exe) extracted from the archive file (.zip). The TOE software can be removed from the platform using the platform's Programs and Features manager. Uninstallation of the TOE will remove all traces of the application except for configuration settings, output files, and audit/log events.

TOE Security Functional Requirements Satisfied: FPT_AEX_EXT.1, FPT_API_EXT.1, FPT_IDV_EXT.1, FPT_LIB_EXT.1, FPT_TUD_EXT.1

8.1.7 Trusted Path/Channels

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 TLS channel between itself and the EPV server. PSM, CPM, and PVWA will each be able to negotiate a connection to the EPV server over TLS v1.2 when access safes that are stored in EPV.

The OE provides a trusted path for communications using IIS when a user or administrator connects to the PVWA Client TSFI or PVWA RESTful API over HTTPS from their web browser or REST client respectively. This HTTPS connection is secured using TLS v1.2 and encrypted using AES-128-GCM, AES-256-GCM, AES-128-CBC, or AES-256-CBC depending on the cipher suite negotiated with the TLS client.

[45] MS – Microsoft

The OE also provides a trusted channel between the TOE and remote targets over TLS v1.2. The RDP Client in the OE is used to create this TLS connection, which encrypts its traffic using AES-128-GCM, AES-256-GCM, AES-128-CBC, or AES-256-CBC depending on the cipher suite negotiated with the remote target.

TOE Security Functional Requirements Satisfied: FTP_DIT_EXT.1

8.1.8 Timely Security Updates

Upon discovery of a security vulnerability in any of CyberArk's products, underlying systems, or embedded 3rd-party libraries, a vulnerability assessment process commences and may vary depending on the vulnerability characteristics.

CyberArk reviews all OS updates to determine if they are applicable to the TOE. Because the TOE platform is hardened, CyberArk reviews all OS updates to determine if they are applicable to the TOE and notifies customers with update instructions as needed. Likewise, CyberArk has no control over third-party patches or updates but will incorporate any necessary third-party updates into a TOE update and notify the customer.

Typical activities resulting from the vulnerability assessment may include (depending on their severity):

- Release of a software patch that addresses the vulnerability.
- Issue a Security Bulletin or other notice to affected customers that discloses the vulnerability and mitigation information.
- Applying necessary security enhancement to the product roadmap.

The following table outlines the steps of the vulnerability assessment process. Some of these steps may take place in parallel:

1. Severity Review: Assessing the vulnerability's severity ranking.
    a. For 3rd-party libraries, review publicly available security rankings and analyses.
2. Mitigation Analysis: Evaluate whether there is a mitigation option (even temporary) that could reduce the severity of the vulnerability until it is permanently fixed.
3. Fix Assessment: Provide time and effort estimation for suggested fix.
4. Vulnerability Addressed: Addressing the vulnerability according to its Service Level Agreement (SLA).

CyberArk addresses the identified vulnerabilities within the following SLA in correlation with the severity and business risk rating:

- Critical
    - Response Time: Immediate (from time of analysis completion). Dependent on fix complexity (may take up to 90 days)
    - Covered Versions: All effected versions within their End of Development period
- High
    - Response Time: Next planned release cadence
    - Covered Versions: Latest version
- Medium / Low
    - Response Time: Added to roadmap and addressed within one of the next releases

- Covered Versions: Latest version

Security issues can be reported to this CyberArk website: https://www.cyberark.com/product-security/. Anyone reporting a security issue will be given a set of keys for encrypting the data for transfer. Current security bulletins may also be viewed from the same URL.

Customers will receive emails related to available updates that contain the link to download the latest software for the TOE. Updates may also be downloaded from the CyberArk Support Vault website: https://support.cyberark.com/.

9. Rationale

9.1 Conformance Claims Rationale

This Security Target extends Part 2 and extends to Part 3 of the *Common Criteria for Information Technology Security Evaluations,* Version 3.1, Revision 4, September 2012. This ST conforms to the AS PP.

9.1.1 Variance Between the PP and this ST

There is no variance between the AS PP and this ST.

9.1.2 Security Assurance Requirements Rationale

The assumptions, threats, OSPs, and objectives defined in this ST are those specified in the AS PP. This ST maintains exact conformance to the AS PP, including the assurance requirements listed in section 5 of the AS PP. The TOE is a standalone application that runs on a Windows platform and is applicable to the AS PP.

10. Acronyms and Terms

This section describes the acronyms and terms used throughout the document.

10.1 Acronyms

Table 16 defines the acronyms used throughout this document.

Table 16 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
AS PP	Protection Profile for Application Software v1.2; April 22, 2016
ASLR	Address Space Layout Randomization
CA	Certificate Authority
CBC	Cipher Block Chaining
CC	Common Criteria
CDP	Certificate Revocation List Distribution Point
CEM	Common Evaluation Methodology
CM	Configuration Management
CMC	Certificate Management over Cryptographic Message Syntax
CPM	Centralized Platform Management
CRL	Certificate Revocation List
CTR	Counter Mode
DEP	Data Execution Protection
DHE	Diffie Hellman Ephemeral
DN	Distinguished Name
DNS	Domain Name System
DRBG	Deterministic Random Bit Generator
DSS	Digital Signature Standard
DVR	Digital Video Recorder/Recording
EAL	Evaluation Assurance Level
EC	Elliptic Curve
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
ECDHE	Elliptic Curve Diffie Hellman Ephemeral

Acronym	Definition
ECDSA	Elliptic Curve Digital Signature Algorithm
EPV	Enterprise Password Vault
EST	Enrollment over Secure Transport
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standard
GCM	Galois Counter Mode
HMAC	Hash-based Message Authentication Code
HTTP	Hypertext Transport Protocol
HTTPS	Hypertext Transport Protocol Secure
ICU	International Components for Unicode
ID	Identification
IE	Internet Explorer
IIS	Internet Information Service
IP	Internet Protocol
IT	Information Technology
ITSEF	Information Technology Security Evaluation Facility
JDK	Java Development Kit
KB	Kilobyte
MAC	Message Authentication Code
MIME	Multipurpose Internet Mail Extensions
N/A	Not Applicable
NIST	National Institute of Standards and Technology
OCSP	Online Certificate Status Protocol
OE	Operating Environment
OID	Object Identifier
OPM	On-Demand Privileges Manager
OS	Operating System
OSP	Organizational Security Policy
PAS	Privileged Access Security
PCRE	Perl Compatible Regular Expressions
PII	Personally Identifiable Information
PP	Protection Profile
PSM	Privileged Session Manager
PSMP	Privileged Session Manager SSH (Secure Shell) Proxy

Acronym	Definition
PUB	Publication
PVWA	Password Vault Web Access
R2	Release Two
RA	Registration Authority
RBG	Random Bit Generation
RDP	Remote Desktop Protocol
RDS	Remote Desktop Services
REST	Representational State Transfer
RFC	Request for Comments
RSA	Rivest, Shamir, Adleman
SAN	Subject Alternative Name
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SP	Service Pack
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TD	Technical Decisions
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TOE Security Function Interface
URL	Uniform Resource Locator

10.2 Terms

Table 17 defines the terms used throughout this document.

Table 17 – Terms

Name	Definition
Administrator/User	Human or IT entity interacting with the TOE from outside of the TOE boundary.
Assurance Activities	Actions that the evaluator will take to determine compliance of a particular TOE with the SFRs.
Common Criteria	Common Criteria for Information Technology Security Evaluation.
Common Evaluation Methodology	Common Evaluation Methodology for Information Technology Security Evaluation.

Name	Definition
Protection Profile	An implementation-independent set of security requirements for a category of products.
Security Target	A set of implementation-dependent security requirements for a specific product.
Target of Evaluation	The product under evaluation. In this case, application software and its supporting documentation.
TOE Security Functionality	The security functionality of the product under evaluation.
TOE Summary Specification	A description of how a TOE satisfies the SFRs in a ST.
Security Functional Requirement	A requirement for security enforcement by the TOE.
Security Assurance Requirement	A requirement to assure the security of the TOE.

Prepared by:
Corsec Security, Inc.

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

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