

PAMLicenseCheck Implementation Guide

Ver 1.0

14 Nov 2023

Timothy A. Sawyer, Global Support Operations, CyberArk Ltd.

Table of Contents

INTRODUCTION	3
ABOUT PAMLICENSECHECK.....	3
DEVELOPMENT AND VERSION COMPATIBILITY	3
<i>Development</i>	3
<i>Version Compatibility</i>	3
PRE-REQUISITES AND INSTALLATION.....	3
PRE-REQUISITES	3
<i>Installation</i>	5
PROPERTIES FILE	5
RUNNING PAMLICENSECHECK	7
LOGS AND TROUBLESHOOTING	7
<i>plc_Main.py.log</i>	7
KNOWN ISSUES.....	8

Introduction

About PAMLicenseCheck

Using a combination of the vault license.xml as well as getting vault users, this will generate a report of license usage in the vault.

Development and Version Compatibility

Development

BulkCPMManage was developed in Python 3.12. This has been tested on Mac (Ventura 13.5), Windows 11 and Windows Server 2019.

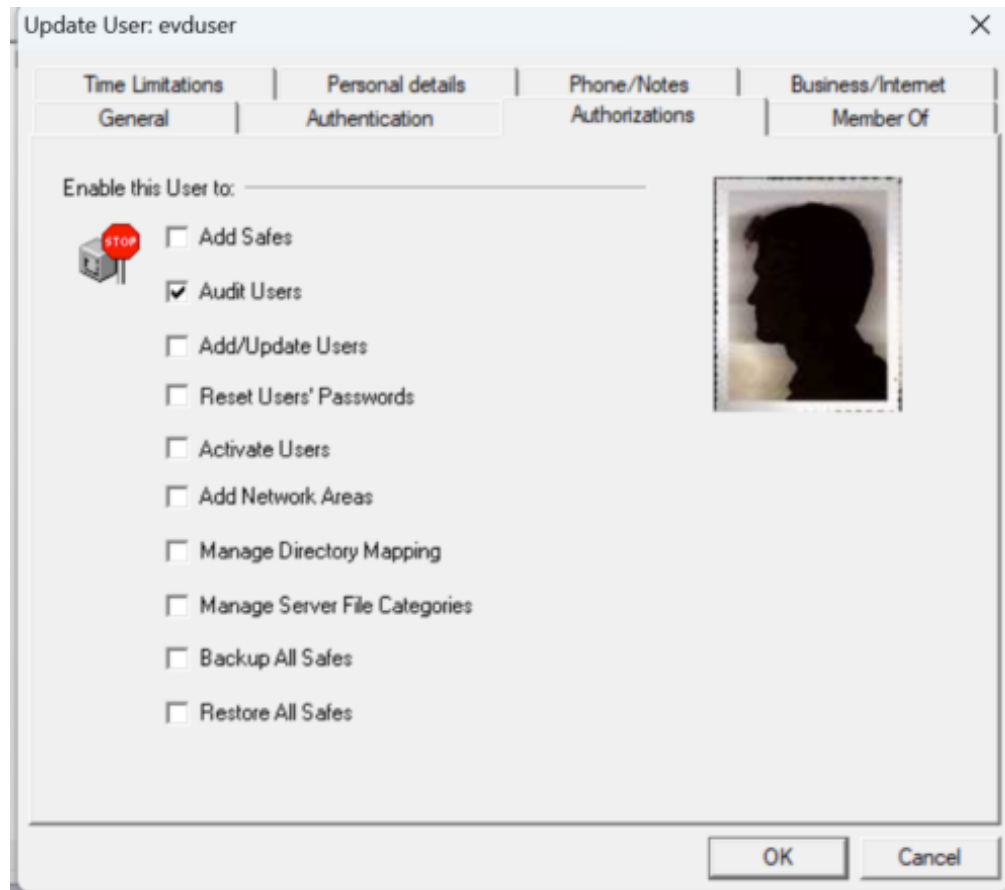
Version Compatibility

This has been tested with Privileged Authentication Manager (PAM) v13.2 (Self-Hosted). It has not been tested with Privilege Cloud.

Pre-Requisites and Installation

Pre-Requisites

- On the server that will be running BulkCPMManage, Python must be installed
- On the vault, a user must be created to make the appropriate API calls.
 - The user must be a vault user as the script uses CyberArk authentication
 - This user should have 'Audit Users' authorizations:



- Verify the following information for the Password Vault Web Access (PVWA) server
 - Hostname (FQDN)
 - PVWA URL
- License
 - The vault license file is required. This can be obtained from the System safe using the PrivateArk Client. The license file should be placed in the 'License' folder (see Properties File)
- Certificates
 - A PEM file containing the root CA as well as all intermediate CA must be configured and stored in a directory.
 1. Export the root CA and all intermediate CA into DER encoded certificates.
 2. For each certificate, convert it into a PEM format with the following command


```
openssl x509 -inform der -in certificate.cer -out certificate.pem
```

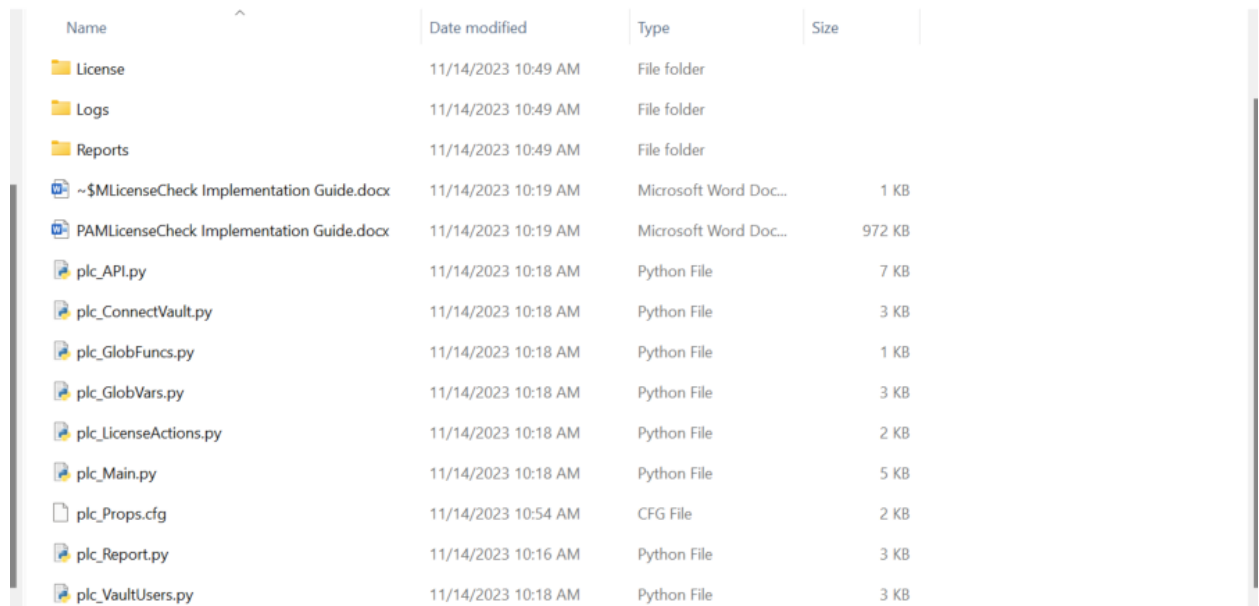
 - certificate.cer is the path to the DER certificate
 - certificate.pem is the path to the new PEM certificate
 - Perform this step on all certificates in the CA chain

3. Open each certificate with a text editor. In another new text file, copy the PEM from each certificate to the new text file. Start with the root CA and then all intermediate certificates
4. Save the file with the PEM extension. It may be necessary to remove other extensions such as .txt

Installation

The script will be packaged as a compressed (.zip) file.

1. Create a top-level directory
2. In this directory, create a directory named “CyberArk_API”
3. Unzip the zip file to the folder created in step 2. This underlying folder should be named “PAMLicenseCheck”



Name	Date modified	Type	Size
License	11/14/2023 10:49 AM	File folder	
Logs	11/14/2023 10:49 AM	File folder	
Reports	11/14/2023 10:49 AM	File folder	
~\$MLicenseCheck Implementation Guide.docx	11/14/2023 10:19 AM	Microsoft Word Doc...	1 KB
PAMLicenseCheck Implementation Guide.docx	11/14/2023 10:19 AM	Microsoft Word Doc...	972 KB
plc_API.py	11/14/2023 10:18 AM	Python File	7 KB
plc_ConnectVault.py	11/14/2023 10:18 AM	Python File	3 KB
plc_GlobFuncs.py	11/14/2023 10:18 AM	Python File	1 KB
plc_GlobVars.py	11/14/2023 10:18 AM	Python File	3 KB
plc_LicenseActions.py	11/14/2023 10:18 AM	Python File	2 KB
plc_Main.py	11/14/2023 10:18 AM	Python File	5 KB
plc_Props.cfg	11/14/2023 10:54 AM	CFG File	2 KB
plc_Report.py	11/14/2023 10:16 AM	Python File	3 KB
plc_VaultUsers.py	11/14/2023 10:18 AM	Python File	3 KB

4. The zip file will also have the proper directory structure

Properties File

The properties file (bcm_Props.props) contains necessary information to run BulkCPMManage. It is divided into three separate sections.

```

###
# Properties file for PAMLicenseCheck
#
###

###
# Installation and other directories
#
###

# Installation directory
s_BaseDir = "E:\\Utils\\CyberArk\\CyberArk_API\\PAMLicenseCheck"
# Logs directory
s_LogDir = "\\Logs\\"
# Reports directory
s_RptDir = "\\Reports\\"
# Folder for vault license
s_LicDir = "\\License\\"

###
# System variables
#
###

# Run in debug mode
b_Debug = False
# Parse a license file
b_ParseLic = True

###
# PVWA and API Information
#
###

# PVWA FQDN
s_PVWA_FQDN = "lab-comp.timslab.tsawyer.local"
# PVWA Context. This means the PVWA URL is <protocol>://<S_PVWA_FQDN>/<S_PVWAContext>[:<s_PVWA_Port>]
s_PVWAContext = "PasswordVault"
s_PVWAProtocol = "https"
s_PVWAPort = 443
# API URL
s_LogonURL = "/API/Auth/Cyberark/Logon/"
s_LogoffURL = "/API/Auth/Logoff/"
s_UsersURL = "/API/Users?ExtendedDetails=true"
s_UserTypeURL = "/API/UserTypes/"
# Path to certificate
s_Cert = E:\\Utils\\CyberArk\\CyberArk_API\\PAMLicenseCheck\\certs\\lab-comp.pem"
# Path to CA certificate (All CA must be bundled into a PEM format)
s_CAPEM = "E:\\Utils\\CyberArk\\CyberArk_API\\PAMLicenseCheck\\certs\\cachain3.pem"

```

- Installation and Logs Folder
 - s_BaseDir is the location to where you uncompressed the zip file. Example:
 - s_LogDir is the location where logs are stored
 - Note: If this will be run on Windows, backslashes must be used. See the top of the bcm_Props.props file and below. Note that backslashes (\\) must be negated with an additional backslash:
 - S_LicDir is the location where the license file was copied
- PVWA Information
 - Change the host, context, protocol, and port as necessary to reflect the current environment. Other than the host, the context, protocol, and port is valid for a majority of PAM implementations
- Certificate Information
 - If the PVWA is configured to run over https, then certificate validation is required.
 - Make sure a directory is created to store certificates. The path to the CA PEM file (see Pre-Requisites) is noted in the s_CAPEM parameter in plc_Props.cfg.
- API Information
 - This section should not have to be modified

Running PAMLicenseCheck

1. At a command prompt, type the following command:
'python -m CyberArk_API.PAMLicenseCheck.plcMain -c <path to configuration file> -l <path to license file>'
2. PVWA FQDN: The value of the PVWA FQDN is based on the parameter s_PVWAFQDN in bcm Props.props.

```
Enter the FQDN for the PVWA (Default: lab-comp.timslab.tsawyer.local) (Press 'Enter' to accept the default or E(x)it to exit: 
```

If this is correct, press 'Enter'. If it is not correct and press 'x' to stop the program. At this point, re-check the PVWA FQDN in bcm Props.props.

3. **PVWA URL:** This is asking for confirmation of the PVWA URL. If it is correct, type 'y'. Type any other key if it is not correct, and check values in the PVWA Information section of `bcm_Props.props`.
4. **Vault User credentials:** Enter the username and then password for the vault user making the API calls. Ensure this user has the appropriate permissions as stated in Pre-Requisites

Logs and Troubleshooting

plc Main.py.log

The plc Main.py.log will show basic logging when debug is not enabled.

```

2023-11-14 14:21:35.880238: #####
plc_Main.py started 2023-11-14 14:21:35.880217

2023-11-14 14:21:38.535850: plc_Main.py: Disclaimer accepted by user.
2023-11-14 14:21:38.535185: plc_Main.py: Debug mode is off.
2023-11-14 14:21:38.535283: plc_Main.py: License file parsing is on.
2023-11-14 14:21:38.541742: plc_LicenseActions.py: Total of 34 user types found.
2023-11-14 14:21:40.627663: plc_API.py: PWM URL set to https://lab-com.tinslab.tsaywer.local/PasswordVault
2023-11-14 14:21:41.691931: plc_API.py: SSL Connection verified, response code is: 200
2023-11-14 14:21:41.692637: plc_ConnectVault.py: Starting vault login.
2023-11-14 14:21:45.859591: plc_ConnectVault.py: Login URL is https://lab-com.tinslab.tsaywer.local/PasswordVault/API/Auth/CyberArk/Logon/
2023-11-14 14:21:46.523986: plc_API.py: Connection with https://lab-com.tinslab.tsaywer.local/PasswordVault/API/Auth/CyberArk/Logon/ succeeded with code 200
Data:
Authorization Header:
{"Authorization": "C2Y2S0YUjM1pTHTZS08Y0WjLLTK3MjYNDxZDYyZWYyZDQ0800TAjNzI3QzGREZFtBGHkUjQURNGFGOMmYtYzFZFY0YRQyUjQND00130BFtMzNA3ZERTHrYmXQzjC7MDaMdAwMjN1RTM5NDcMh1bTBMjLENTA1RTYjRjMjNjE1QjM0NDBBM0QzZjZlUjVGVhZmZkM2hKQjY0Y2QjU0NE1Q2AZMDA0MDAwMDA7", "Content-Type": "application/json"}
Cookies:
{"CA11111": "080080081E395E048118A29505E2236615C448A303CF5E273C3660066CE49BC6308000000", "CA22222": "D4905727CFDFE10F2EA1873162DFAE4F4C82C87CA18771DE3E2187B", "CA55555": "cyberark", "mobileState": "Desktop"}
https://lab-com.tinslab.tsaywer.local/PasswordVault/API/Auth/CyberArk/Logon/ returned 200
plc_Main.py: User types: 25
2023-11-14 14:21:47.883782: plc_Report.py: Report will be written to %Volumes/Data/Programs/CyberArk/Marketplace/CyberArk_API/PAMLicenseCheck/Reports/plc_Main.py_LicenseReport.txt
2023-11-14 14:21:47.823786: plc_ConnectVault.py: Disconnected from vault.
2023-11-14 14:21:47.930794: plc_API.py: Connection with https://lab-com.tinslab.tsaywer.local/PasswordVault/API/Auth/Logout/ succeeded with code 200
2023-11-14 14:21:47.920890:
plc_Main.py completed.

```

plc Main.py LicenseReport.txt

This shows the licensing information.

```
PAM License Check Report
Generated 2023-11-14, 14:21:47
Notes:
* Licensing information for Built-InAdmins can be ignored as this is not included in the license.
* If percentages are at or above 90%, CyberArk recommends contacting your account executive for information on adding to your existing license.
*****

Overall Statistics:
Total Users: 21

User Type: Built-InAdmins
Total Users: 5, Licensed for -1, Use Percentage: (-500.00%)
Users:
  Administrator
  Auditor
  Backup
  Operator
  DR

User Type: EPVUser
Total Users: 6, Licensed for 25, Use Percentage: (24.00%)
Users:
  EPMAgent
  tsawyer
  apladmin
  evduser
  apltest
  PTAUser
```

Note that this report is overwritten with each execution, so it is recommended to copy any reports before re-running.

Known Issues

- Do not run PAMLicenseCheck when other applications that analyze traffic are running (e.g., Wireshark, Fiddler) as this may interfere with SSL connections
- Built-InAdmins is a user type that is not part of the vault license. These statistics can be ignored