ASMIS Control Prototype

Testing and demonstration program for Appointment Scheduling Managment Information System

This Python program simulates many of the security controls that must be implemented within the user account login process. This is a command line input and output version of features that will need to be implmented in a web application framework, user feedback normally observed as HTML page updates is done through command line messages to the user/tester.

Features

- Simulates IP blacklisting using global threat intelligence lists
- Similates username and password authentication controls
- Mulitple password attack controls implemented as software controls
- Activity tracking through session state rather than IP
- Sensitive data at rest is protected via encryption
- Similates multifactor authentication
- Simulates role based access control

Control messages appear regularly throughout program execution, these are intended to demonstrate what controls are currently in process and often report stage artifacts such as session identifier or the IP address of active connection. Review the testing plan report for full insight into the control implmentation, rationale as well as test case results

Tech

This prototype is written in Python 3, no guarantee of functioning in 2.X, The prototype also requires the following three third party modules be installed for proper function. The packages are known to install without issue using pip for python 3.

pytimedinput, https://pypi.org/project/pytimedinput/ bcrypt, https://pypi.org/project/bcrypt/ stdiomask, https://pypi.org/project/stdiomask/ cryptography, https://pypi.org/project/cryptography/

Five accounts covering four different roles have been included in the prototype code, consult the testing guide for account details. You are welcome to create your own users for testing, the createUserData program avaliable at the following GIT repository can assist, https://github.com/DLeece-Essex/asmis-testing.

Installation

No installation required beyond a functional Python 3 interpreter. Download the Python program to a location of your choice and execute with the Python interpreter

EG. Windows command line execution python c:\temp\ASMIS_Control_Prototype-v1.py

If you encounter errors with the program the mostly likely cause is one or more third party modules has not been installed. EG. Windows command line execution pip install --user pytimedinput pip install --user stdiomask pip install --user bcrypt

Review online documentation if you are unfamilliar with Python package manager installation. https://www.depts.ttu.edu/hpcc/userguides/application_guides/python.packages.local_installation.php

License

*