Lab Assignment
& Solution

Cybersecurity Professional Program

Introduction to Python
for Security

File System & Error Handling

PY-04-LS9

Tasks & Questions

Note: Solutions for the instructor are shown inside the green box.

Lab Objective

Review material presented in class and conduct additional research for some topics.

Lab Mission

Research some subjects and answer questions based on the material learned in class.

Lab Duration

15–20 minutes

Requirements

- Knowledge of user input handling
- Knowledge of variable handling
- Knowledge of function handling

Resources

- Environment & Tools
    - Windows, macOS, or Linux
        - Python 3
        - PyCharm

Textbook References

- Chapter 4: File System and Error Handling
    - Section 3: Module Definition and Usage

Lab Task: Research with the SYS module

Answer questions based on the material you learned in class and research as needed.

1 Research the Python ***SYS*** module and explain its purpose.

The Python ***SYS*** module provides functions and variables that can manipulate different parts of the Python runtime environment. It enables usage of system-specific parameters and functions.

2 Briefly describe ***os.path.curdir***.

os.path.curdir returns the current directory ***('.')***.

3 Briefly describe ***os.path.isdir(dir)***.

os.path.isdir(dir) returns ***True*** if the directory exists.

4 Briefly describe ***os.path.isfile(file)***.

os.path.isfile(file) returns ***True*** if the file exists.

5 Briefly describe ***os.path.exists(dir)***.

os.path.exists(dir) returns ***True*** if the directory exists ***(full pathname or filename)***.

6 Briefly describe ***os.path.getsize(filename)***.

os.path.getsize(filename) returns the size of the file.

7 Learn about and explain what the ***Signal*** module can be used for.

A signal is an operating system feature that enables a program to be notified of an event and have it handle the event asynchronously. Signals can be generated by the system or sent from one process to another. Since signals interrupt the regular flow of a program, some operations (especially I/O) may end in an error if a signal is received when they are run. Alarms are an example of usage. Alarms are special types of signals that tell the OS to notify a program when a period of time elapses. The OS documentation mentions that this is useful to avoid the permanent blockage of I/O operations or other system calls.