Secure Operating Systems – IE2032
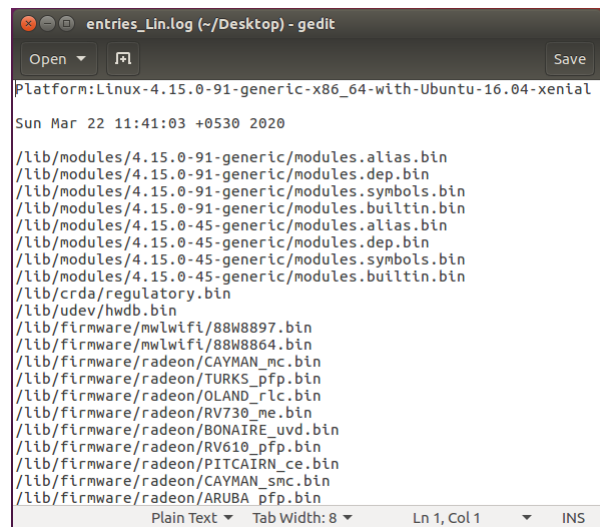
Assignment 2

Objectives:

- Installing and getting familiarize with Python
- Using library functions to interact with System Kernel
- Using basic shell commands with additional parameters

Deliverables:

Create a platform independent single python script which can do the following tasks:
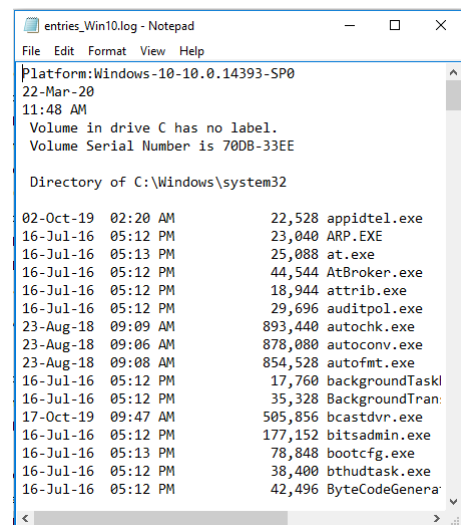
In **Linux** environment (Ubuntu):

- Platform name should be mentioned as the first line
- Second line should contain the current system date and time
- Generate a list of all the '.bin' files in directories and subdirectories inside the 'lib' folder and store them in a log file named 'entries_lin.log'.
- By the end of the procedures, 'entries_Lin.log' file should be available on the Desktop.

```
entries_Lin.log (~/Desktop) - gedit
Open ▾        🗐                               Save
Platform:Linux-4.15.0-91-generic-x86_64-with-Ubuntu-16.04-xenial

Sun Mar 22 11:41:03 +0530 2020

/lib/modules/4.15.0-91-generic/modules.alias.bin
/lib/modules/4.15.0-91-generic/modules.dep.bin
/lib/modules/4.15.0-91-generic/modules.symbols.bin
/lib/modules/4.15.0-91-generic/modules.builtin.bin
/lib/modules/4.15.0-45-generic/modules.alias.bin
/lib/modules/4.15.0-45-generic/modules.dep.bin
/lib/modules/4.15.0-45-generic/modules.symbols.bin
/lib/modules/4.15.0-45-generic/modules.builtin.bin
/lib/crda/regulatory.bin
/lib/udev/hwdb.bin
/lib/firmware/mwlwifi/88W8897.bin
/lib/firmware/mwlwifi/88W8864.bin
/lib/firmware/radeon/CAYMAN_mc.bin
/lib/firmware/radeon/TURKS_pfp.bin
/lib/firmware/radeon/OLAND_rlc.bin
/lib/firmware/radeon/RV730_me.bin
/lib/firmware/radeon/BONAIRE_uvd.bin
/lib/firmware/radeon/RV610_pfp.bin
/lib/firmware/radeon/PITCAIRN_ce.bin
/lib/firmware/radeon/CAYMAN_smc.bin
/lib/firmware/radeon/ARUBA_pfp.bin
        Plain Text ▾   Tab Width: 8 ▾     Ln 1, Col 1      ▾    INS
```

In **Windows** environment (Windows 10):

- Platform name should be mentioned as the first line
- Second line should contain system date
- Third line should contain system time
- Generate a list of all the '.exe' files available in 'C:\Windows\system32' directory and store them in a log file created in the C drive with the name of 'entries_Win10.log'.

```
entries_Win10.log - Notepad                 —   □   ×
File  Edit  Format  View  Help
Platform:Windows-10-10.0.14393-SP0
22-Mar-20
11:48 AM
 Volume in drive C has no label.
 Volume Serial Number is 70DB-33EE

 Directory of C:\Windows\system32

02-Oct-19  02:20 AM           22,528 appidtel.exe
16-Jul-16  05:12 PM           23,040 ARP.EXE
16-Jul-16  05:13 PM           25,088 at.exe
16-Jul-16  05:12 PM           44,544 AtBroker.exe
16-Jul-16  05:12 PM           18,944 attrib.exe
16-Jul-16  05:12 PM           29,696 auditpol.exe
23-Aug-18  09:09 AM          893,440 autochk.exe
23-Aug-18  09:06 AM          878,080 autoconv.exe
23-Aug-18  09:08 AM          854,528 autofmt.exe
16-Jul-16  05:12 PM           17,760 backgroundTaskl
16-Jul-16  05:12 PM           35,328 BackgroundTrans
17-Oct-19  09:47 AM          505,856 bcastdvr.exe
16-Jul-16  05:12 PM          177,152 bitsadmin.exe
16-Jul-16  05:13 PM           78,848 bootcfg.exe
16-Jul-16  05:12 PM           38,400 bthudtask.exe
16-Jul-16  05:12 PM           42,496 ByteCodeGenera
<                                            >
```

NOTE: Same python script should be able to operate on both environments. Refer lab sheets to get an insight on how to bridge the gap between terminal commands and a program.

HINT: You can use built-in python libraries 'os' and 'platform' (import os, import platform)
You may need to execute some shell commands inside your python script.
Use [command] >> [file.type] command format to redirect output to a file.