Service Monitor ReadMe:

About this project:

In this project I implemented a service monitor in Python. This tool monitor the services running in the system, and report on changes that can be critical for us as SOC people.

More about the project:

The tool support Windows and Linux – Ubuntu.
This tool has 2 modes:

Monitor: For X time that the user sets, the program samples every X time all the services running on the computer and shows whether a change is observed from the previous sample. Is there a service that is no longer running, or is there a service a new one is running in the system.
In this mode there are 2 relevant log files:
"service_list.log" - which will contain the titles of the time in which a sample was taken and below each title all the services and their status for that time sample.
"status_log.log" - This log file is for tracking purposes.
This file show the changes that occurred between samples for the changed services.

Manual: In this mode we will use the "service_list.log" file to load 2 samples from different time frames and make a comparison. The program gets a date and time for 2 events, loads the 2 samples from the file, and displays changes like the monitor mode (A new process created in the more recent sample, a process that no longer runs in the more recent sample, etc.) All those changes will be written in "status_log.log" file.

Libraries Information:

Os- for operating system check if file exist, also for running ubuntu command for checking the services, and in order to see last modified of some files.

Platform- to check in which operating system to support.

Re- To find the dates and times in the relevant files (there was a use of date and time regex).

Sys- to use argv (input from terminal).

Datetime- for input user and record the date and time of each log.

Subprocess- for running windows command for checking the services.

Csv- to execute services to a csv file

Smtplib- connection with gmail.com to execute email hacker track information.

Project structure:

The project has 2 main classes: "ServiceMonitor " , "Hacker_Identifier".
The first is the main program, it checking the relevant operating system, running the appropriate command to check services status and taking samples according to user request (monitor or manual).
The last one purpose is to report about unwanted changes in the log files and send an alert email about it.
In fact, first of all we would like to run our tool at a given time in monitor mode to check what are the open servers and at any X time get a sample of the changes that have occurred.
After operating the tool in monitor mode we can run each time we want the tool in manual mode to get samples of the changes that have occurred between 2 different times.
Before activation of the tool, we can check that no change has been made to our log files by using "Hacker_Identifier".
We will run the "Hacker_Identifier" and it will check when the file was last updated and whether it was done through our tool or by malicious sabotage by someone else, and if so, we will receive an email notification about it!

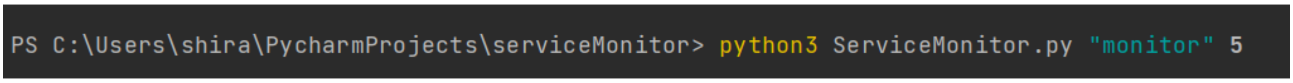
How to run the project:

Enter the project directory, open cmd (Windows) or terminal (Linux) through project's path and run the following command:

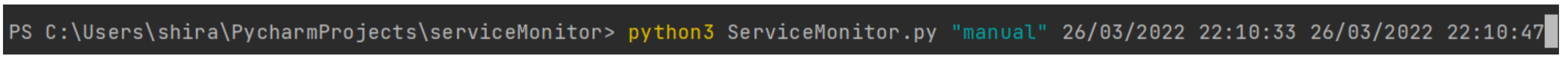
* python3 ServiceMonitor.py "monitor" "XXX" to run the project in monitor mode every XXX time.

OR

* python3 ServiceMonitor.py "manual" D1/M1/Y1 H1:M1:S1 D2/M2/Y2 H2:M2:S2 to run the project in manual mode between first and second dates.

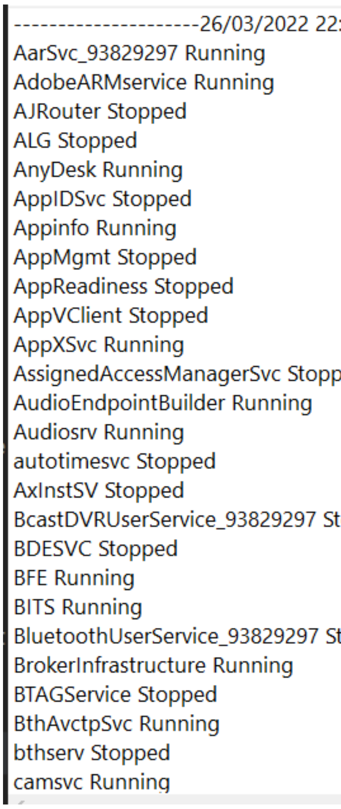


```
PS C:\Users\shira\PycharmProjects\serviceMonitor> python3 ServiceMonitor.py "monitor" 5
```

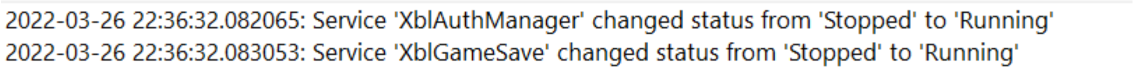


```
PS C:\Users\shira\PycharmProjects\serviceMonitor> python3 ServiceMonitor.py "manual" 26/03/2022 22:10:33 26/03/2022 22:10:47
```

Then you will get those log files in your project directory:



```
--------------------26/03/2022 22:10:33------------------------
AarSvc_93829297 Running
AdobeARMservice Running
AJRouter Stopped
ALG Stopped
AnyDesk Running
AppIDSvc Stopped
Appinfo Running
AppMgmt Stopped
AppReadiness Stopped
AppVClient Stopped
AppXSvc Running
AssignedAccessManagerSvc Stopped
AudioEndpointBuilder Running
Audiosrv Running
autotimesvc Stopped
AxInstSV Stopped
BcastDVRUserService_93829297 Stopped
BDESVC Stopped
BFE Running
BITS Running
BluetoothUserService_93829297 Stopped
BrokerInfrastructure Running
BTAGService Stopped
BthAvctpSvc Running
bthserv Stopped
camsvc Running
```

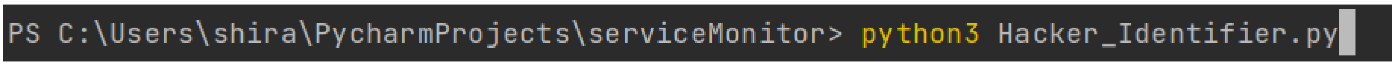


```
2022-03-26 22:36:32.082065: Service 'XblAuthManager' changed status from 'Stopped' to 'Running'
2022-03-26 22:36:32.083053: Service 'XblGameSave' changed status from 'Stopped' to 'Running'
```

"status_log.log"

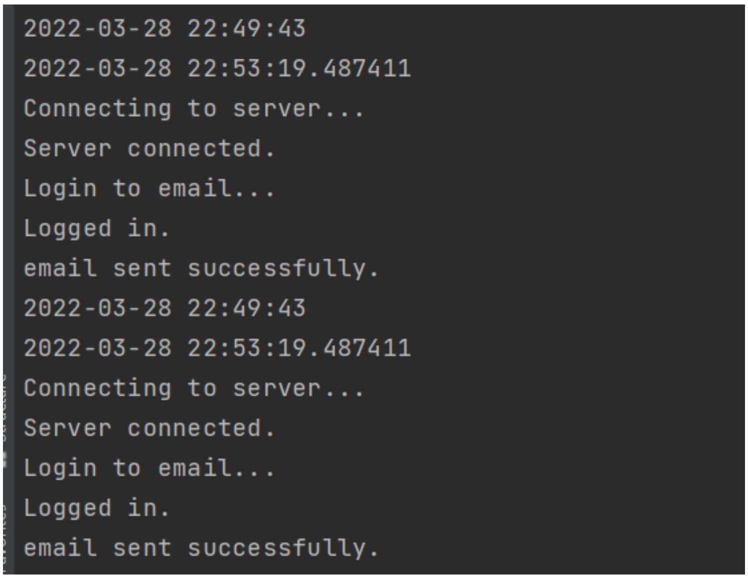
"services_list.log"

To run the "Hacker_Identifier" :



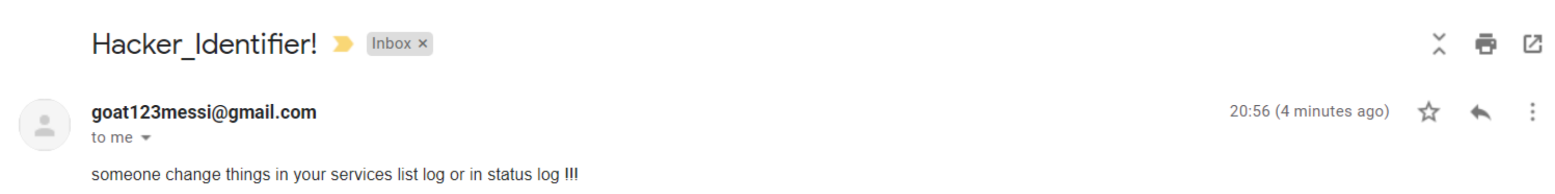
```
PS C:\Users\shira\PycharmProjects\serviceMonitor> python3 Hacker_Identifier.py
```

Then you will see this in your terminal, if there was a malicious touch on the log files:



```
2022-03-28 22:49:43
2022-03-28 22:53:19.487411
Connecting to server...
Server connected.
Login to email...
Logged in.
email sent successfully.
2022-03-28 22:49:43
2022-03-28 22:53:19.487411
Connecting to server...
Server connected.
Login to email...
Logged in.
email sent successfully.
```

And you can go check your email box and see this message:



Hacker_Identifier!

goat123messi@gmail.com
to me

20:56 (4 minutes ago)

someone change things in your services list log or in status log !!!