Ransomware detector

Written by: Adi Weisberg

Id: 313245268

As a learning assignment, we were asked to write a Python script that verifies that no file is encrypted in a specific path.

The script is based on the following assumptions:

- The script will only check text files.
- There will be text files (.txt) containing only ascii characters.
- Encryption does not necessarily occur on all folder files together, or on an entire file necessarily.

The script I wrote is divided into the following sections:

1) **my_handler.py**

I choose to use the observer's pattern design within this class. The purpose of this class is to define observers for all files in the given path. The observers are waiting for an event to trigger.
The events that require attention in this case are:

- Case of creating a new file – it might be encrypted.
- Case of modifying a new file - it might be encrypted.
- Case of moving or renaming existed file – One indication to know how to check Ransomware attack is when there is an increase in file renames and your data becomes encrypted. Therefore, if we identify many of file renames, it's potential Ransomware issue.
- Case of deleting an existed file – the attacker might encrypt the data to new file and delete the source file. I choose only to alert about deleting a file, rather than checking for encryption – because no file has changed.

As the event occurs, the subject tells the observers that it has occurred

In this file I used the following imports:

- **time** from Python will be used to sleep the main loop
- **os** provides a portable way of using operating system dependent functionality.
- **watchdog.observers.Observer** is the class that will watch for any change, and then dispatch the event to specified the handler.
- **watchdog.events.PatterMatchingHandler** is the class that will take the event dispatched by the observer and perform some action. This class include:
  - **on_created:** Executed when a file or a directory is created
  - **on_modified:** Executed when a file is modified or a directory renamed
  - **on_moved:** Executed when a file or directory is moved
  - **on_deleted:** Executed when a file or directory is deleted.

2) ransomware_check.py

This script performs the following tests for each file in the given path, which are intended to determine whether the file is encrypted:

- ✓ Validation of words in English:
  - If 3 or more errors are found in a row of a file, this line will be recognized as encrypted.
  - If a file contains a line containing a long word of characters that are not identified as English, this line will be recognized as encrypted.
- ✓ This test is performed in the file "my_handler.py" - If the number of files that renamed is equal to or greater than 3, they are suspected of encryption.
- ✓ If the file has passed all tests successfully, a message will be printed stating that the file is not encrypted.

In this file I used the following imports:

- **enchant.checker** provides a class **SpellChecker** - SpellChecker objects are created in the same way as Dict objects - by passing a language tag to the constructor. The "set_text" method is used to set the text which is to be checked. Once this is done, the SpellChecker object can be used as an iterator over the spelling mistakes in the text and return the errors.

- **glob** - The glob module implements globbing of directory contents. The glob.glob returns the list of files with their full path (unlike os.listdir()) and is more powerful than os.listdir that does not use wildcards.

```
round 1
round 2
round 3
deleted: C:\Users\עד\PycharmProjects\Ransomware_detetor\test_folder\test4.txt
round 4
modified: C:\Users\עד\PycharmProjects\Ransomware_detetor\test_folder\test6.txt
```

I chose to "encrypt" lines 1, 2, 7, 8, 9 - and indeed, the script identified the suspicious lines.

```
[*] Checking the following files:
 ['test_folder\\test1.txt', 'test_folder\\test2.txt', 'test_folder\\test3.txt', 'test_folder\\test4.txt', 'te
The file 'st_folder\test1.txt' is not encrypted!
The file 'st_folder\test2.txt' is not encrypted!
The content of file test_folder\test3.txt in row 1 may be encrypted!
The content of file test_folder\test3.txt in row 2 may be encrypted!
The content of file test_folder\test3.txt in row 7 may be encrypted!
The content of file test_folder\test3.txt in row 8 may be encrypted!
The content of file test_folder\test3.txt in row 9 may be encrypted!
The file 'st_folder\test4.txt' is not encrypted!
The file 'st_folder\test5.txt' is not encrypted!
The content of file test_folder\test6.txt in row 1 may be encrypted!
The content of file test_folder\test6.txt in row 2 may be encrypted!
The content of file test_folder\test6.txt in row 7 may be encrypted!
The content of file test_folder\test6.txt in row 8 may be encrypted!
The content of file test_folder\test6.txt in row 9 may be encrypted!
```

Hope you liked the script,

313245268 - Adi.