

[illegible]

[DFC-2023-251] Write-Up

작성자	류나연
분석 일자	2024.05.28~
작성 일자	2024.05.30
분석 대상	Users.zip
문서 버전	1.0
작성자 E-mail	01star01ek@gmail.com

0. 목차

1. 문제3

2. 분석 도구3

3. 환경3


4. Write-Up.....4

5. Flag.....9

6. 별도 첨부 10

7. Reference 11

1. 문제

URL	-
문제 내용	<p>[번역본] (*원본 영문 문제는 별도 첨부에 작성되어 있음)</p> <p>수사관들은 범죄현장에서 범행에 사용된 것으로 추정되는 맥북을 압수합니다. 하지만 범행의 용의자가 특정되지 않아 사건 해결이 어렵습니다. 증거 파일을 분석하여 범인의 신원을 확인합니다.</p> <p>대상 해시(MD5) Users.zip 56E3072B8D12D449B57E47A90BB35CAF</p> <p>문의사항</p> <p>1) 범죄와 관련된 것으로 보이는 모든 파일을 찾아내고, 파일명과 업로드 또는 다운로드 시간(UTC+9)(20점)을 파악합니다</p> <p>2) 용의자의 이름과 이메일 주소를 확인합니다. (80점)</p> <p>3) dbx를 복호화하기 위해 도구를 제출합니다. (150점)</p>
문제 파일	 <p>Users.zip</p>
문제 유형	System forensic
난이도	3 / 3

2. 분석 도구

도구명	다운로드 링크	Version
DB Broser for SQLite	https://sqlitebrowser.org/dl/	3.12.2

3. 환경

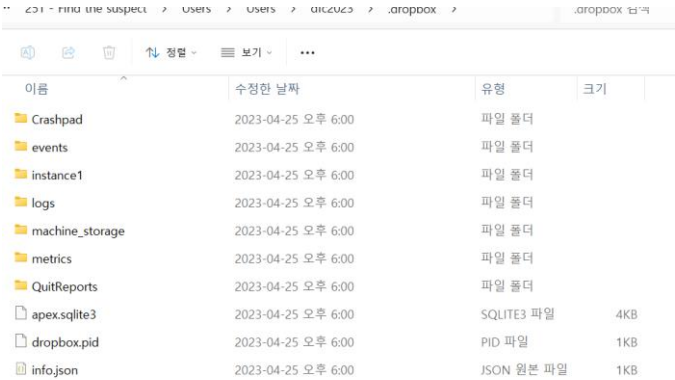
OS
Windows 11 Home

4. Write-Up

파일명	Users.zip
용량	3.36GB
SHA256	8635a9c7ca7de9a5cefab1bd008f15ee0d1d91182cb249bd9d4801197c5988a7
Timestamp	2023-05-06 01:59:10

Q1. 범죄와 관련된 것으로 보이는 모든 파일을 찾아내고, 파일명과 업로드 또는 다운로드 시간(UTC+9)(20 점)을 파악합니다

1. 구조 확인

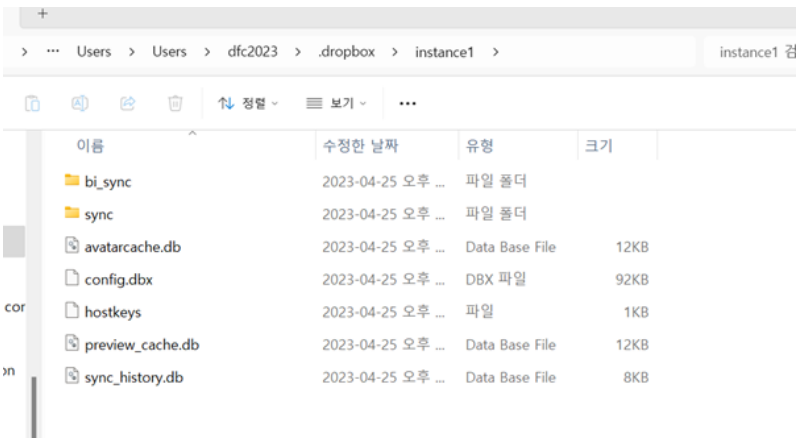


이름	수정한 날짜	유형	크기
Crashpad	2023-04-25 오후 6:00	파일 폴더	
events	2023-04-25 오후 6:00	파일 폴더	
instance1	2023-04-25 오후 6:00	파일 폴더	
logs	2023-04-25 오후 6:00	파일 폴더	
machine_storage	2023-04-25 오후 6:00	파일 폴더	
metrics	2023-04-25 오후 6:00	파일 폴더	
QuitReports	2023-04-25 오후 6:00	파일 폴더	
apex.sqlite3	2023-04-25 오후 6:00	SQLITE3 파일	4KB
dropbox.pid	2023-04-25 오후 6:00	PID 파일	1KB
info.json	2023-04-25 오후 6:00	JSON 원본 파일	1KB

[사진 1] dropbox 폴더 내 구조

문제 파일을 확인하기 위해 전체적으로 폴더들을 훑으며 접근 가능한 항목들에 무엇이 있는지 판별하였다.

2. db 파일 확인을 통한 파일 업로드, 다운로드 기록 확보



이름	수정한 날짜	유형	크기
bi_sync	2023-04-25 오후 ...	파일 폴더	
sync	2023-04-25 오후 ...	파일 폴더	
avatarcache.db	2023-04-25 오후 ...	Data Base File	12KB
config.dbx	2023-04-25 오후 ...	DBX 파일	92KB
hostkeys	2023-04-25 오후 ...	파일	1KB
preview_cache.db	2023-04-25 오후 ...	Data Base File	12KB
sync_history.db	2023-04-25 오후 ...	Data Base File	8KB

[사진 2] instance1폴더 내 들어있는 db 파일들

해당 db파일들을 다 확인해본 결과 sync_history.db 파일에서 유의미한 결과를 발견할 수 있었다.

3. 다운로드 시간 변환

int_type	direction	file_id	local_path	server_path	offer_user	timestamp
		원본	원본	원본	원본	
1	download	XZVILS2XNGAAAAAAAABg	A:\ers\dc\2023\Dropbox\압글주소O\MP0.txt	368257977/압글주소O\MP0.txt		0 1681081949
2	download	XZVILS2XNGAAAAAAAABh	A:\ers\dc\2023\Dropbox\1월_레스합화한_판매_장부.xlsx	368257977/1월_레스합화한_판매_장		0 1681081949
3	download	XZVILS2XNGAAAAAAAACa	A:\ers\dc\2023\Dropbox\1월_대과_판매_장부.xlsx	368257977/1월_대과_판매_장부.xlsx		0 1681081949
4	download	XZVILS2XNGAAAAAAAACQ	A:\ers\dc\2023\Dropbox\교육_결보.xlsx	368257977/교육_결보.xlsx		0 1681081949
5	upload	XZVILS2XNGAAAAAAAADg	A:\ers\dc\2023\Dropbox\레스합화한_생물품.jpg	368257977/레스합화한_생물품.jpg		0 1679625730
6	upload	XZVILS2XNGAAAAAAAADp	A:\ers\dc\2023\Dropbox\면허가정소2.png	368257977/면허가정소2.png		0 1679625947
7	upload	XZVILS2XNGAAAAAAAACw	A:\ers\dc\2023\Dropbox\면허가정소1.png	368257977/면허가정소1.png		0 1679622508

[사진 3] sql lite를 이용해서 확인한 db 내용

해당 표내 Time stamp 에 적힌 시간들은 변환기 <https://ko.rakko.tools/tools/29/> 를 이용해서 utf-9 기준 시간대로 변경해주었다.

파일명	시간
입금주소.txt	2023-04-09 23:12:29
1 월_메스암페타민_판매_장부.xlsx	2023-04-09 23:12:29
1 월_대마_판매_장부.xlsx	2023-04-09 23:12:29
고객_정보.xlsx	2023-04-09 23:12:29
메스암페타민_샘플용.jpg	2023-02-17 08:48:59
던지기장소 2.png	2023-02-17 08:35:47
던지기장소 1.png	2023-02-17 08:35:38

4. 숨겨진 구문 찾기

파일 편집 보기

MDfIeaddownloadXv1LSzXoNQAAAAAAAAAAcW/Users/dfc2023/Dropbox/1.png3982357937/던지기장소_1.png<ZD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAADA/Users/dfc2023/Dropbox/1.png3982357937/던지기장소_2.png<<ZD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAAADQ/Users/dfc2023/Dropbox/1.png3982357937/메스암페타민 샘플용.jpg기
MDfIeaddownloadXv1LSzXoNQAAAAAAAAAACG/Users/dfc2023/Dropbox/1.png3982357937/대마구매자_nomad_별그 캡처.jpgGD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAACQ/Users/dfc2023/Dropbox/1.png3982357937/고객 정보.xlsx3FDD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAACA/Users/dfc2023/Dropbox/1.png3982357937/1월 대마 판매 장부.xlsx3FZD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAABw/Users/dfc2023/Dropbox/1.png3982357937/1월 메스암페타민 판매 장부.xlsx3FD

MDfIeaddownloadXv1LSzXoNQAAAAAAAAABg/Users/dfc2023/Dropbox/1.png3982357937/입금주수(XMR).bt3982357937/입금주수(XMR).bt3F

출 18, 열 267 35/7,823자 100% Unix (LF) UTF-8

[사진 4] 메모장으로 열어본 sync_history.db

정답 확인을 해보니 답이 다르길래 다른 파일을 찾아보았으나 없어 이를 메모장과 hxd 로 열어보았다. 그러면 숨겨져 있던 내역이 확인되었다. 이부분도 놓치지 말아야 하니 조심하시길 바란다.

해당 파일의 timestamp 를 구하기 위해서는 HXD 와 진법 계산기 사이트
<https://www.digikey.kr/ko/resources/conversion-calculators/conversion-calculator-number-conversion> 를 사용했다.

```

Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F Decoded text
00001BC0 67 63 EF 3C 63 81 3E 06 0A 15 13 19 39 81 4F 65 gci<c.>....9.Oe
00001BD0 08 04 66 69 6C 65 61 64 64 75 70 6C 6F 61 64 58 ..fileadduploadX
00001BE0 7A 56 31 4C 53 7A 58 6F 4E 51 41 41 41 41 41 41 zVILSzXoNQAAAAA
00001BF0 41 41 41 44 51 2F 55 73 65 72 73 2F 64 66 63 32 AADQ/Users/dfc2
00001C00 30 32 33 2F 44 72 6F 70 62 6F 78 2F E1 84 86 E1 023/Dropbox/á„tá
00001C10 85 A6 E1 84 89 E1 85 B3 E1 84 8B E1 85 A1 E1 86 ...;á„há„á„;á„tá
00001C20 B7 E1 84 91 E1 85 A6 E1 84 90 E1 85 A1 E1 84 86 á„'á„!á„.á„;á„t
00001C30 E1 85 B5 E1 86 AB 5F E1 84 89 E1 85 A2 E1 86 B7 á„pát« á„há„cát·
00001C40 E1 84 91 E1 85 B3 E1 86 AF E1 84 8B E1 85 AD E1 á„'á„'át~á„<á„.á
00001C50 86 BC 2E 6A 70 67 33 39 38 32 33 35 37 39 33 37 t4.jpg3982357937
00001C60 3A 2F EB A9 94 EC 8A A4 EC 95 94 ED 8E 98 ED 83 :/e@“iŠ“i·“iŽ“if
00001C70 80 EB AF BC 5F EC 83 98 ED 94 8C EC 9A A9 2E 6A eē“4 if“i““EiŠ@.j
00001C80 70 67 63 EF 3F 7E 00 00 00 00 00 C3 15 13 19 pgc17?.....Ã...
00001C90 39 81 45 73 08 04 66 69 6C 65 61 64 64 75 70 6C 9.Es..fileaddupl
00001CA0 6F 61 64 58 7A 56 31 4C 53 7A 58 6F 4E 51 41 41 oadXzVILSzXoNQAA
00001CB0 41 41 41 41 41 41 41 43 67 2F 55 73 65 72 73 2F AAAAAACg/Users/
00001CC0 64 66 63 32 30 32 33 2F 44 72 6F 70 62 6F 78 2F dfc2023/Dropbox/
00001CD0 E1 84 83 E1 85 A2 EB A7 88 E1 84 80 E1 85 AE E1 á„fá„cēs“á„éá„@á
00001CE0 84 86 E1 85 A2 E1 84 8C E1 85 A1 5F 6E 6F 6D 61 „tá„cá„Eá„j_noma
00001CF0 64 5F E1 84 90 E1 85 A6 E1 86 AF E1 84 80 E1 85 d á„.á„;á„t á„éá„
00001D00 B3 5F E1 84 8F E1 85 A2 E1 86 B8 E1 84 8E E1 85 ' á„.á„cát, á„Žá„
00001D10 A5 2E 6A 70 67 33 39 38 32 33 35 37 39 33 37 3A ¥.jpg3982357937:
00001D20 2F EB A9 94 EC 8A A4 EC 95 94 ED 8E 98 ED 83 /e@“iŠ“i·“iŽ“if

```

[사진 5] 구한 timestamp 값 (16진수)

먼저 이미 알고 있는 timestamp 값을 이용하여 1676623739 의 16 진수를 구했고 이를
 통해 구한값을 HXD 에 검색하여 확장자 뒤 4 바이트가 timestamp 값을 의미함을 알게
 되었다. 따라서 2 번째 jpg 뒤의 4 바이트 값을 구하였다. 이를 10 진수로 바꾸면
 1676623739 였고 결국 2023-02-17 08:48:59 해당 시간이 답이다.

파일명	시간
입금주소.txt	2023-04-09 23:12:29
1 월_메스암페타민_판매_장부.xlsx	2023-04-09 23:12:29
1 월_대마_판매_장부.xlsx	2023-04-09 23:12:29
고객_정보.xlsx	2023-04-09 23:12:29
메스암페타민_샘플용.jpg	2023-02-17 08:48:59
던지기장소 2.png	2023-02-17 08:35:47
던지기장소 1.png	2023-02-17 08:35:38
대마구매자_nomad_텔그_캡처.jpg	2023-02-17 08:48:59

Q2. 용의자의 이름과 이메일 주소를 확인합니다. (80 점)

Q3. dbx 를 복호화하기 위해 도구를 제출합니다. (150 점)

해당 문제는 연관되어 있어 같이 답변하겠습니다.

왜냐하면 dbx 외에는 유의미한 파일이 없어 분명히 이름과 이메일이 dbx 내에 있을 것이라고
 추측되었기 때문이다.

[WHS-2] .iso

1. dbx 파일 복호화 하기

dbx 이외의 타형식의 파일에서 유의미한 정보가 적혀있지 않아 dbx 파일을 확인하고자 했다.

dbxViwer 를 통해 확인해봤으나 확인이 불가능했다. 따라서 dbx 파일을 살펴보니 해당 파일이 경우에 따라 암호화가 되어 있을수도, 안되어 있을수도 있다고 하였다. 이 파일의 경우 보이지 않으니까 암호화가 되었다고 판단되어 이에 대해 분석후 사용 가능한 툴이 있을지 찾아보았다.

그러나 <https://github.com/dnicolson/dbx-keygen-macos> 와 같이 사용자 로그인 비밀번호를 알고 있어야 하거나, 실제 환경에서 진행될때만 가능한 툴들만 있었다. 이에 따라 해당 비밀번호를 알아내기 위해서는 직접 코드를 짜야 한다는 결론에 다다랐다. 따라서 아래의 링크들을 참고하여 코드를 작성하였다.

1. <https://book.hacktricks.xyz/generic-methodologies-and-resources/basic-forensic-methodology/specific-software-file-type-tricks/local-cloud-storage>
2. <https://github.com/dnicolson/dbx-keygen-macos>

즉 이를 해결하기 위해서는 암호키가 필요하고 -> 해당 키는 DB_KEY 라는 이름을 가지는데 이 키는 -> PBKDF2(평문키 + 고정된 SALT 값) 함수를 통해 생성되며 -> 우리 hostkeys 파일에 있는 키는 AES-128-CBC 라는 암호화 방식으로 암호화 된 상태인 것이다.

이때 PBKDF2 는 해시 함수를 1066 번 반복하며 DROPBOX 는 고정된 salt 값을 사용한

따라서 우리의 복호화 방법은

1. Hoskkeys 파일에서 USER_KEY 을 복호화한다.
2. 복호화된 키와 고정된 SALT 값을 PBKDF2 해시 함수에 입력한다.
3. 해시 함수를 1066 번 반복한다.
4. DB_KEY 로 복호화하여 확인한다.

입다.

이때 필요한데 없는 정보는 바로 USER_KEY 를 복호화할 때 사용할 key, iv 값이다.

이를 알아보기 위해서 <https://github.com/dnicolson/dbx-keygen-macos> 사이트에서 해당 코드들을 분석하였다.

```
def id2s(self, _id):
```

```
    return hashlib.md5('ia9%sX' % _id + 'a|ui20').digest()
```



[WHS-2] .iso

이 구문을 통해 key 값이 해당 방법으로 정해짐을 알 수 있다. 그러나 iv 값은 어떻게 정해지는지 알 수 없었다. 이에 따라 user_key 값을 구할 수 가 없다는 결론에 다다랐으며 이를 구현해보고자 했으나 중간에 막혀 해결하지 못하였다.

➔ 해당 부분에 대한 고민 및 배움이 더 필요해보인다.

5. Flag

1.

파일명	시간
입금주소.txt	2023-04-09 23:12:29
1 월_메스암페타민_판매_장부.xlsx	2023-04-09 23:12:29
1 월_대마_판매_장부.xlsx	2023-04-09 23:12:29
고객_정보.xlsx	2023-04-09 23:12:29
메스암페타민_샘플용.jpg	2023-02-17 08:48:59
던지기장소 2.png	2023-02-17 08:35:47
던지기장소 1.png	2023-02-17 08:35:38
대마구매자_nomad_텔그_캡처.jpg	2023-02-17 08:48:59

6. 별도 첨부

- 원본 영문 문제

Description Investigators impound a MacBook believed to have been used in the crime at the crime scene. However, it is difficult to solve the case because the suspect of the crime is not specified. Analyze the evidence file to identify the criminal.

Target	Hash (MD5)
Users.zip	56E3072B8D12D449B57E47A90BB35CAF

Questions

- 1) Find all files that appear to be related to the crime, and identify the file name and upload or download time (UTC+9) (20 points)
- 2) Identify suspect's name and email address. (80 points)
- 3) Submit the tool to decrypt the dbx. (150 points)

7. Reference

- Dbx 파일
<https://book.hacktricks.xyz/generic-methodologies-and-resources/basic-forensic-methodology/specific-software-file-type-tricks/local-cloud-storage>
- MAC dbx복호화 툴
<https://github.com/dnicolson/dbx-keygen-macos>