암호화된 파일의 비밀번호 복구 연구 동향

윤세영[1], 김현지[2], 서화정[3]
[1]한성대학교 융합보안학과 석사과정
[2]한성대학교 정보컴퓨터공학과 박사과정
[3]한성대학교 융합보안학과 교수
sebbang99@gmail.com, khj1594012@gmail.com, hwajeong84@gmail.com

Research Trends on Password Recovery of Encrypted Files

Se-Young Yoon[1], Hyun-Ji Kim[2], Hwa-Jeong Seo[3]
[1,3]Dept. of Convergence Security, Hansung University
[2]Dept. of Computer Information Engineering, Hansung University

요 약

IT 기술과 매체의 발전으로 자료의 디지털화가 진행되면서 디지털 증거는 현대의 범죄 수사에서 중요한 부분을 차지하고 있다. 이러한 디지털 증거들이 암호화되어 있는 일이 빈번하게 발생함에 따라, 수사관은 수사 과정에서 직접 암호화를 해제해야 하는 어려움을 겪고 있다. 해당 문제에 대응하기 위해 암호화된 파일의 비밀번호를 복구하는 연구가 활발히 진행되어 왔으며, 암호 연산을 빠르게 처리할 수 있는 프로세서를 활용하여 복구 속도를 향상시키는 방안 또한 연구되고 있다. 본 논문에서는 현재 사용되고 있는 비밀번호 복구 도구들을 분석하고, 높은 사용률을 보이는 문서들의 비밀번호를 복구하는 기존 연구들과 함께 향후 연구의 방향성을 살펴본다.

1. 서론

소유자가 비밀번호를 분실하거나 정보 접근을 방해하기 위한 암호화로 인해 권한을 가진 수사관이 파일에 접근하는 데 어려움을 겪는 경우가 많다. 복잡하게 설정된 비밀번호는 복구 과정에서 상당한 시간이 소요되며, 경우에 따라 비밀번호를 복구하는 것 자체가 불가능해 정보에 접근하지 못할 수 있다. 비밀번호는 컴퓨터 시스템에서 기본적인 인증 형식으로 활용되기 때문에, 비밀번호를 복구하는 기법에 대한 연구는 꾸준히 진행되고 있다[1]. 본 논문은 현재에도 사용되고 있는 주요 비밀번호 복구 도구들과 함께 높은 사용률을 보이는 압축 파일, PDF 및 Excel 문서를 대상으로 비밀번호를 복구하는 기존의 연구에 대해서 살펴본다.

2. 관련 연구

2.1 비밀번호 복구 기술

일반적인 비밀번호 복구 방법에는 무차별 대입 공격(Brute-force attack)과 사전 공격(Dictionary attack), 레인보우 테이블 공격(Rainbow Table attack)[2]이 있다. 무차별 대입 공격은 사용할 수 있는 모든 문자와 숫자를 조합하여 대입하는 방식이므로 많은 시간과 자원을 필요로 한다. 따라서 비밀번호로 자주 쓰이는 단어를 사전 파일로 만들어 두거나, 특정 암호 알고리즘으로 미리 해시된 해시값을 레인보우 테이블에 저장해 둔다면 기존의 무차별 대입 공격보다 더 적은 시간과 자원으로 비밀번호를 복구할 수 있다. 그러나 이 방식 또한 비밀번호나 해시값이 사전과 테이블에 저장되어 있지 않으면 결국 모든 경우의 수를 고려할 수밖에 없다.

암호화된 파일에는 파일 고유의 암호 알고리즘으로 해시된 해시값이 저장되어 있다. 해당 암호 알고리즘으로 예상 비밀번호에 대응하는 해시값을 만들어낸 뒤, 파일에 저장된 해시값과 비교하여 일치하는 값을 찾는 방식으로 비밀번호를 복구한다. 이 방법은 단순하지만 가능한 해시값을 전부 비교해야 한다는 점에서 많은 양의 계산이 필요하다. 따라서 길이가 길고 복잡한 비밀번호도 빠르게 복구하기 위해 CPU 대신 GPU만 사용하거나, CPU와 GPU를 함께 사용하여 연산 과정의 처리 속도를 높이는 연구가 진행되고 있다[3].

2.2 압축 파일

압축 파일은 압축 알고리즘을 사용하여 아카이브 파일의 크기를 줄인 것이다. 아카이브 파일은 한 개

이상의 파일을 하나로 묶어 저장한 파일이며, 파일 디렉터리의 구조와 같이 파일에 대한 설명을 담은 메타데이터가 포함되어 있다. 일반적으로 압축 파일은 데이터 압축 시 무손실 압축(Lossless compression)을 수행하며, 일부 또는 전체 암호화가 가능하고, RAR, ZIP, 7z의 파일 포맷을 가지고 있다.

암호화된 압축 파일 중 ZIP 파일 포맷을 갖는 파일의 경우 다음과 같은 방식으로 암호화 및 복호화 과정을 거친다. 우선 주어진 비밀번호를 바탕으로 해시 함수인 PBKDF2(Password-Based Key Derivation Function 2)를 사용하여 AES 키를 생성한다. 이때, PBKDF2 함수는 HMAC-SHA1 알고리즘을 1000번 반복 실행하여 암호화 과정을 복잡하게 만들기 위해 사용된다. 이렇게 생성된 AES 키는 ZIP 파일을 암호화할 때 사용되며, 암호화된 ZIP 파일을 복호화 할 때에도 입력된 비밀번호에 대한 값을 비교하기 위해 사용된다[4].

2.3 비밀번호 복구 관련 도구

2.3.1 hashcat

hashcat[5]은 MIT 라이센스에 따라 오픈소스로 공개되어 있는 비밀번호 복구 도구이다. MD5, SHA512 등 350개 이상의 다양한 해시 알고리즘을 이용하여 암호화된 파일 및 시스템의 비밀번호를 복구하는 데 사용된다. 현재 hashcat에서는 PDF, Microsoft Office와 각기 다른 버전들을 모두 포함하여 21개의 문서 파일 비밀번호 복구 포맷을 지원하고 있으며, 압축 파일로는 RAR, 7-Zip, WinZip, PKZIP 등 24개의 포맷을 이용할 수 있다. hashcat은 CPU 및 GPU 가속을 지원하기 때문에 대규모의 무차별 대입 공격이 가능하다. 공식 웹에서 다운로드할 수 있는 최신 버전은 v6.2.6이며, 2022년 9월에 업데이트되었다.

2.3.2 John the ripper

John the Ripper[6]는 Unix 버전(Linux, AIX, QNX, Solaris, BSD)과 MacOS, Windows의 운영체제에서 사용할 수 있는 오픈 소스 비밀번호 복구 도구이다. 기존의 John the Ripper는 CPU 플랫폼에서만 이용할 수 있었지만, 2023년 3월 NVIDIA GPU 드라이버와 함께 Amazon Linux 2에서도 사용할 수 있도록 업데이트되었다. 현재 John the Ripper에서는 RAR, ZIP, 7-Zip에 대한 포맷을 지원하고 있으며, AES로 암호화된 WinZip은 JtR 1.7.8-jumbo-2 이상에서 지원하고 있다. 문서 파일로는 PDF, docx, iWork 포맷을 사용할 수 있다. 이외에도 'john --list=formats' 명령어를 사용하여 지원하는 포맷을 확인해 볼 수 있다.

3. 암호화된 파일에 대한 비밀번호 복구 연구 동향

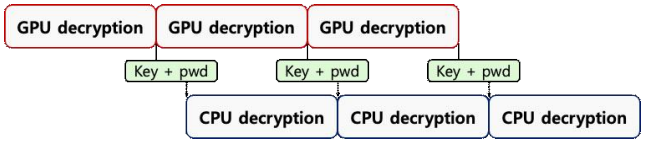
본 장에서는 압축 파일, PDF 및 Excel 문서를 대상으로 암호화된 파일에 대한 비밀번호 복구 연구 동향에 대해 살펴본다.

3.1 WinRAR3(RAR)

WinRAR은 압축 파일을 생성하고 관리하는 유틸리티 소프트웨어이다. WinRAR은 압축 시에 RAR와 ZIP 파일 포맷을 지원하며, 압축을 풀 때에는 TAR, 7z를 포함하여 12개의 파일 포맷을 지원하고 있다. 2020년에 발표된 Qingbing Ji et al. [7]은 WinRAR 버전 3에서 파일 이름이 암호화되지 않은, 기본 암호화 모드를 대상으로 연구를 진행했다. WinRAR3의 암호화 방법으로는 AES 암호화 알고리즘 및 SHA-1 해시 함수가 사용되었다. 해당 논문의 저자들은 <표 1>과 같은 환경에서 CPU와 GPU의 장점을 활용하여 압축 파일의 비밀번호를 복구하는 것을 효과적으로 수행하기 위한 파이프라인 기법을 제안했다. GPU에서 병렬화를 시도하여 AES 암호화 알고리즘 및 SHA-1 알고리즘을 계산하도록 함으로써 연산 속도를 높였으며, 압축을 해제하기 위해 압축 알고리즘을 실행하는 부분은 CPU에서 수행하도록 했다. 또한 CPU와 GPU를 함께 사용할 때 발생하는 대기 시간을 줄이기 위해 (그림 1)의 파이프라인 모드를 설계하였다. 결과적으로 <표 2>와 같이 1K, 10M, 100M의 크기를 갖는 압축 파일에서 8자리 비밀번호에 대해 기존 속도보다 2배 이상 빠른 성능 향상을 보였다.

<표 1> 실험 환경

CPU	Xeon(R)E5-2620
GPU	NVIDIA 1080Ti
CUDA	10.2
operating system	Linux CentOS7



(그림 1) GPU와 CPU를 이용한 파이프라인 모드

<표 2> WinRAR3 최적화 전 후 속도 비교

Size of compressed file	Speed before optimization	Speed after optimization
1K	10981/s	24423/s
10M	9738/s	22423/s
100M	6235/s	16423/s

3.2 PDF(version 1.4-1.6)

2023년에 발표된 Kim et al. [8]은 PDF 문서 1.4-1.6 버전의 암호 해독 알고리즘을 CUDA GPU 상에서 최적화 구현하였다. 해당 버전의 PDF 암호화 알고리즘에서 반복적으로 사용되는 MD5와 RC4의 최적화를 중심으로 연구를 진행했다. MD5 알고리즘에서는 변경되지 않는 일부 메시지 워드의 덧셈 연산을 제거했으며, RC4는 32비트 워드를 통합하여 8비트 워드로의 변환 과정을 없애, 덧셈 및 XOR 연산 횟수가 줄어듦으로써 알고리즘의 연산 속도가 높아졌다. 이에 더해 Autotune 기법을 사용하여 블록당 스레드 수, 그리드 당 블록 수의 값을 탐색하였다. 결과적으로 <표 4>와 같이 RTX 3060과 RTX 3090이라는 동일한 환경에서 hashcat은 각각 25,693 kp/s, 57,601 kp/s 처리량을 보인데 반해, 해당 기법을 통해 최적화 구현된 알고리즘은 31,460 kp/s, 66,351 kp/s의 처리량을 달성하였다. 따라서 22.5%, 15.2%만큼 알고리즘 성능이 향상되었다고 볼 수 있다.

<표 3> 실험 환경

GPU	GeForce RTX 3060
	GeForce RTX 3090

<표 4> 초당 계산 횟수 비교

Reference	Environment	Speed
hashcat 6.2.5	RTX 3060	25,693 kp/s
[7]	RTX 3060	31,460 kp/s
hashcat 6.2.5	RTX 3090	57,601 kp/s
[7]	RTX 3090	66,351 kp/s

3.3 Microsoft Excel(version 2003)

2020년에 발표된 Zhang et al. [9]은 Microsoft의 Excel 문서에 대해 암호화 중간 키(The intermediate key)를 복구하여 암호화된 문서 자체를 복호화 하는 방법을 제시했다. 본 연구는 암호화된 파일의 정보에 접근하기 위해 비밀번호를 크래킹(Cracking) 하는 것이 아니라, 레인보우 테이블 공격을 이용하여 암호화 중간 키를 복구하고, 해당 키로 문서를 복호화하여 정보에 접근하는 방식으로 진행되었다. Excel 문서 2003년 버전까지의 기본 암호화 방식은 RC4로 이루어져 있다. 따라서 해당 논문의 저자들은 Excel 문서의 데이터 저장 구조와 RC4 알고리즘을 함께 분석했다. 이후 분석한 것을 토대로 특정 위치에서 고정된 평문을 활용한 레인보우 테이블을 통해 중간 키스트림을 복구하고, 이를 사용하여 암호화된 데이터 블록을 순차적으로 복호화 한다. 결과적으로 3분 이내라는 일정한 시간 동안 비밀번호의 길이나 복잡성에 영향을 받지 않고 Excel 문서를 복호화 했다. 이는 암호화된 파일의 정보에 접근하기 위해 비밀번호를 직접 복구하는 방법 외에 파일 자체를 복호화 할 수 있는 방법이 있다는 가능성을 보여주었다.

<표 5> 실험 환경

CPU	Intel core i5-8265U @1.6GHz 1.80GHz
RAM	8GB
Operating system	Windows 10, 64 bit.

4. 결론

암호화된 파일에 접근하기 위해 기존 암호화 알고리즘을 분석하여 최적화 혹은 파일 자체를 복호화하는 방식을 사용하거나, CPU나 GPU의 아키텍처를 활용하여 비밀번호 복구 성능을 개선하는 연구들이 수행되었다. 현대 컴퓨팅 환경에서 무차별 대입 공격을 위한 단순 영문자 및 숫자 조합의 수가 11자리를 넘어가면, 5천억 개 이상의 경우의 수가 필요하다. 여기에 특수문자까지 섞인다면 다항 시간 내에 비밀번호를 찾아낼 수 없을 것이다. 암호화된 파일에 대한 비밀번호 복구 연구가 오랜 시간 지속되고 있음에도, 복잡한 비밀번호에 대한 복구를 빠른 시간 내에 해결할 수 있는 방법이 부족하다. 이러한 한계점을 극복하기 위해서는 향상된 성능의 프로세서를 사용하거나 파일 암호화에 사용되는 알고리즘에 대한 고속 구현 등의 추후 연구가 더 필요할 것으로 생각된다. 또한 국내의 경우에는 .hwp 및 .hwpx 확장자를 갖는 한글 파일의 사용률이 높은데, 한글 파일의 암호화 알고리즘이 공개되어 있지 않아서 비밀번호 복구 방안에 대해 활발히 논의되지 않고 있다. 한글 파일은 국내의 기업 및 개인 사용자가 많은 만큼 이에 대한 비밀번호 복구 연구 또한 추가적으로 이루어져야 할 것으로 보인다.

5. Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2018-0-00264, Research on Blockchain Security Technology for IoT Services, 50%) and this work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00627, Development of Lightweight BIoT technology for Highly Constrained Devices, 50%).

참고문헌

[1] Scott Contini, Yiqun Lisa Yin, “Forgery and partial key-recovery attacks on HMAC and NMAC using hash collisions.”, In International Conference on the Theory and Application of Cryptology and Information Security, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 37-53.

[2] Narayanan, Arvind, and Vitaly Shmatikov. “Fast dictionary attacks on passwords using time-space tradeoff.”, In: Proceedings of the 12th ACM conference on Computer and communications security, p. 364-372, 2005.

[3] A. Zonenberg, “Distributed Hash Cracker: A Cross-Platform GPU-Accelerated Password Recovery System.”, Rensselaer Polytechnic Institute, 27, pp.395-399, 2009.

[4] PH Phong, PD Dung, DN Tan, NH Duc and NT Thuy, “Password recovery for encrypted ZIP archives using GPUs.”, Proceedings of the 2010 Symposium on Information and Communication Technology. ACM, 2010. pp.28-33.

[5] Hashcat, hashcat: World's fastest and most advanced password recovery utility, 2022, [Internet], Available: https://hashcat.net/hashcat/

[6] Openwall, John the Ripper password cracker, 2024, [Internet], https://www.openwall.com/john/

[7] Qingbing Ji, Hao Yin, “Speedup and Password Recovery for Encrypted WinRAR3 without Encrypting Filename on GPUs.”, Journal of Physics: Conference Series, Vol.1673, 012047, 2020.

[8] Hyun Jun Kim, Si Woo Eum, Hwa Jeong Seo, “PDF Version 1.4-1.6 Password Cracking in CUDA GPU Environment.”, KIPS Trans. Comp. and Comm. Sys, Vol.12, No.2, pp.69-76, 2023.

[9] Zhang, Lijun, Cheng Tan, and Fei Yu, “Fast Decryption of Excel Document Encrypted by RC4 Algorithm.”, 2020 IEEE 20th International Conference on Communication Technology (ICCT), IEEE, pp.1572-1576, 2020.