ISSN(Print) : 1976-5304, ISSN(Online) : 2713-931X
https://kdfs.jams.or.kr, DOI : 10.22798/kdfs.2022.16.1.99

메모리 분석 기반의 암호화폐 트랜잭션 정보 추출 방안 연구

임민택*, 강정윤**, 박준성***, 이문규****, 정현덕*****, 서정택******
건국대학교 글로컬캠퍼스 경찰학과(학부생)*, 서울여자대학교 정보보호학과(학부생)**, 동신대학교 정보보호학과(학부생)***, 순천향대학교 정보보호학과(학부생)****, 영산대학교 사이버보안학과(학부생)*****, 가천대학교 컴퓨터공학과(교수)******

A Study on the Extraction of Cryptocurrency Transaction Information Based on Memory Analysis

MinTaek Lim*, JeongYoon Kang, JunSung Park***, MoonGyu Lee****, HyeonDeok Jeong*****, JungTaek Seo******

Dept. of Police Science, Konkuk University Glocal Campus*, Dept. of Information Security, Seoul Women's University, Dept. of Information Security, Dongshin University***, Dept. of Information Security Engineering, Soonchunhyang University****, Dept. of Cyber Security, Youngsan University*****, Dept. of Computer Engineering, Gachon Universuty******

요 약

암호화폐는 P2P 네트워크에서 암호화 알고리즘에 따라 채굴되며, 채굴자를 포함한 네트워크 참가자들은 거래소를 통하여 현금화하기도 한다. 익명성을 보장하는 암호화폐의 특성상 범죄 자금의 은닉 및 세탁에 자주 사용되며 특히 암호화폐를 직접 점유 및 소유할 수 있는 하드웨어 지갑의 사용률이 증가하고 있다. 하지만 이러한 하드웨어 지갑 사용을 분석하고 추적하기 위한 포렌식 도구가 없기 때문에 암호화폐와 관련된 수사에 어려움이 있다. 따라서 본 논문은 하드웨어 지갑 사용의 추적 및 분석을 위해 디스크와 메모리에 기록된 암호화폐 거래 관련 아티팩트를 분석하고 이를 추출하는 포렌식 도구를 개발 및 제안한다. 추출된 암호화폐 거래 행위를 분석하는 것은 디지털 포렌식적으로 유의미한 자료가 될 것이며, 이러한 하드웨어 지갑의 아티팩트들은 암호화폐 추적 관련 수사에 활용될 수 있을 것이다.

주제어 : 암호화폐, 메모리 포렌식, 디스크 포렌식, 하드웨어 지갑

ABSTRACT

Cryptocurrency is mined according to a cryptographic algorithm in a P2P network, and network participants, including miners, sometimes cash out through exchanges. Due to the peculiarity of cryptocurrencies that guarantee anonymity, they are often used for hiding and laundering criminal funds, and in particular, the usage rate of hardware wallets that can directly occupy and own cryptocurrencies is increasing. However, there are no forensic tools to analyze and track the use of these hardware wallets, it is difficult to investigate cryptocurrency. Therefore, in this paper, we develop and propose a forensic tool that analyzes and extracts artifacts related to cryptocurrency transactions recorded on disk and memory for tracking and analysis of hardware wallet usage. Analyzing the extracted cryptocurrency transaction behavior will be a meaningful digital forensic data, and these hardware wallet artifacts can be utilized for investigations related to cryptocurrency tracking.

Key Words : Cryptocurrency, Disk Forensics, Memory Forensics, Hardware Wallets

※ 본 연구는 한국정보기술연구원이 주관하는 차세대 보안 리더 양성프로그램 사업의 지원을 받아 수행된 연구임
- Received 17 February 2022, Revised 18, February 2022, Accepted 04 March 2022
- 제1저자(First Author) : MinTaek Lim (Email : alsxor101@gmail.com)
- 교신저자(Corresponding Author) : JungTaek Seo (Email : seojt@gachon.ac.kr)

Ⅰ. 서 론

비트코인과 이더리움은 블록체인 기술을 이용한 가상의 암호화폐이다. 암호화폐는 P2P(Peer-to-Peer) 네트워크로 운영되며, 중앙기관의 통제와 관리 없이 개인 간의 거래가 가능하다는 특징을 가진다. 암호화폐는 직접 암호화폐 채굴을 수행하거나 암호화폐 거래소를 통해 구매하는 방법을 통해 소유할 수 있다. 우리나라는 대량 채굴이 불가능한 환경을 가지고 있어 대부분 거래소를 통해 암호화폐를 구매한다.

암호화폐 기술이 사람들의 관심을 받은 것은 익명성이 보장된다는 점이었지만 우리나라의 경우, 암호화폐 관련 법인 '특정 금융거래정보의 보고 및 이용 등에 관한 법률'로 인해 실명인증 후 거래소를 통해야 암호화폐 구매가 가능하다. 범죄자가 거래소에 실명인증 후 범죄에 사용된 암호화폐를 현금화할 경우, 수사기관이 거래소에 압수수색검증영장을 집행하여 추적할 수 있게 된 것이다. 이와 같은 이유로 암호화폐 거래에서 보안성과 익명성을 제공하는 하드웨어 지갑 사용률이 증가하고 있다[1].

하드웨어 지갑은 USB 형태의 암호화폐 지갑으로 고유의 지갑 주소를 가진다. 지갑 주소를 통해 암호화폐를 거래하면서 생기는 정보를 트랜잭션이라고 하며, PC에 남는 트랜잭션 관련 흔적은 크게 하드웨어 지갑과 관련한 것과 암호화폐 거래와 관련된 아티팩트로 구분할 수 있다. 그중 하드웨어 지갑과 관련한 아티팩트에는 지갑 주소, 니모닉 코드, PIN 코드, 지갑 연결 흔적 등이 해당되며, 암호화폐 거래 관련 아티팩트에는 암호화폐의 종류, 송수신 주소, TXID 등이 있다.

거래소에 보관된 암호화폐의 경우, 이용자는 이를 출금할 권리인 채권을 가지고 있다. 반면에 하드웨어 지갑은 자신이 소유한 암호화폐를 USB 형태의 지갑에 보관할 수 있는 특징이 있다. 이러한 하드웨어 지갑을 네트워크에 연결하기 위해 하드웨어 지갑 제조사들은 사용자를 위한 하드웨어 지갑 전용 PC 애플리케이션을 제공하고, 사용자는 애플리케이션을 통해 거래소를 이용하지 않고 암호화폐 거래가 가능하다.

또한, 하드웨어 지갑과 PC 애플리케이션은 한 암호화폐를 다른 암호화폐로 교환하는 Swap(스왑) 기능을 제공한다. 이 기능을 사용할 경우, 송신자가 스왑을 담당하는 거래소에 암호화폐를 송신하고 거래소 내부에서 스왑이 이루어진다. 따라서 암호화폐를 송수신한 TXID 이외에 교환 과정에서 발생한 정보는 이용자의 PC에 남지 않아, 거래소의 협조를 통하지 않고는 암호화폐 추적이 어렵다. 마찬가지로 Chainalysis 社의 'Reactor' 등 암호화폐 추적 솔루션을 사용하기 위해서는 TXID 혹은 지갑주소를 사전에 알고 있어야 검색 가능한데, 수사 대상자가 묵비권을 행사하여 진술을 거부하고 암호화폐 주소를 제공하지 않을 경우, 지갑 관련 정보를 얻을 수 없어 솔루션 사용이 불가능할 수 있다.

따라서 본 논문에서는 하드웨어 지갑 사용 시 메모리 파일 상에 기록되는 하드웨어 지갑의 정보를 추출하여 암호화폐 추적 및 수사에 도움 될 수 있도록 연구하였다. 하드웨어 지갑을 사용할 경우, 메모리 파일에는 TXID, 지갑주소, 송수신 암호화폐의 종류, 암호화폐의 잔액, 추가적으로 Swap 기능을 이용할 경우, Swap ID 등의 정보가 기록되는데, 이러한 정보를 추출하여 암호화폐 거래 추적에 대한 효율적인 방안과 도구 개발을 제안한다.

연구를 위해 Ledger Nano S(이하 Ledger)와 Trezor One(이하 Trezor)의 두 가지 하드웨어 지갑을 분석하였고 추가적으로 소프트웨어 지갑인 Exodus를 분석해 유사한 점을 확인했다. 트랜잭션을 발생시키기 위해 송수신한 암호화폐는 비트코인(BTC), 이더리움(ETH), 리플(XRP)로 선정하였는데, 비트코인과 이더리움는 전세계적으로 제일 많이 거래되는 암호화폐로 전세계 거래량의 65%를 차지하며[2], 현금화할 경우 가장 높은 가치를 가지고 있어 범죄자금의 은닉에 많이 사용되기 때문이다[3].

본 논문은 다음과 같은 형식으로 구성되어 있다. 2장의 관련 연구에서는 본 논문과 관련된 연구를 설명하고, 3장에서는 본 논문에서 필요로 하는 하드웨어 지갑과 암호화폐의 배경지식에 관해 설명한다. 4장은 하드웨어와 암호화폐 관련 데이터를 수집하여 분석하는 과정을 설명한다. 5장에서는 분석 내용을 바탕으로 개발한 포렌식 도구를 소개하고, 6장에서 종합적인 결론을 맺는다.

Ⅱ. 관련 연구

장세진 등[4]은 불법거래로 의심되는 비트코인 거래를 추적하고 모니터링하는 도구(TxMon)를 개발 및 제안하였다. 이를 통해 범죄혐의가 있는 비트코인 주소 리스트를 기반으로 해당 주소와 거래한 모든 비트코인 주소 중에서 범죄와 관련성이 높은 비트코인 주소를 추정하는 방법을 제시하였다. 해당 연구에서 TxMon을 통한 불법거래 추적의 가능성을 높이기 위해서 충분한 Named List를 확보하고 불법 거래의 가능성이 높은 비트코인을 Unnamed List로 지속적으로 추가하는 방법을 제안하고 있다. 본 연구에서는 하드웨어 지갑을 사용한

기록이 있는 메모리 덤프 파일을 비롯한 PC 아티팩트를 통해서 불법 자금이 있는 코인 주소, 트랜잭션 추출, 지갑 복구를 목표로 연구를 진행하였다.

신혜영 등[5]은 비트코인 트랜잭션 데이터를 분석하여 트랜잭션의 출력 값을 이용한 거래를 추적하고 분석하는 방법을 제시하였다. 이를 통해 동일한 지갑에서 관리되는 지갑 주소와 거래에서 사용된 비트코인을 관리하는 지갑 주소를 추정하고 트랜잭션 출력 값이 사용된 거래와 해당 거래에서 사용된 지갑 주소 데이터를 수집에 대한 가능성을 제시하였다. 이미 해당 연구에서 언급한 트랜잭션 등을 통해 거래를 추적하는 솔루션이 상용화되어 있으며, 본 연구에서는 트랜잭션을 검증하기 위해 Chainalysis 社의 'Reactor'를 이용하였다.

Stephan Zollner 등[6]은 메모리 덤프 파일과 pagefiles.sys, hiberfil.sys 파일에서 트랜잭션 흔적이 존재할 수 있다는 가능성을 제시하였으며, 브라우저 아티팩트의 경우에는 로그인 정보도 찾을 수 있을 뿐 아니라 일부 브라우저에서는 PIN코드와 니모닉 코드를 추출할 수도 있지만, 오탐 여부를 확인하기 쉽지 않은데, 특히 메모리에 남아있는 니모닉 코드 추출의 중요성에 대해서 언급하고 있다. 메모리에 남아있는 니모닉 코드를 추출하는데 성공한다면, 대상 지갑을 온전히 복구할 수 있다는 점에서 니모닉 코드 추출은 중요한 의미를 지닌다. 해당 연구를 통해, 니모닉 코드 추출의 중요성과 필요성을 다시 한번 확인할 수 있었다. 때문에 니모닉 코드 추출부분은 본 연구에서 중요한 의미를 지닌다. 본 연구에서는 메모리, 디스크 포렌식을 통하여 니모닉 코드의 흔적을 찾는 과정을 설명한다.

Tyler Thomas 등[7]은 하드웨어 암호화폐 지갑의 메모리 포렌식 플러그인의 종류로 Memory FORESHADOW를 소개하였다. 해당 연구는 Ledger와 Trezor 두 하드웨어 지갑을 대상으로 진행되었고 Volatility 2버전을 이용해 Windows7 64bit 환경에서 진행되었다. 하드웨어 지갑의 메모리 덤프 결과, JSON 형식으로 거래 내용이 저장되어 있으며, 해당 플러그인은 임의의 많은 데이터를 읽고 JSON 구조가 종료되는 위치를 찾기 위한 YARA 스캔을 수행한다. 본 연구도 하드웨어 지갑(Ledger, Trezor)을 이용한 메모리 덤프파일을 대상으로 진행되었고 Windows10 64bit, Volatility 3버전을 사용했다는 점에서 차이가 있다. 또한, 정규식을 통하여 암호화폐 지갑 주소, 트랜잭션을 구별하고 API를 이용하여 추가 검증한 연구이다.

Michael Doran[8]의 연구에서는 소프트웨어 지갑의 디지털포렌식 아티팩트 대다수가 "wallet.dat"파일에 위치한다는 점과 소프트웨어의 지갑의 "red.dat"파일에는 "undo 데이터"가 포함되어 사용자는 블록을 체인 상태에 대한 패치로 보고 실행 취소 데이터를 역 패치로 볼 수 있음을 제시하였다. Bitcoin 지갑 주소, 트랜잭션, Bitcoin 애플리케이션과 일치하는 다수의 결과를 추출할 수 있다. 해당 연구의 경우, USB 메모리를 이용한 분석이라 메모리 분석에 관한 내용이 없었고 레지스트리의 변화에 대한 분석 또한 없어 한정적인 연구이다.

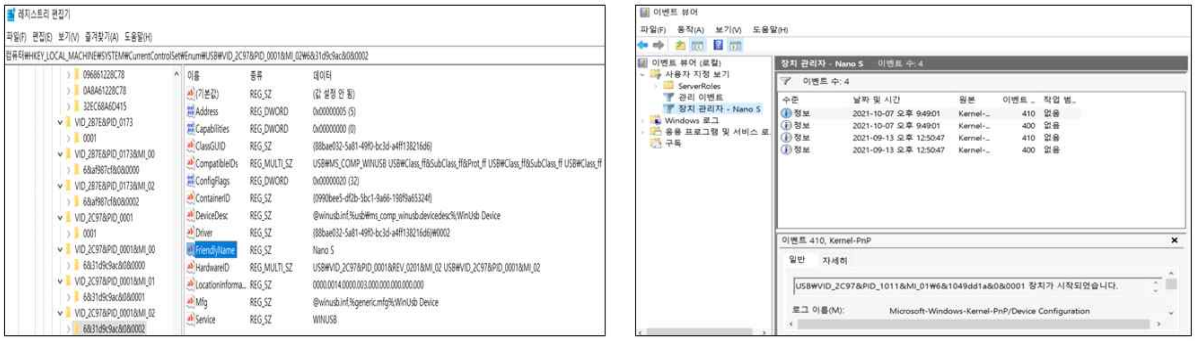
박순태 등[9]은 다크웹과 암호화폐를 이용한 사이버 범죄에 대한 설명과 이에 대응하려는 방법으로 암호화폐 흐름 추적 및 가상자산 취급업소(VASP)를 식별할 수 있는 방법을 제시하였다. 해당 연구팀은 국내 3개 취급업소(VASP)로부터 소수의 알려진 지갑 주소를 확보하면, 이전/이후 발생하는 거래를 이용하여 특정 지갑 주소 발급자(VASP)를 식별할 수 있음을 확인하였다. 그리고 단순한 Heuristic 기법을 사용하여도 일정 수준의 암호화폐 비 익명화에 대한 가능성을 제시하였다.

Ⅲ. 암호화폐 지갑 및 배경지식

3.1 하드웨어 지갑

하드웨어 지갑이란, 암호화폐 거래 시에 개인의 자산을 보다 안전하게 보호할 수 있도록 고안된 수단이다. 하드웨어 지갑은 소프트웨어 지갑 또는 웹 거래소와 달리 항상 인터넷에 연결되어 있지 않은 특성을 가지므로 개인 PC에 관련 애플리케이션을 설치하는 등의 방식으로 함께 사용해야 한다. PC 애플리케이션과 연결을 위해 하드웨어 지갑 내에 암호화폐마다 다른 애플리케이션을 설치해야 지갑 주소가 생성되고, 지갑 주소는 하드웨어 지갑을 완전 초기화하기 전까지 하드웨어 지갑 내 암호화폐 애플리케이션을 지우더라도 재설치를 통해 복구할 수 있다.

하드웨어 지갑은 PC에 연결 시 레지스트리에 연결 흔적이 남아 분석을 통해 하드웨어 지갑 연결 흔적을 확인할 수 있다. 연구를 위해 사용했던 Ledger와 Trezor 모두 HKLM\SYSTEM\ControlSet\Enum\USB 레지스트리 경로에서 연결 흔적을 찾을 수 있었으며, 이벤트 로그에서 하드웨어 지갑 연결 기록을 확인할 수 있었다[그림 1]. 이러한 외부장치 연결 아티팩트는 USB 형태를 가진 하드웨어 지갑이라면 반드시 남는 정보인 것을 알 수 있다.



〈Figure 1〉 Hardware wallet connection identified in registry and event log

3.2 지갑 설정 정보(Mnemonic Code와 PIN code)

니모닉 코드(Mnemonic Code)란 쉽게 기억되는 성질을 가진 단어로, 개인 지갑 생성 시 2048개의 단어 중 무작위로 부여되는 12개 또는 24개 영어단어의 집합이다. 니모닉 코드를 통해 개인 지갑을 온전히 복제 및 복구할 수 있으므로 하드웨어 지갑 제조사들은 종이 등에 적어 지갑의 소유자만 접근할 수 있는 장소에 별도 보관하도록 요구한다. 만약 사용자가 편의를 위해 PC 내 니모닉 코드를 문서 등의 형태의 파일로 저장하고 있다면, 수사 대상자 소유의 하드웨어 지갑을 PIN 코드 인증 없이 복제할 방법이 되기도 한다.

하드웨어 지갑 중 Trezor의 경우, 일반 지갑과 히든 지갑 중 선택하여 생성 가능하다. 일반 지갑은 니모닉 코드 입력을 통한 지갑 복구 및 복제가 가능하지만 히든 지갑은 불가능하다. Ledger의 경우, 히든 지갑 기능을 제공하지 않아 니모닉 코드 입력을 통해 하드웨어 지갑의 복제가 가능했다.

PIN 코드는 비밀번호와 유사하게 본인 인증의 수단으로 쓰이는 4~10자리의 숫자다. 특히 하드웨어 지갑에서의 암호화폐를 거래하거나 관련 설정을 변경할 때 요구된다. Ledger에서 올바르지 않은 PIN 코드를 여러 번 인증할 경우, 하드웨어 지갑의 사용이 불가능하다. 이 경우, 니모닉 코드를 입력하여 복구하거나 지갑을 초기화해야 한다. Trezor에서는 PIN 코드를 잘못 입력할수록 재입력 시간을 연장시키는 방법으로 지갑을 보호하고 있다.

3.3 암호화폐 거래 정보

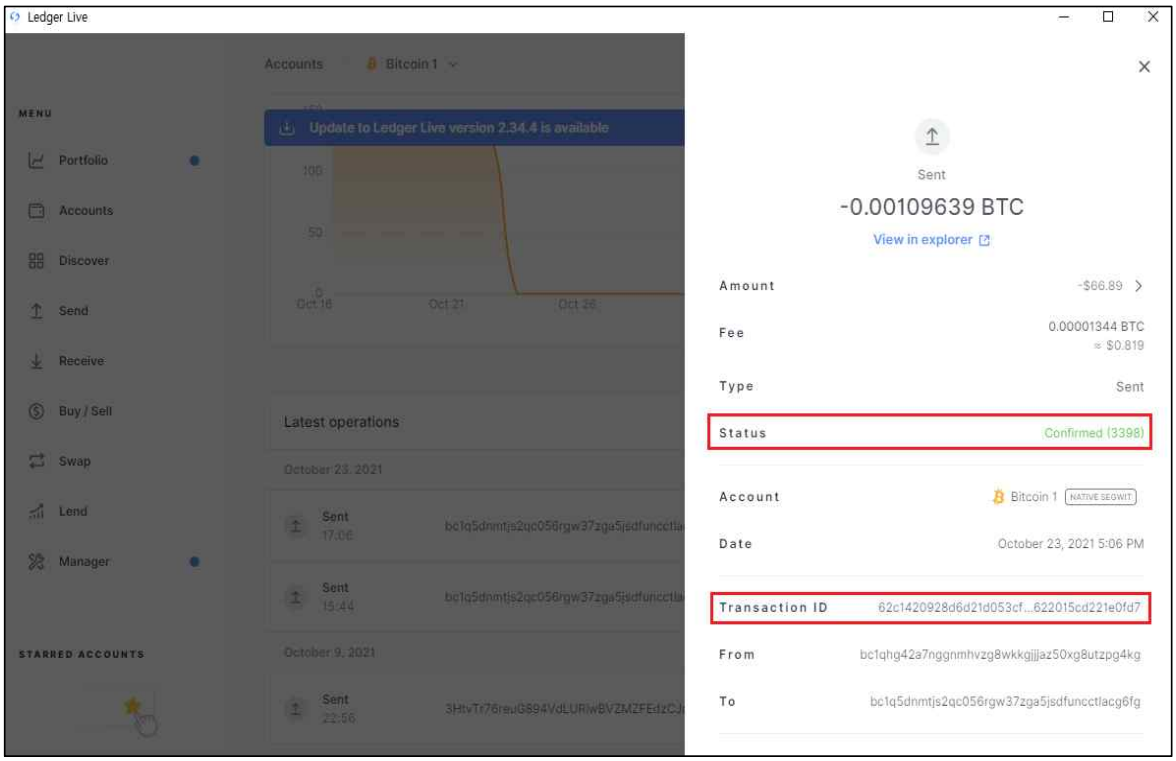
3.3.1 지갑 주소

지갑 주소란 특정 암호화폐에 대한 하드웨어 또는 소프트웨어 지갑 등을 구별하기 위한 값이다. 암호화폐 지갑 주소는 개인키와 공개키 쌍을 생성하여 사용할 수 있다. 또한, 지갑은 트랜잭션을 생성하는 기능이 있다. 하드웨어 지갑은 기기마다 고유의 지갑주소를 가지며, 지갑 내에 암호화폐 종류마다 다른 애플리케이션을 설치해야 지갑주소를 만들 수 있다. USB 형태의 하드웨어 지갑은 제조사 별로 메모리 용량이 다르므로 Ledger는 2~3개 정도의 암호화폐 지갑주소를 만들 수 있고, Trezor는 3-4개 지갑주소를 만들 수 있다. 암호화폐의 지갑주소는 종류마다 다른 형태를 가지고 있다. 비트코인은 Bech32 인코딩에 의해 생성 된 주소이다. 이 주소 유형은 'bc1'로 시작하는 특징이 있으며[10], 리플은 'r', 이더리움은 '0x'로 시작한다. 따라서 각 주소마다 서로 다른 정규식을 사용하여 지갑 주소를 식별할 수 있다.

〈Table 1〉 Wallet address formet and regular expression

암호화폐	지갑 주소 형태	정규표현식
Bitcoin	bc1q9tqzryx7vmv30c0xkfa7vlve4t60js0299qg50	\b(bc(0([ac-hj-np-z02-9]{39}|[ac-hj-np-z02-9]{59})|1[ac-hj-np-z02-9]{8,87})|[13][a-km-zA-HJ-NP-Z1-9]{25,35})\b
Ripple	raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA	r[0-9a-zA-Z]{33,35}
Ethereum	0x8496A6b1805346c3daF697906B898ADf72906Aaf5	0x[a-fA-F0-9]{40}

3.3.2 암호화폐 Transaction



〈Figure 2〉 Transaction history recorded on Ledger Live

트랜잭션이란 통상 데이터베이스의 상태를 변화시키는 작업의 단위를 말하며, 암호화폐에서의 트랜잭션은 한 개인이 소유하는 소유권을 다른 개인으로 이전하기 위한 목적으로 작성된 서명 정보로 정의한다[4]. 이때 TXID는 각 트랜잭션의 식별자로, 64자의 영문 소문자와 숫자의 조합으로 구성된다. TXID 조회를 통해 거래가 잘 이행되었는지 확인 가능하다.

송금하기 위한 수량의 암호화폐와 TXID, 수신자 지갑 주소를 기록한 후 송금을 신청함으로써 트랜잭션을 발생시킬 수 있다. 암호화폐 거래 정보들은 일정 시간마다 블록체인 기반의 블록에 입력되고 네트워크에 기록된다. 이후 블록체인 네트워크가 거래 내용을 확인하고 이전의 블록체인 블록과 연결되면 승인 절차를 걸쳐 암호화폐 거래가 완료된다. Ledger live(Ledger Nano S의 전용 PC 애플리케이션)에서는 거래가 완료될 경우, [그림 2]과 같이 status가 'confirmed'로 변경된다.

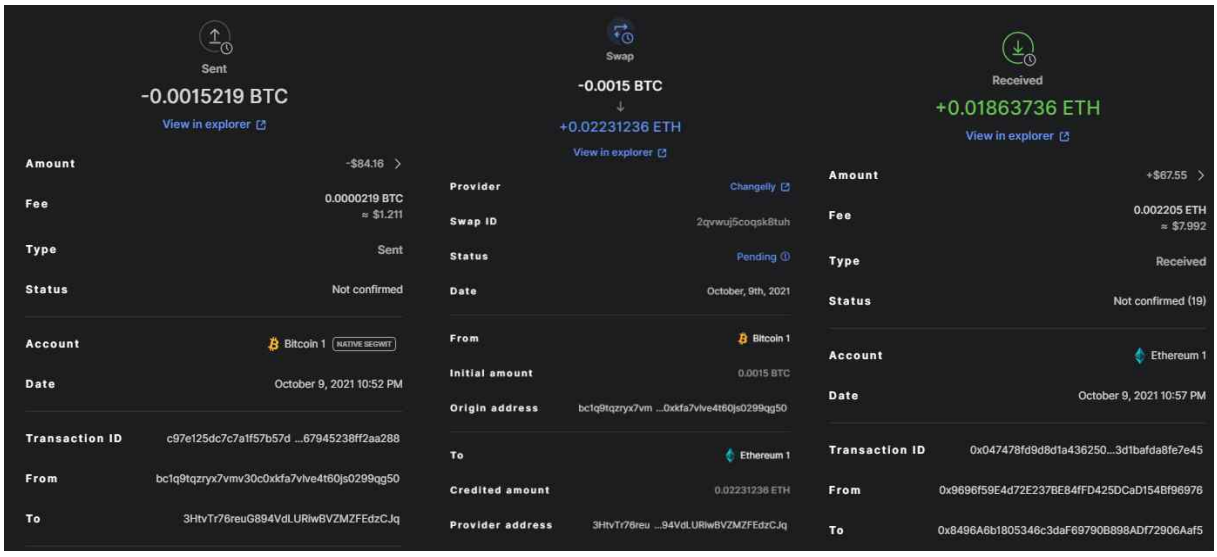
〈Table 2〉 Transaction details recorded in the block

name	Byte	description
Version	4	트랜잭션 버전
input count	압축 크기	입력부 개수
previous output	32	이전 출력부의 TXID
output index	4	이전 출력부 중 잔액을 사용할 출력부 번호
Script length	압축 크기	스크립트 길이
Script sig	Var	Signature 스크립트 전자서명
Sequence	4	Sequence 번호
output count	압축 크기	출력부 개수
value	8	송신할 금액
Script length	압축 크기	스크립트 길이
Script Pubkey	Var	수신자의 공개키 해시값
Lock time	4	500million 이하이면 블록 높이 혹은 Unix timestamp

비트코인의 트랜잭션 발생 시 〔표 2〕와 같은 상세 데이터가 블록에 기록된다. 각 필드에서 숫자로 표시된 것을 고정 크기이며, Var는 가변 크기, 압축 크기는 값의 범위에 따라 크기가 정해진다. JSON 형식의 정보 내용에는 TXID, 송수신 지갑 주소, hash 값, 키 정보 등이 기록되어 있어 메모리에서 정규식을 통해 TXID와 지갑 주소를 분석할 수 있다. 이러한 트랜잭션의 상세 내용은 암호화폐마다 다른 형태이지만, 내용의 대부분이 같은 종류의 데이터를 포함하기 때문에 추출 시 JSON 형식의 문자열을 검색하면 된다.

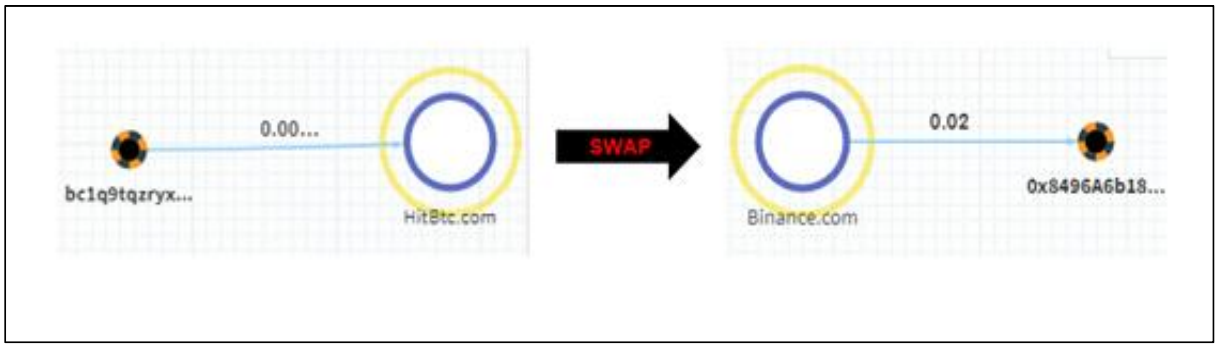
3.3.3 Swap

Swap(스왑)은 하드웨어 제조사와 계약한 암호화폐 거래소에서 특정 암호화폐를 다른 암호화폐로 교환할 수 있는 기능이다. 스왑의 과정은 사용자가 스왑 거래소의 지갑주소에 원하는 암호화폐를 송신하고 거래소 내부에서 금액에 맞는 암호화폐를 교환한 후 사용자가 원하는 지갑주소에 송신하면서 종료된다.



〈Figure 3〉 send, swap, received transaction history

특히 스왑은 암호화폐를 교환하는 과정에서 믹서(mixer)와 같이 출처를 추적하기 어렵게 만들 수 있다. 〔그림 3〕는 Ledger Live에 기록된 스왑 거래 내용이다. 〔그림 3〕의 왼쪽부터 스왑 거래소로 송신한 트랜잭션 기록과 스왑 기록, 수신 기록이다. 하나의 하드웨어 지갑에서 스왑 거래소와 송수신을 모두 진행할 경우, 메모리에 관련 TXID와 지갑주소, 암호화폐 종류 등 많은 정보가 기록되어 모든 과정을 분석할 수 있었다.



〈Figure 4〉 Swap process confirmed by Reactor (BTC -〉 ETH)

하지만 Reactor 검색 결과, 〔그림 4〕와 같이 비트코인을 이더리움으로 스왑하기 위해 HitBtc 거래소로 송신하고 스왑 완료 후 이더리움을 다시 송신한 거래소는 Binance였다. 스왑을 위해 거래소로 송신 후에는 솔루션만을 사용한 추적이 어렵다는 것을 확인할 수 있었다. 따라서 메모리에 남는 트랜잭션 정보와 Reactor 등의 솔루션 결과를 비교해 분석할 필요성이 있다.

Ⅳ. 데이터 수집 및 분석 과정

메모리 파일은 여러 가지 포렌식 소프트웨어를 이용하여 물리적인 램을 분석하는 일련의 과정인 메모리 포렌식을 수행하기 위한 덤프 파일이다. 본 논문에서는 메모리 파일 생성을 위해 커널 레벨에서 동작하는 Belkasoft 社의 라이브 메모리 캡처 도구인 'Belka Live RAM Capturer'을 사용하였다. 본 연구와 도구 개발 환경은 Windows 10 Education 19043.1237 버전이므로 본 논문에서의 메모리 파일이란, 'Belka Live RAM Capturer' 도구를 사용한 Windows 10 19043.1237 버전 환경의 메모리 덤프 파일을 의미한다. 분석을 위해 트랜잭션을 발생시킨 암호화폐는 비트코인(BTC), 이더리움(ETH), 리플(XRP)이다.

〈Table 3〉 Analysis environment

Category	PC Info
OS Version	Windows 10 Pro Education
OS Build	19043.1237
RAM Capacity	4.00GB
SYSTEM	64 Bit

〈Table 4〉 Analysis Tool

Program Name	Program Version
Belka Live RAM Capturer	1.0
Volatility 3	3-1.0.1
HXD	2.5.0.0
Ledger LIVE	2.32.2
Trezor Suite	21.10.2

〈Table 5〉 Bitcoin transactions

TXID	Send	Received	Amount
c97e125dc7c7a1f57b57d0e71150c01deddb3789da83195a67945238ff2aa288	bc1q9tqzryx7vmv30c0xkfa7vlve4t60js0299qg50	bc1qdyvnm7ayrvxu5x4ljefpa6yu7xeq69u9ld7zmh	0.00363581 BTC
9d0fb1247c31fd7d154f030a77458e68034e30d81b5c3609f7b122f0af6f8b10	351yZhnPPbV8Y9C4NBaK1w4GSoRkUzUaU7	bc1qus52ul03xll06yxdyjy5q9gdgdq20jvrgu0zze	20.96877012 BTC
53fa46959eedf0cbe15672b6f19e0f5f76e0e13eb8fd3da410eb038e39485ec7	bc1qus52ul03xll06yxdyjy5q9gdgdq20jvrgu0zze	bc1q9tqzryx7vmv30c0xkfa7vlve4t60js0299qg50	0.00365373 BTC

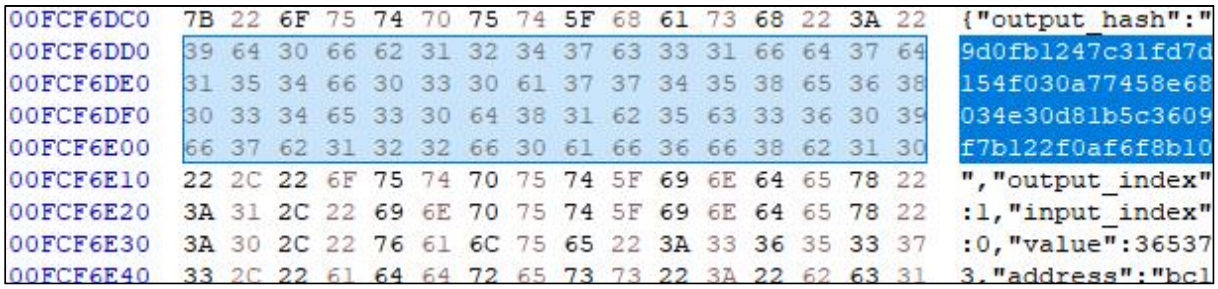
〈Table 6〉 Ethereum(ETH) transactions

TXID	Send	Received	Amount
17c2819183151c3f9150893cbcfc20f2a8bffd46638122d303ca1523a6fa4eae	0x8496a6b1805346c3daf69790b898adf72906aaf5	0x25672b04b810c67112bcf81ce152be35588a26d8	0.01684186 ETH

〈Table 7〉 Ripple (XRP) transaction

TXID	Send	Received	Amount
31A88C6685422785FF6C7CB2A768AEA918D2E9D6BFA9218E438B64E0A1D78A3	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb	raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA	10.00001 XRP
5A86F9D6820264B34F8801FA36C6C45DC72FFBEF02FBFA2EDAA9C33FC10B2AF0	rshRbDTDVUA38vQxax9T7jBC1Bb3H7xQTR	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb	20.66 XRP
ECFA57394ADF5570F836BDFFA47385324BA66FF8BED3EB94D2035F18D7524B33	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb	rshRbDTDVUA38vQxax9T7jBC1Bb3H7xQTR	30 XRP
1F67F10BCB4396D8B905A0F4936E8166F34CECFF4975C3FC290956035C48FC98	rHuULof8mk1m7wffrmsBAVB3g6yAHivbmQ	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb	40.67 XRP

4.1 TXID 추출



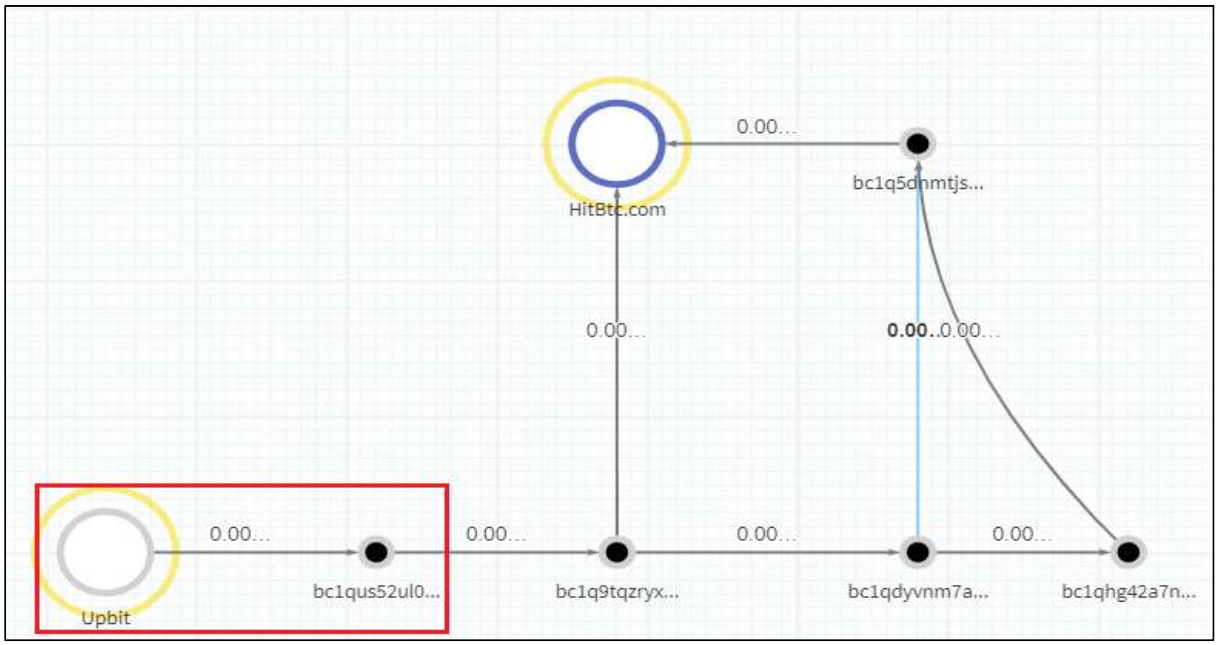
〈Figure 5〉 JSON's TXID extracted from memory

〈Figure 6〉 The full contents of the extracted JSON

〈Table 8〉 Contents of extracted JSON

Inputs	Result
output_hash	9d0fb1247c31fd7d154f030a77458e68034e30d81b5c3609f7b122f0af6f8b10
value	365373
address	bc1qus52ul03xll06yxdyjy5q9gdgdq20jvrgu0zze
Outputs	**Result**
address	bc1q9tqzryx7vmv30c0xkfa7vlve4t60js0299qg50
value	363581

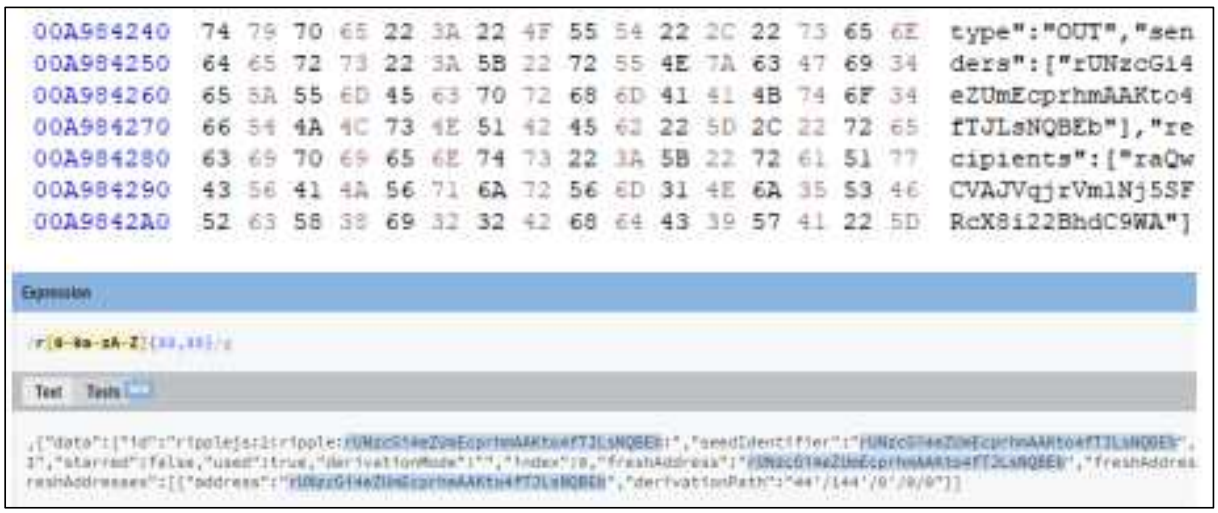
'Belka Live RAM Capturer' 도구로 Ledger 하드웨어 지갑에서 트랜잭션을 발생시킨 후, 메모리를 캡처한 결과를 보면, 〔그림 5〕와 같이 JSON 형식으로 트랜잭션 기록이 메모리에 남는다는 것을 확인하였다. JSON의 내용에는 'id', 'hash', 'received_at', 'fees', 'input data', 'output data', 'block'이 있었고, 'input data', 'output data'에는 각각 송수신 지갑의 상세 데이터가 〔표 2〕와 같은 형태로 기록되어 있었으며, 추출된 데이터는 〔표 5〕와 같았다. 〔그림 6〕의 JSON 전체 내용을 바탕으로 Chainalysis의 Reactor 솔루션을 활용하여 실제 존재하는 데이터인지 검증했다.



〈Figure 7〉 Reactor search results

〔그림 7〕은 해당 트랜잭션에 대한 정보로 송수신 비트코인 주소를 확인할 수 있다. 또한, 암호화폐 송신량을 비롯하여 스왑을 위해 이용한 거래소 주소까지 모두 확인이 가능하였다. 따라서 메모리에 기록된 트랜잭션 정보를 추출할 수 있게 된다면, 추출한 지갑주소에서 거래된 암호화폐의 흐름을 알 수 있다. Ledger와 마찬가지로 Trezor 또한 JSON 형식으로 트랜잭션 기록이 남는 것을 확인하였으며, 부가적으로 분석한 소프트웨어 지갑 Exodus도 JSON 형식으로 메모리에 남는 것을 확인했다.

4.2 지갑 주소 추출



〈Figure 8〉 Wallet address extracted from memory and regular expression

〈Table 9〉 Extracted wallet address

Category	Value
sender	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb
recipients(received)	raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA

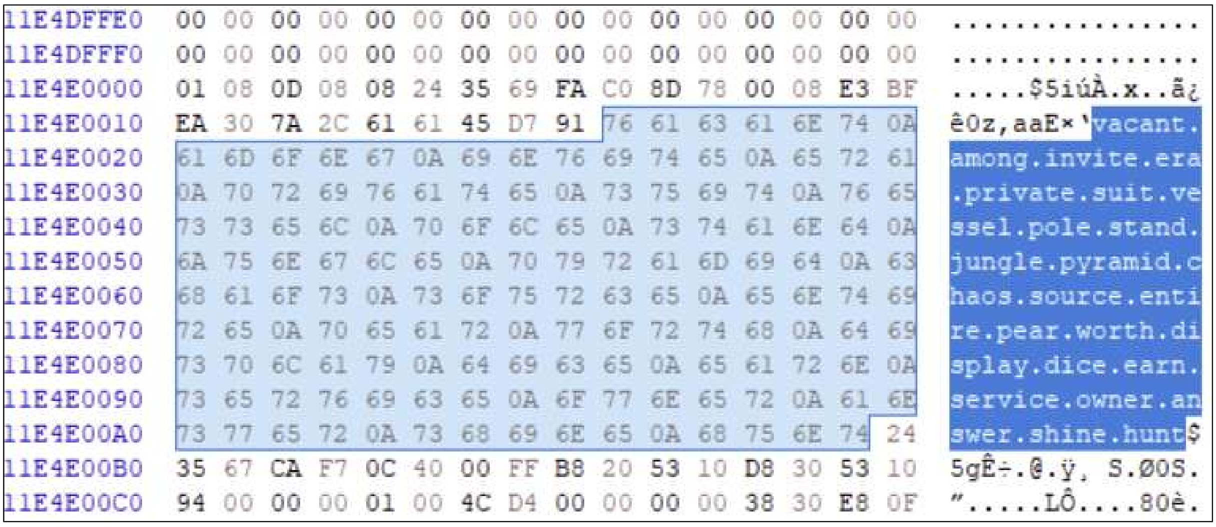
실제 메모리의 물리 위치로 이동해서 검증한 결과, 해당 오프셋에 JSON 형태의 데이터를 확인할 수 있었고

송신한 암호화폐의 종류 및 주소 모두 확인되었다. 메모리 덤프 파일에서는 리플 주소로 식별되는 JSON 형태의 데이터가 존재한다. JSON 데이터의 내용을 확인하면, 송신 주소는 rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb이고 수신 주소는 raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA이다. 이 주소를 〔표 1〕의 정규식을 이용하여 추출하였으며, 〔표 7〕의 실제 거래 기록과 같았다.

리플 이외 비트코인, 이더리움 모두 JSON 형식으로 메모리에 기록되는 것을 확인했으며, 트랜잭션 데이터 추출 결과가 실제 발생시킨 트랜잭션과 같음을 확인했다. 이러한 추출 결과는 암호화가 되어 있지 않은 평문 데이터이기 때문에 추출된 정보를 식별 및 가공하여 활용할 수 있었다.

4.3 니모닉 코드 추출

니모닉 코드를 통해 지갑 복구를 진행했다면, PC 메모리에는 해당 지갑의 24개의 니모닉 코드 전부가 남게 되는 것을 〔그림 9〕와 같이 확인할 수 있다. 니모닉 코드를 알게 되면, 해당 지갑은 온전히 복구할 수 있지만, 니모닉 코드를 메모리에서 추출하는 것은 하드웨어 지갑 사용자가 니모닉 코드로 지갑을 복구 및 초기화한 경우의 제한적인 상황에서 진행이 가능하다. 따라서 메모리에 니모닉 코드가 남아있지 않을 경우를 대비해 PC내에 파일로 저장 되어있을지도 모르는 니모닉 코드를 찾는 것 또한 배제할 수 없다.



〈Figure 9〉 Ledger's mnemonic code extracted from memory

〈Table 10〉 regular expression of the mnemonic code

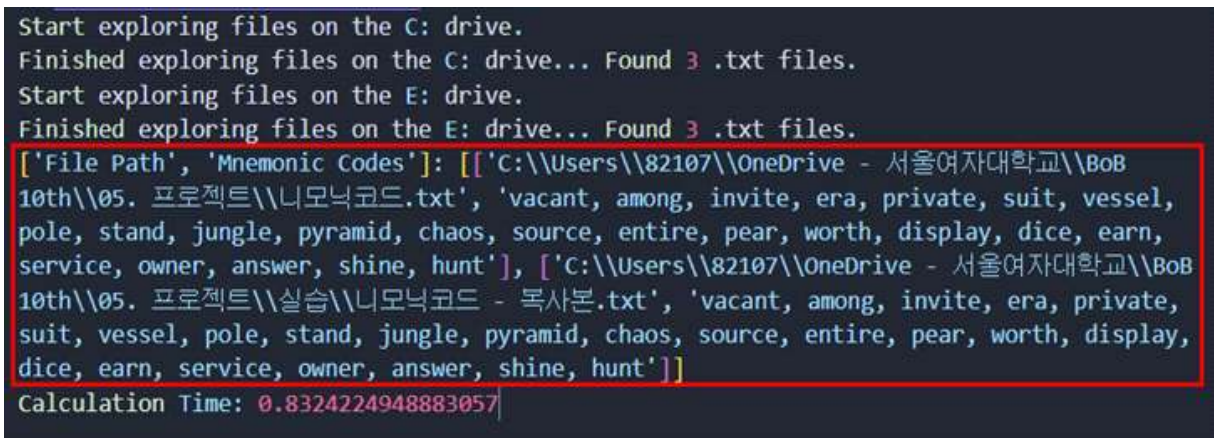
Category	Value
Mnemonic Code Regular Expression	exr = re.compile('[\\\\n]?[a-zA-Z]{3,8}\\\\n[a-zA-Z]{3,8}')

본 연구에서 직접 작성한 니모닉코드 추출 스크립트는 드라이브, 메모리 덤프 파일을 대상으로 정규표현식을 사용해 니모닉 코드를 검색한다. 니모닉 코드의 형식을 가진 텍스트가 2개 이상 추출될 수 있으므로, 무작위로

생성되는 니모닉 코드의 특성을 활용해 정확성을 평가 후 니모닉 코드일 확률이 가장 높은 리스트를 출력하는 형식으로 니모닉 코드를 추출한다. 추출된 니모닉 코드는 공개된 2048개의 니모닉 코드 목록과 대조하여 최종 결과물을 출력한다[그림 10].

〈Figure 10〉 Extract the mnemonic code from memory

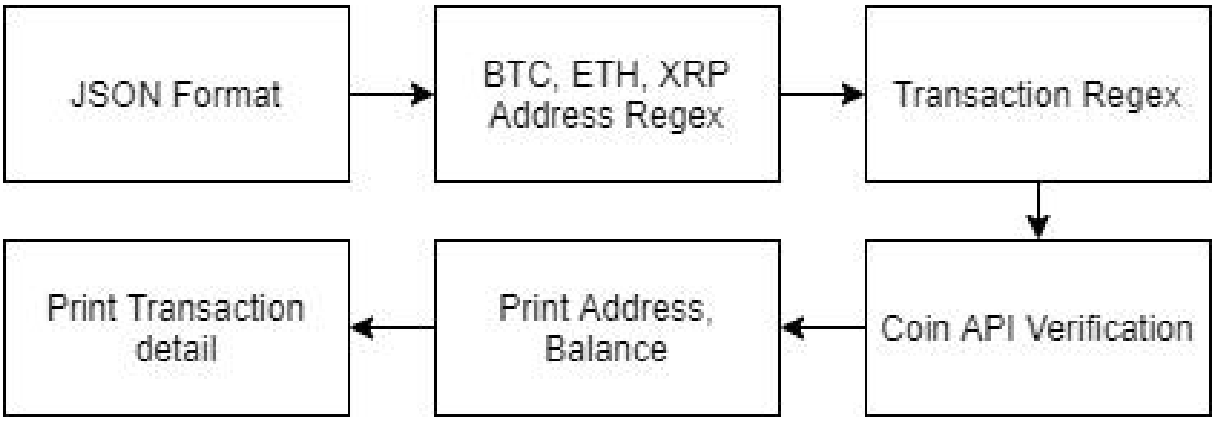
메모리에서 니모닉 코드를 추출하는 것과 마찬가지로 사용자가 니모닉 코드를 txt파일 등의 형태로 PC 내에 저장했을 경우, 디스크 포렌식을 이용해서 "\n"을 기준으로 니모닉 코드 추출이 가능하다. 본 연구에서는 "니모닉코드.txt"안에 니모닉 코드를 텍스트 형태로 저장 후 제작한 스크립트를 통해 추출하였다[그림 11]. 이렇게 추출된 니모닉 코드는 하드웨어 지갑을 복구/복제할 수 있기 때문에 암호화폐 관련 수사에 유용하게 사용할 수 있다.



〈Figure 11〉 Extract the mnemonic code from disk

Ⅴ. 암호화폐 트랜잭션 추출 도구 제안

5.1 개발환경 및 도구의 작업 순서



〈Figure 12〉 CryptoScan diagram

본 논문은 현장 수사의 실효성을 가지기 위해 가장 많이 사용되는 windows 10과 메모리 분석 도구인 Volatility의 최신 버전을 사용하여 도구 제작을 진행했다.

〈Table 11〉 Tool development environment

Category	PC Info
OS Version	Windows 10 Pro Education
OS Build	19043.1237
RAM Capacity	4.00GB
SYSTEM	64 Bit
Language	Python3
Library	logging, re, requests, json, time, datetime
IDE	Visual Studio Code

〔표 11〕의 환경에서 암호화폐 트랜잭션을 발생시키고, 'Belka Live RAM Capturer' 도구로 메모리 덤프를 진행하였다. 메모리 덤프 파일에서 확인한 암호화폐(비트코인, 이더리움, 리플) 송수신 주소를 비롯하여 트랜잭션 정보를 추출하기 위한 수단으로 Windows 10에서 동작하는 Volatility 3-1.0.1 메모리 분석 도구를 활용하였다. 해당 메모리 분석 도구에서 동작하는 플러그인 "CryptoScan"〔11〕을 제작하여 암호화폐 송수신 주소, 트랜잭션 정보를 추출하는 것을 목표로 했다.

5.2 도구 시험 결과 및 우수성

4880	808	smartscreen.ex	0xb788df5ed080	8	-	1	False	2021-10-07 12:24:47.000000	N/A	Disabled
6896	5408	Ledger Live.ex	0xb788d9105080	29	-	1	False	2021-10-07 12:24:47.000000	N/A	Disabled
4696	6896	Ledger Live.ex	0xb788dded5080	11	-	1	False	2021-10-07 12:24:50.000000	N/A	Disabled
2336	6896	Ledger Live.ex	0xb788df45a080	12	-	1	False	2021-10-07 12:24:51.000000	N/A	Disabled
4204	6896	Ledger Live.ex	0xb788deff52c0	16	-	1	False	2021-10-07 12:24:51.000000	N/A	Disabled
7852	2724	audiodg.exe	0xb788dfc19080	5	-	0	False	2021-10-07 12:24:57.000000	N/A	Disabled
2452	6896	Ledger Live.ex	0xb788dfc1e080	15	-	1	False	2021-10-07 12:25:01.000000	N/A	Disabled

〈Figure 13〉 Check the Ledger Live.exe PID through the window.pslist plugin

4436	5268	Trezor Suite.e	0xda0510e6f080	33	-	1	False	2021-10-26 09:34:06.000000	N/A	Disabled
8092	684	svchost.exe	0xda051266e080	10	-	0	False	2021-10-26 09:34:07.000000	N/A	Disabled
1432	4436	Trezor Suite.e	0xda05123d8080	16	-	1	False	2021-10-26 09:34:08.000000	N/A	Disabled
6344	4436	Trezor Suite.e	0xda0511d45080	14	-	1	False	2021-10-26 09:34:08.000000	N/A	Disabled
2716	4436	Trezor Suite.e	0xda05123e10c0	20	-	1	False	2021-10-26 09:34:08.000000	N/A	Disabled
3596	4436	trezord.exe	0xda0512ac4080	12	-	1	False	2021-10-26 09:34:08.000000	N/A	Disabled

〈Figure 14〉 Check the Trezor Suit.exe PID through the window.pslist plugin

Volatility에서 pslist 플러그인을 통해 하드웨어 지갑 클라이언트 프로그램을 식별할 수 있다. CryptoScan 플러그인을 실행하기 전에 〔그림 14〕와 같이 pslist 플러그인을 통해 해당 하드웨어 지갑 PC 애플리케이션의 PID(Process ID)를 확인해야 한다. CryptoScan은 하드웨어 지갑 애플리케이션의 PID 단위로 암호화폐 지갑주소, 트랜잭션 정보를 탐색할 수 있도록 암호화폐 종류마다 옵션을 나누었다. 또한, 탐색하고자 하는 암호화폐의 종류를 옵션으로 지정할 수 있도록 제작하였다.

하드웨어 지갑 PC 애플리케이션 중 Ledger Live와 Trezor Suit의 프로세스를 대상으로 하며, 비트코인(BTC), 이더리움(ETH), 리플(XRP) 세가지 암호화폐의 분석이 가능하다. 연구에 사용된 하드웨어 지갑 이외에도 소프트웨어 지갑 Exodus도 같은 결과를 출력할 수 있었으며, PC 애플리케이션을 사용하는 대부분의 하드웨어 지갑은 유사할 것으로 보인다.

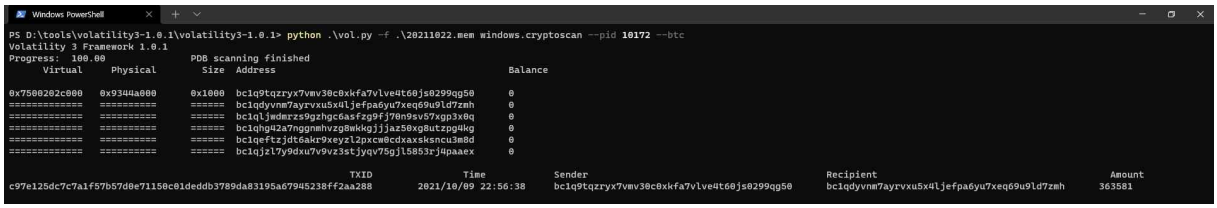
메모리에 송수신 주소와 트랜잭션 정보가 JSON 형태로 기록되기 때문에 해당 플러그인에서는 탐색하고자 하는 영역을 대상으로 정규표현식을 활용하여 JSON 데이터를 탐색한다. JSON 형식의 데이터가 존재한다면, 지갑 주소 정규식과 트랜잭션 정규식을 활용하여 데이터를 추출하고, 추출된 데이터를 주소, 트랜잭션을 검증할 수 있는 API로 전송한다. API 요청 결과, 현재 주소의 잔액, 트랜잭션 발생 시각을 비롯하여 송수신 주소, 보낸 암호화폐의 종류, 양을 모두 확인할 수 있다.


```
{"data":
{"block_height":703947,"block_hash":"00000000000000000001e08671e9862d863e8aa856bd4fa250344feeee892e02","block_time":1633608787,"created_at":1633608797,"confirmations":4499,"fee":8089,"hash":"9d0fb1247c31fd7d154f030a77458e68034e30d81b5c3609f7b122f0af6f8b10","inputs_count":1,"inputs_value":2096877012,"is_coinbase":false,"is_double_spend":false,"is_sw_tx":false,"lock_time":0,"outputs_count":4,"outputs_value":2096868923,"sigops":20,"size":434,"version":2,"vsize":434,"weight":1736,"witness_hash":"9d0fb1247c31fd7d154f030a77458e68034e30d81b5c3609f7b122f0af6f8b10","inputs":[{"prev_addresses":
["35lyZhnPPbV8Y9C4NBaK1w4GSoRkUzUaU7"],"prev_position":0,"prev_tx_hash":"77f63f68b61d7521de029a142e36d4bd9f0ac9b993ccaeb15fa86fb6c8988fd1","prev_type":"P2SH","prev_value":2096877012,"sequence":4294967295}],"outputs":[{"addresses":
["1KYz1276R8yHvPMRCuGz7gZNw89bPf14sC"],"value":1497832,"type":"P2PKH","spent_by_tx":"3d7f1eb57f3fac0ec3b458196355d70a04f736fd6f10a0cb831876720d62f5765","spent_by_tx_position":2},{"addresses":
["bc1qus52ul03xll06yxdyjy5q9gdgdq20jvrguOzze"],"value":365373,"type":"P2WPKH_V0","spent_by_tx":"53fa46959eedf0cbe15672b6f19e0f5f76e0e13eb8fd3da410eb038e39485ec7","spent_by_tx_position":0},{"addresses":
["1CYkuddaQGzs7vKDdtAtSXBgf1fbTGwbdL"],"value":47287010,"type":"P2PKH","spent_by_tx":"35621458cba0638c8edf4a06ecf9502fc6a7a5417ecdcec0af752e2509c0a0e1","spent_by_tx_position":2},{"addresses":
["35lyZhnPPbV8Y9C4NBaK1w4GSoRkUzUaU7"],"value":2047718708,"type":"P2SH","spent_by_tx":"6e72163270b70b8ca33b8d5938c1cd5946fc112325c832538d8d4775e8e6cfff","spent_by_tx_position":0}]},"err_code":0,"err_no":0,"message":"success","status":"success"}
```

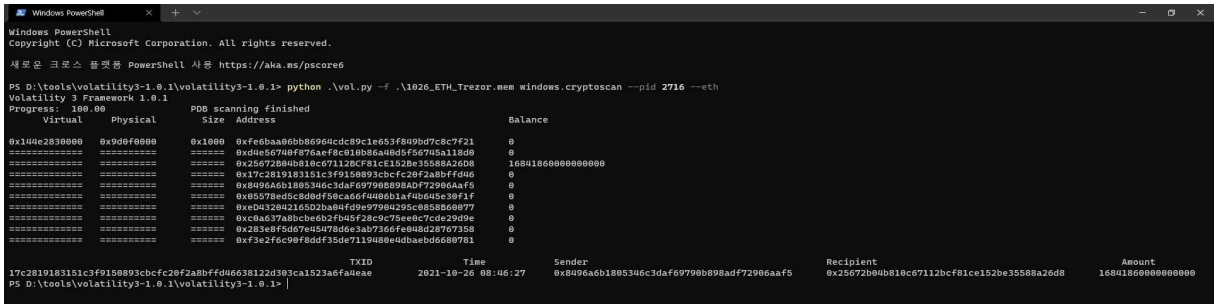

〈Figure 15〉 API request result

〔그림 15〕의 JSON 형태로 출력된 API 결과를 바탕으로 플러그인의 출력 칼럼을 구성하였다. CryptoScan 플러그인 출력 결과, 암호화폐의 송수신 주소가 메모리의 어디에서 발견되었는지에 대한 트랜잭션 상세 정보를 식별할 수 있다. 메모리에 남아있는 트랜잭션 정보를 추출하여 거래소로 보내진 암호화폐를 비롯한 개인 지갑으로 보낸 정보까지 모두 조회함으로써 암호화폐 거래 과련 수사에 도움 될 수 있을 것이다.

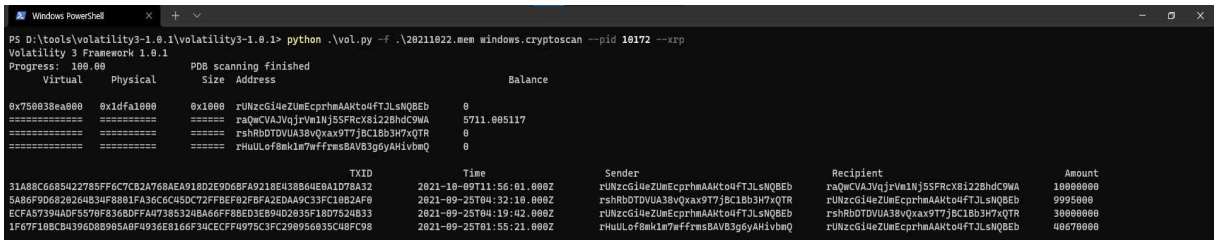
예) python ./vol.py -f 〔memory.mem〕 windows.cryptoscan --pid 〔Process ID〕 --〔btc,eth,xrp〕



〈Figure 16〉 CryptoScan's --btc option output



〈Figure 17〉 CryptoScan's --eth option output



〈Figure 18〉 CryptoScan's --xrp option output

본 연구에서는 최종 결과를 쉽게 확인할 수 있도록 '-pdf' 명령어 사용 시 플러그인 옵션을 통해 PDF로 출력 결과를 내보내는 기능을 제공하고 있다. Volatility 3도 추출한 암호화폐 트랜잭션 관련 정보들을 알아보기 쉽게 하였으며, 보고서 형식으로 출력할 수 있다. 이 PDF export 기능을 통하여 메모리에서 검색된 코인 주소와 트랜잭션 흔적을 문서에서 쉽게 식별할 수 있다. 또한, 각 주소에 남아있는 코인의 양을 통해 범죄자들의 은닉 자금 추적에 활용할 수 있을 것이다.

CryptoScan Report

Report Name: CryptoScan_204343
Report Created: 2021-11-06 20:43:43
Analysis Version: 1.0

Cryptocurrency Transaction Info - Wallet Address

Address	Balance
rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb	0
raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA	5711.005117
rshRbDTDVUA38vQxax9T7jBC1Bb3H7xQTR	0
rHuULof8mk1m7wffrmsBAVB3g6yAHivbmQ	0

Cryptocurrency Transaction Info - Transaction Info

Tag	Value
TXID	31A88C6685422785FF6C7CB2A768AEA918D2E9D6BFA9218E438B64E0A1D78A32
Time	2021-10-09T11:56:01.000Z
Sender	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb
Receiver	raQwCVAJVqjrVm1Nj5SFRcX8i22BhdC9WA
Amount	10000000
TXID	5A86F9D6820264B34F8801FA36C6C45DC72FFBEF02FBFA2EDAA9C33FC10B2AF0
Time	2021-09-25T04:32:10.000Z
Sender	rshRbDTDVUA38vQxax9T7jBC1Bb3H7xQTR
Receiver	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb
Amount	9995000
TXID	ECFA57394ADF5570F836BDFFA47385324BA66FF8BED3EB94D2035F18D7524B33
Time	2021-09-25T04:19:42.000Z
Sender	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb
Receiver	rshRbDTDVUA38vQxax9T7jBC1Bb3H7xQTR
Amount	30000000
TXID	1F67F10BCB4396D8B905A0F4936E8166F34CECFF4975C3FC290956035C48FC98
Time	2021-09-25T01:55:21.000Z
Sender	rHuULof8mk1m7wffrmsBAVB3g6yAHivbmQ
Receiver	rUNzcGi4eZUmEcprhmAAKto4fTJLsNQBEb
Amount	40670000

〈Figure 19〉 PDF EXPORT document capture

Ⅵ. 결론 및 향후 연구방안

본 연구는 하드웨어 지갑(Ledger Nano S, Trezor One)을 이용하여 트랜잭션을 발생시킨 후 메모리 포렌식을 통해 메모리에 기록되는 암호화폐 관련 정보를 확인하는 방법과 이를 추출하는 Python 3 스크립트 및 Volatility 3 플러그인인 'CryptoScan'과 더불어 디스크 포렌식을 이용한 하드웨어 지갑 이용 기록, 니모닉 코드 추출 및 복구 방법을 제시하였다. 하드웨어 지갑을 이용한 PC 메모리 대상 암호화폐 트랜잭션 분석은 기존 연구에서 밝혀내지 못했던 송수신 지갑 주소, TXID, 암호화폐 잔액을 비롯해 니모닉 코드, PIN 코드 등을 확인할 수 있으므로 디지털포렌식 수사 관점에서 중요성을 지닌다. 또한, Windows 7 환경에서 Volatility 2를 사용해 메모리 파일을 분석한 기존의 연구 환경과 다르게, 가장 보편적으로 사용되는 Windows 10 환경에서 Volatility 3을 활용하여 연구 결과의 실용성을 보다 높였다.

메모리 포렌식을 통하여 확인한 JSON 형식의 암호화폐 트랜잭션 관련 정보는 암호화폐 종류별 다른 특징을 가지는 지갑 주소를 정규표현식〔표 1〕을 통해 추출하였으며, 추출된 주소 정보를 비롯한 트랜잭션의 검증은 btc.com〔12〕, etherscan〔13〕, cryptoapis.io〔14〕 등에서 제공되는 API를 통해 이루어졌다. 또한, 지갑을 복구하는 과정에서 메모리에 기록되는 PIN 코드 및 니모닉 코드 역시 정규표현식을 이용한 Python3 스크립트를 이용해 추출했다. 기존의 도구를 활용하여 추출한 값은 메모리 포렌식 도구 'CryptoScan'을 개발하기 위한 자료로 사용했다.

기존 암호화폐 수사에서는 하드웨어 지갑 발견 시에 수사 대상자에게 정보 제공을 요청해야만 하드웨어 지갑을 분석할 수 있었다. 하지만 본 연구에서 개발한 'CryptoScan' 메모리 포렌식 도구를 사용할 경우, 수사 대상자가 하드웨어 지갑의 정보를 제공하지 않더라도 메모리에서 암호화폐 관련 데이터를 추출할 수 있다. 추출 가능한 데이터는 TXID, 지갑주소, 암호화폐 잔액, 니모닉 코드로 암호화폐 수사에 활용할 수 있는 정보이다. 또한, 추출된 데이터는 Chainalysis의 Reactor 등의 사전에 트랜잭션 정보을 파악해야 하는 솔루션에 사용될 수 있으며, 최종 보고서 형식의 PDF 파일로 출력할 수 있다.

향후에는 본 연구를 바탕으로 니모닉 코드로 지갑 복구가 불가했던 Trezor One의 히든 지갑에 관한 추가 연구를 진행하고, 기존에 개발한 txt 파일 대상의 니모닉 코드 추출 스크립트는 HWP, Docx, PDF 등 다양한 문서 프로그램에서 니모닉 코드와 PIN 코드를 탐색할 수 있도록 업데이트할 예정이다. 또한, 본 논문에서 분석한 하드웨어 지갑의 최신 기기들은 모바일 기기와 블루투스 연결 기능을 제공하므로 이후 모바일 포렌식을 통한 암호화폐 트랜잭션 분석으로 연구 분야를 확장할 것이다.

참 고 문 헌 (References)

[1] businesswire, https://www.businesswire.com/news/home/20211029005446/en/Global-Hardware-Wallet-Market-2021-2026-Exchange-Hacks-Driving-Investment-in-Safer-Cold-Storage-Wallets---ResearchAndMarkets.com.

[2] Coinmarketcap, https://www.coinmarketcap.com.

[3] Chainalysis, "The 2020 state of crypto crime", https://go.chainalysis.com/rs/503-FAP-074/images/2020-Crypto-Crime-Report.pdf.

[4] Sejin Jeong, Nohyun Kwak, Brent Byunghoon Kang, "A Study of Bitcoin Transaction Tracking Method through Illegal Community",KAIST, Graduate School of Information Security, *Journal of The Korea Institute of Information Security & Cryptology*, Vol 28, No 3, 2018.

[5] Hye-yeong Shin, "A study on Transaction Tracking and Analysis through Bitcoin Transaction Data", Keimyung University, 2021.

[6] Stephan Zollner, "An Automated Live Forensic and Postmortem Analysis Tool for Bitcoin on Windows Systems", *IEEE Access,* Vol 7, 158250-158263, 2019.

[7] Tyler Thomas, Mathew piscitell, Ilya Shavrorv and Ibrahim Baggili, "Memory FORESHADOW: Memory FOREnSics of HArDware CryptOcurrency wallets - A Tool and Visualization Framework", *Forensic Science International: Digital Investigation*, 33, 2020.

[8] Michael Doran, "A Forensic Look at Bitcoin Cryptocurrency", *SANS Institute*, 2015.

[9] SoonTai Park, Yonghee Shin, HongKoo Kang, "Tracking illegal transaction of cryptocurrencies that are abused for cybercrime", *Communications of the Korean Institute of Information Scientists and Engineers*, 38, 40-47, 2020.

[10] Jin-hee Lee, Min-jae Kim, Junbeom Hur, "Multi-Layer Bitcoin Clustering through Off-Chain Data of Darkwe", *Journal of The Korea Institute of Information Security & Cryptology*, Vol. 31, No. 4, pp. 715-729, 2021.

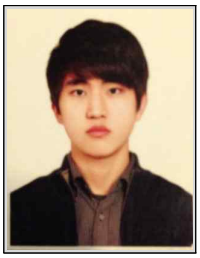
[11] CryptoScan, https://github.com/BoB10th-BTC/plugin.

[12] Btc, https://www.btc.com.

[13] Etherscan, https://www.etherscan.io.

[14] Cryptoapis, https://www.cryptoapis.io.

저 자 소 개



임민택 (MinTaek Lim)
준회원

2014년 3월 ~ 현재 : 건국대학교 글로컬 경찰학과 학사과정
관심분야 : 디지털 포렌식, 수사학 등

강정윤 (JeongYoon Kang)
준회원

2018년 3월 ~ 현재 : 서울여자대학교 정보보호학과 학사과정
관심분야 : 디지털 포렌식, 정보보호 등

박준성 (JunSung Park)
준회원

2020년 3월 ~ 현재 : 동신대학교 정보보호학과 학사과정
관심분야 : 디지털 포렌식, 정보보안

이문규 (MoonGyu Lee)
준회원

2020년 3월 ~ 현재 : 순천향대학교 정보보호학과 학사과정
관심분야 : 디지털 포렌식, 역공학

정현덕 (HyeonDeok Jeong)
준회원

2016년 3월 ~ 현재 : 영산대학교 사이버보안학과 학사과정
관심분야 : 디지털 포렌식, 웹해킹

서정택 (JungTaek Seo)
준회원

1999년 2월 : 한국교통대학교 컴퓨터공학 학사졸업
2001년 2월 : 아주대학교 컴퓨터공학 공학석사
2006년 2월 : 고려대학교 정보보호공학 공학박사
2000년 11월 ~ 2016년 2월 : 국가보안기술연구소 책임연구원/연구부장
2016년 3월 ~ 2021년 2월 : 순천향대학교 정보보호학과 부교수
2021년 3월 ~ 현재 : 가천대학교 컴퓨터공학과 부교수
관심분야 : CPS, 제어시스템 등