비트코인 셀프 커스터디 가이드

사고 사례와 모범 사례에서 도출한 셀프 커스터디 지침

초안 제출일: 2026-07-28
최종 승인일: 2026-09-15

About

메타노미아는 크립토 시장과 기술의 변화를 연구·분석하는 싱크탱크입니다.

Researcher

이지환 Jeehwan Lee

Corresponding Author

오태민 Taemin Oh

Contents

01

두 건의 사고

2026년, 셀프 커스터디의 위험을 서로 다른 방향에서 보여준 사고가 두 건 있었다. 하나는 하드웨어 지갑이 만들어 준 열쇠 자체에 결함이 있었던 사건이고, 다른 하나는 국가 기관이 압류한 지갑의 복구 암호를 스스로 공개해 버린 사건이다. 두 사고가 함께 보여주는 것은 셀프 커스터디에서는 지갑 관리의 모든 책임이 지갑 주인에게 돌아온다는 점이다. 실수로 잃든 제조사 결함으로 잃든 대신 물어 줄 회사가 없고, 개인 셀프 커스터디를 폭넓게 보장하는 보험도 사실상 없다. 그래도 셀프 커스터디를 할 수밖에 없는 사람들이 있고, 이 글은 그런 사람들을 위해 썼다. 셀프 커스터디가 믿을 만한 기관에 맡기는 것보다 꼭 안전하다고는 말할 수 없다. 그래서 이 글은 기관과 국가가 코인 보관 문제를 어떻게 풀고 있는지를 참고해, 개인이 감당할 수 있는 수준의 방어책을 정리했다. 이미 셀프 커스터디를 운용하고 있어 당장 점검할 항목이 필요한 독자는 5장부터 읽어도 좋다.

사용자가 아무것도 잘못하지 않아도 잃을 수 있다

2026년 7월 30일, 콜드카드(Coldcard)라는 하드웨어 지갑을 쓰던 사람들의 비트코인이 한꺼번에 빠져나가기 시작했다. 25분 만에 약 500개 지갑에서 약 594 비트코인(당시 약 3,800만 달러)이 사라졌다. 첫 공격 뒤 세 차례가 더 이어져 총 네 차례의 탈취가 발생했고, 5,200개가 넘는 주소에서 약 1,816 비트코인(당시 시세로 약 1억 1,600만 달러)이 없어졌다.[1]

원인은 사용자의 실수가 아니었다. 2021년 3월 17일에 배포된 펌웨어 4.0.0에서 시드를 무작위로 만드는 과정에 결함이 생겼다. 그 결과 생성될 수 있는 시드의 범위가 설계 목표인 2의 256제곱 가지보다 크게 좁아졌다. 핀테크 기업 블록(Block)의 분석에 따르면 콜드카드의 구형 모델인 Mk2와 Mk3는 시드를 만들 때 암호학적으로 안전한 무작위 값을 추가하지 못했다. 후속 모델인 Mk4·Q·Mk5도 보안 칩에서 얻은 값을 충분히 활용하지 못하고, 그중 32비트만 난수 생성기에 반영했다. 32비트는 약 43억 가지를 구분할 수 있는 크기지만 암호학적으로는 매우 작은 수준이다. 다만 실제 탐색 난도는 기기 식별값, 부팅 시점, 난수 생성기 호출 이력 등 공격자가 알고 있는 정보에 따라 달라졌다. 공격자는 취약한 난수 생성 방식을 자신의 컴퓨터에서 재현해 시드 후보들을 만든 뒤, 여기서 나온 주소를 블록체인에 공개된 지갑 주소와 대조했다. 일치하는 주소를 찾으면 해당 시드로 지갑의 자금을 옮길 수 있었다. 이 결함이 5년 넘게 발견되지 않는 동안 해당 기기로 시드를 만든 사용자는 자신이 위험에 노출됐는지 알기 어려웠다.[2]

FIGURE 01 **콜드카드 펌웨어 결함으로 인한 시드 엔트로피 붕괴**

설계 의도는 256비트였으나, 취약한 난수 생성 과정에서 시드의 탐색 범위는 아래 구간으로 줄었다. 단위는 비트.

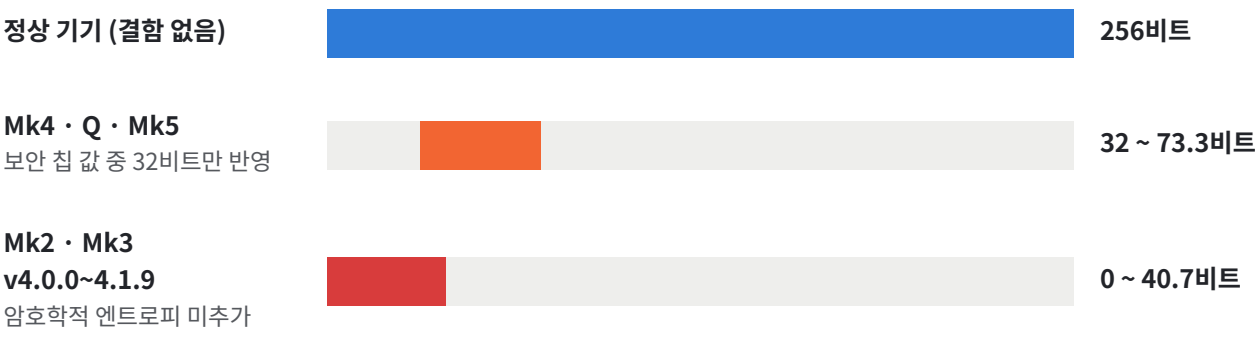


FIGURE 01 **콜드카드 펌웨어 결함으로 인한 시드 엔트로피 붕괴 (계속)**

막대그래프의 좌우 구간에서 왼쪽 끝이 하한, 오른쪽 끝이 상한이다. 하한은 공격자가 기기 내부의 타이머 값까지 알아냈을 때 남는 경우의 수이고, 상한은 그 값을 모를 때 나올 수 있는 보안 강도의 최대값이다. 이 최대값은 기술 분석 보고서가 제시한 값이며, 제조사는 Mk4·Q·Mk5의 엔트로피를 약 72비트로 밝혔다. 2026년 7월 30일 시작된 탈취는 이 결함을 이용했다.

자료: Block 엔지니어링 기술 분석 및 Coinkite 공식 권고

얄궂게도 콜드카드는 오래전부터 보안으로 이름난 브랜드였고, 그 명성을 믿은 수많은 개인이 코인을 잃었다. 콜드카드의 제조사(Coinkite)는 투명성을 내세워 펌웨어 소스코드를 공개해 왔고, 문제의 펌웨어도 2021년 3월부터 줄곧 공개돼 있었다. 그런데 정작 사용자에게는 자기 펌웨어가 취약한지 확인할 도구가 없었다. 공개된 코드는 오히려 해커에게 취약점을 뒤질 재료가 되었다. 이번 공격의 주체가 사람이 아니라 인공지능 언어모델일 수 있다는 이야기까지 나왔다.[3]

니모닉 관리가 느슨하면 기술 없이도 털린다

같은 해 2월 26일, 국세청은 고액 체납자에게서 세금을 받아 낸 성과를 알리는 보도자료를 냈다. 자료에는 압류한 가상자산 지갑의 사진이 실려 있었는데, 그 사진에 지갑을 통째로 복구할 수 있는 24개 단어, 즉 니모닉 코드가 가려지지 않은 채 그대로 찍혀 있었다.[4] 자료가 공개된 지 하루 만에 그 지갑의 PRTG 토큰 400만 개(당시 거래소 표시가격 기준 약 69억 원)가 몽땅 다른 지갑으로 빠져나갔다. 2월 28일 자신이 가져갔다고 밝힌 사람이 전량을 돌려놨지만, 약 2시간 30분 만에 다시 다른 지갑으로 사라졌다.[5]

이 사고에 정교한 해킹 기술은 없었다. 공개된 사진을 읽은 것이 전부다. 비트코인이 아닌 다른 토큰에서 벌어진 일이지만, 니모닉 관리가 얼마나 중요한지는 비트코인이라고 다를 것이 없다. 국가 기관도 이런 실수를 한다. 개인이 같은 실수를 할 가능성은 더더욱 가볍게 볼 일이 아니다.

셀프 커스터디를 해야 하는 사람들, 그리고 그 가치

그럼에도 셀프 커스터디를 택할 수밖에 없는 사람들이 있다. 자산을 누구에게도 맡기지 않고 직접 쥐고 있으려는 사람, 거래소가 요구하는 고객 확인 절차(Know Your Customer, KYC)를 거치기 어려운 사정이 있는 사람, 그리고 본인 지갑으로 직접 서명해야만 쓸 수 있는 탈중앙 금융(Decentralized Finance, DeFi) 서비스를 이용하는 사람이다. 비트코인을 자기 지갑에 둔 채로 참여하는 바빌론(Babylon) 스테이킹이나, 본인 비트코인 지갑으로 거래에 직접 서명해야 하는 오디널스(Ordinals)와 룬(Runes) 거래는 셀프 커스터디 없이는 시작조차 할 수 없다.[6] 규모로 보면 아직 큰 시장은 아니지만, 이 서비스를 쓰려는 사람에게 셀프 커스터디는 선택사항이 아니다.

셀프 커스터디에는 개인에게 돌아오는 분명한 이득도 있다. 거래소나 수탁 회사에 맡긴 코인은 화면에 뜨는 숫자, 즉 그 회사에 대한 청구권이다. 회사가 무너지거나 출금을 멈추거나 계정을 묶으면 잔고는 그대로인데 코인은 꺼낼 수 없다. 지난 몇 년 사이 큰 거래소가 무너지면서 이용자가 자산을 돌려받지 못한 일이 여러 차례 반복됐다. 반면 내가 키를 쥐고 있으면 자산을 움직일 권한은 나에게만 있다. 남의 영업 상태나 정책 변경에 내 자산이 좌우되지 않는다는 것, 그것이 셀프 커스터디가 주는 실질적인 이득이다.

생태계 전체로도 의미가 있다. 비트코인이 소수의 큰 회사에 몰려 있으면 그 회사들이 새로운 단일 실패지점이 된다. 한 곳이 뚫리거나 무너질 때 흔들리는 범위가 그만큼 넓어지고, 한자리에 모인 대량의 코인은 그 자체로 큰 표적이 된다. 반대로 보유가 여러 개인에게 흩어져 있으면 한 곳의 사고가 전체로 번지지 않는다. 3장에서 볼 기관들의 원칙이 하나같이 분산으로 수렴하는 것도 같은 이유다.

다만 이 가치들이 셀프 커스터디를 권할 근거가 되지는 않는다. 준비 없이 시작하면 자산을 잃을 위험이 커지고, 그 손실은 온전히 본인 몫이다. 이 글은 셀프 커스터디를 권하는 글이 아니라, 하기로 한 사람이 덜 잃도록 돕는 글이다.

02

사고는 어디에서 일어났는가

1장의 사고들은 사실 비트코인의 역사에서 새로운 일이 아니며 이전부터 되풀이되던 여러 유형의 사고들 중 하나일 뿐이다. 개인키에도 수명주기가 있다. 만들어지고, 기록되고, 보관되고, 서명에 쓰이고, 필요할 때 복구된다. 지금까지 알려진 셀프 커스터디 사고는 거의 전부 이 다섯 단계 중 어딘가에서 터졌다. 비트코인의 암호화 알고리즘 자체가 뚫려서 자산이 사라진 사례는 확인된 바 없다.

FIGURE 02 **개인키 수명주기의 5단계와 실제 사고가 발생한 지점**

사고는 암호가 해독되어 일어나지 않았다. 아래 다섯 단계 중 어느 한 곳이 무너져 발생했다.

1. 생성 난수와 환경	2. 백업 기록 매체	3. 보관 물리적 위치	4. 서명 거래 검증	5. 복구 재현 가능성
콜드카드 펌웨어 결함 (2026, 약 1,816 BTC) 랜드스톰 · 밀크새드	국세청 보도자료에 니모닉 사진 노출 (2026, 약 69억 원)	비트파이넥스 클라우드 라스트패스 보관함 울브리히트 노트북	안드로이드 논스 재사용 (2013) 잔돈 주소 조작 공격	백업 미검증으로 인한 자산 영구 손실 (통계 미집계)

읽는 법

방어는 다섯 단계 전부에 필요하다. 한 단계만 강화해도 다른 단계가 무너지면 자산은 사라진다. 5장의 실천 방안은 이 다섯 단계를 순서대로 다룬다.

생성 단계: 난수가 부족하면 지갑은 시작부터 약해진다

개인키의 안전은 그 값이 예측할 수 없는 무작위 값이라는 전제 위에 서 있다. 이 전제가 무너지면 이후의 방어도 함께 약해진다. 1장의 콜드카드 사고가 딱 이 유형이다. 기기를 오프라인으로만 쓰고 금속판에 백업했어도, 시드가 만들어질 때 무작위성이 부족했다면 그 노력만으로는 자산을 지킬 수 없다. 패스프레이즈(시드에 덧붙이는 본인만 아는 추가 암호)를 사용했다면 그 강도에 따라 공격을 막을 수 있지만, 시드의 취약성 자체가 사라지는 것은 아니다. 보안이 패스프레이즈 하나에 의존하게 되므로 전체 구성은 여전히 취약하다.

같은 유형의 문제는 전에도 있었다. 2011년부터 2015년 사이 일부 웹 지갑이 브라우저 안에서 개인키를 생성하는 과정에서 발생한 난수 취약점을 랜드스톰(Randstorm)이라 부른다. 당시 사용된 자바스크립트 라이브러리가 일부 브라우저에서 보안성이 낮은 난수 함수를 사용하면서 가능한 개인키의 범위가 좁아졌다. 그 결과 이 시기에 만들어진 일부 지갑은 외부에서 개인키를 복원할 수 있는 것으로 확인됐다.[7] 이후 다른 프로그램에서도 비슷한 문제가 발견됐다. 암호화폐 지갑 도구인 리브비트코인 익스플로러(Libbitcoin Explorer)의 시드 생성 기능에서 발견된 취약점을 밀크새드(Milk Sad, CVE-2023-39910)라 부른다.

이 프로그램은 시드를 만들 때 컴퓨터의 시각을 32비트 값으로 바꿔 난수 생성기의 시작값으로 사용했다. 그 결과 가능한 시드는 2의 32제곱, 약 43억 가지로 제한됐다. 이는 성능이 좋은 개인용 컴퓨터로 후보를 모두 대입해 개인키를 찾아낼 수 있는 수준이었다.[8]

> **이 단계에서 개인이 할 수 있는 것**
>
> 난수 생성 장치가 제대로 작동하는지 개인이 검증할 방법은 없다. 대신 주사위나 동전으로 만든 무작위 값을 직접 넣을 수 있는 기기를 쓰면, 기기의 난수 장치가 고장 나 있어도 사용자가 넣은 무작위성이 남는다. 콜드카드 사고에서도 주사위를 충분히 굴려 시드를 만든 사용자는 위험 대상에서 빠졌다. 5.2절에서 주사위 입력을 권하는 이유다.

백업 단계: 기록하는 방식이 곧 유출 경로가 된다

시드를 기록하는 순간이 두 번째 약점이다. 1장의 국세청 사고가 극단적인 예다. 니모닉이 적힌 종이를 사진으로 찍어 공개 자료에 실은 것만으로 자산이 털렸다.

개인에게 더 흔한 유출 경로는 디지털 기기에 저장하는 것이다. 2022년 라스트패스(LastPass) 침해에서 공격자는 온라인 비밀번호 보관함을 통째로 손에 넣은 뒤 잠금 암호를 풀었다. 피해자들의 공통점은 시드 구문(니모닉)을 온라인 보관함의 메모 칸에 적어 뒀다는 것이었다. 미국 수사 당국은 이 침해가 약 1억 5천만 달러 규모의 암호화폐 절도로 이어졌다고 결론지었고, 2025년 3월에 약 2,400만 달러어치를 압수했다.[9]

사진으로 남기는 것도 위험하다. 카스퍼스키(Kaspersky)가 공개한 스파크캣(SparkCat) 악성코드는 스마트폰 사진첩을 뒤져 글자를 읽어 내는 방식으로 시드 구문을 찾는다. 이 악성코드는 공식 앱 장터의 심사 절차에서도 걸러지지 않고 유포되었으며, 한국어를 포함한 여러 언어의 문자 인식 모델을 사용하여 니모닉으로 보이는 문자열을 찾아다녔다.[10]

보관 단계: 키가 보관된 위치가 곧 공격지점이 된다

키를 물리적으로 어디에 두느냐가 세 번째 공격 지점이다. 2013년 10월 미국 연방수사국(FBI)은 실크로드(Silk Road) 운영자를 체포할 때 노트북이 켜진 채 로그인된 상태 그대로 노트북을 확보했다. 잠기기 전에 압수했기 때문에 지갑까지 바로 들어갈 수 있었다.[11] 2016년 비트파이넥스(Bitfinex) 해킹 자금은 범인이 개인키를 클라우드 계정에 보관하고 있었다. 수사기관이 그 계정에 접근하면서 2022년 2월 약 94,000 비트코인이 압수됐다.[12]

공통점은 하나다. 키를 온라인 기기나 클라우드에 두는 순간, 지키는 힘은 암호학이 아니라 그 저장 공간의 잠금장치에서 나온다. 그리고 클라우드 계정은 법적 절차로도 열리고 해킹으로도 열린다.

서명 단계: 컴퓨터 화면은 신뢰의 기준이 아니다

네 번째 지점은 거래에 서명할 때다. 비트코인 서명에는 매번 일회용 난수인 논스가 쓰이는데, 같은 논스가 두 번 쓰이면 공개된 서명만 갖고도 개인키를 계산해 낼 수 있다. 2013년 8월 비트코인 프로젝트는 안드로이드(Android)의 난수 결함 때문에 일부 지갑 앱이 논스를 반복 생성해 자금 탈취에 노출됐다는 경고를 발표했다.[13] 이 사건 이후, 하드웨어 지갑을 고를 때 결정적 논스 생성 방식(RFC6979)을 지원하는지가 중요한 기준이 되었다. 이 방식이 서로 다른 거래에 같은 논스가 재사용되는 것을 막아 주기 때문이다.

또 다른 위협은 잔돈 주소 조작이다. 비트코인은 보내고 남은 금액이 잔돈으로 내 지갑에 돌아오는데, 감염된 컴퓨터의 악성코드가 이 잔돈이 돌아올 경로를 엉뚱한 값으로 바꿔치기하면 키를 갖고 있어도 그 경로를 모르는 한 돈을 찾지 못하게 된다.[14] 방어 원칙은 단순하다. 서명 전 거래 내용은 컴퓨터 화면이 아니라 하드웨어 지갑 화면으로 확인한다.

복구 단계: 검증되지 않은 백업은 백업이 아니다

마지막은 복구다. 이 단계의 사고는 뉴스에도 잘 나오지 않는다. 피해자가 한 명씩이고 범인이 없기 때문이다. 시드 단어를 잘못 적었거나, 패스프레이즈를 잘못 기억했거나, 멀티시그 구성 정보를 잃어버려서 내 돈에 내가 접근하지 못하게 되는 경우다. 통계에 잡히지 않을 뿐, 셀프 커스터디에서 실제로 자산을 잃는 가장 흔한 경로일 가능성이 있다.

이 단계의 고약한 점은 사고가 난 시점과 그것을 알아차리는 시점이 몇 년씩 벌어진다는 것이다. 백업을 잘못 적은 사람은 복구가 필요해진 순간에야 그 사실을 알게 되고, 그때는 이미 늦었다. 5.6절의 소액 전체 절차 테스트가 바로 이 문제를 겨냥한다.

사람과 물리 세계

다섯 단계 밖에도 위험은 있다. 2020년 6월 하드웨어 지갑 제조사 렛저(Ledger)의 쇼핑몰 데이터베이스가 뚫려 고객의 이름, 주소, 전화번호가 담긴 상세 기록 약 27만 2천 건과 이메일 약 100만 건이 새 나갔다. 렛저는 7월에 침해 사실을 알렸지만 상세 기록의 규모는 그해 12월에야 바로잡아 밝혔다.[15] 개인키가 털린 것은 아니었다. 하지만 암호자산을 가진 사람의 실명과 집 주소가 범죄자 손에 들어갔다는 점에서, 물리적 위협의 발판이 됐다.

보유 사실이 알려지면 기술과 무관한 위협이 시작된다. 사람을 직접 협박해 시드를 내놓으라는 공격에는 어떤 암호도 소용없다. 대응은 두 가지다. 보유 사실과 신원, 주소가 서로 연결되지 않게 관리하는 것, 그리고 협박당하는 상황에서 활용할 미끼 지갑을 갖추는 것이다. 미끼 지갑의 경우 5.1절의 패스프레이즈 이야기로 이어진다.

03

기관과 국가는 키를 어떻게 다루는가

앞 장이 사고가 발생한 지점을 살폈다면, 이 장은 기관과 국가의 사례에서 개인이 적용할 수 있는 교훈을 찾는다. 기관들의 키 관리 방식은 감사와 인증 때문에 꽤 많이 문서로 공개되어 있다. 여기서는 그 기술을 하나하나 나열하는 대신, 개인이 가져다 쓸 수 있는 교훈 다섯 가지를 추리고 각각 그 교훈을 가장 잘 보여주는 사례를 붙였다. 개인이 기관만큼 자원을 갖출 수는 없어도, 핵심 취지는 충분히 따라 할 수 있다.

교훈 1) 키가 만들어지는 순간의 환경을 통제한다

2018년 보도에 따르면 코인베이스(Coinbase)는 개인키(지갑 내 자산을 움직일 수 있는 비밀 열쇠)를 만들 때 전자파를 차단하는 텐트 안에서 작업한다. 한 변이 약 2.4미터인 정육면체 텐트로, 전자파를 막는 그물망을 둘러 패러데이 상자(Faraday cage) 역할을 하도록 한다. 설치 장소는 여러 보안 구역 중에서 무작위로 고르고, 전력선을 타고 정보가 새는 것까지 막으려고 차폐된 전원을 쓴다. 텐트 안에는 인터넷에 한 번도 연결된 적 없는 노트북 두 대와 접이식 탁자와 조명, 프린터가 전부다. 어느 노트북을 쓸지는 동전 던지기로 정하고, 작업이 끝나면 노트북 두 대를 모두 파기한다. 이 과정에 하루가 거의 다 들어간다.[16]

이 절차가 막으려는 것은 분명하다. 열쇠가 만들어지는 순간 그 공간에서 어떤 신호도 밖으로 새 나가지 않게 하고, 열쇠 생성에 쓴 기기는 다시 쓰지 않는 것이다. 개인이 텐트까지 갖출 수는 없다. 하지만 신호를 내보내거나 촬영과 녹음이 되는 기기를 전부 끈 방에서 열쇠를 만들고, 임시로 쓴 메모를 남기지 않는 것은 그대로 따라 할 수 있다. 구체적인 방법은 5.2절에 있다.

교훈 2) 한 사람이 단독으로 자산을 움직일 수 없게 한다

기관에서는 돈을 빼려면 여러 사람의 승인이 필요하다. 키를 만들고 백업하고 복구하는 일은 여러 명이 한자리에 모여 정해진 각본대로 진행하는데, 이 절차를 키 세리머니(key ceremony)라고 부른다. 출금도 여러 명 가운데 정해진 수가 함께 승인해야 하고, 중요한 작업은 반드시 두 사람 이상이 같이 처리한다. 누구도 혼자서는 자산을 움직일 수 없다.

개인에게는 같이 승인해 줄 동료가 없는 경우가 대부분이다. 하지만 이 원칙의 핵심은 사람 수가 아니라, 한 곳만 뚫려도 전부 무너지는 지점, 즉 단일 실패지점(single point of failure)을 없애는 데 있다. 개인은 이것을 키 여러 개로 구현한다. 서로 독립된 세 개의 키 가운데 두 개가 모여야 자산이 움직이는 멀티시그(multisig, 다중서명) 구성이 그 방법이고, 5.1절에서 다룬다.

교훈 3) 자산을 한 곳에 모아 두지 않는다

엘살바도르는 국가가 보유한 비트코인을 인터넷과 분리된 물리 금고로 옮긴 뒤, 한 지갑에 몰려 있던 준비금을 여러 주소로 다시 나눴다. 보도에 따르면 약 6,284 비트코인을 14개 주소에 나누면서 어느 주소에도 500 비트코인을 넘게 두지 않았다.[17] 이 유는 보안 강화와 단일 실패지점 줄이기였다.

개인에게는 주소 분산과 별개로 백업·키·자산을 나누는 원칙이 필요하다. 하나는 백업과 키를 서로 떨어진 여러 장소에 나눠 두는 것, 다른 하나는 전 재산을 지갑 하나에 몰아넣지 않는 것이다. 평소 쓰는 소액 지갑과 장기 보관용 지갑을 나눠 두면, 자주 만지는 지갑에서 실수가 나도 피해가 거기서 끝난다.

교훈 4) 중요한 작업은 대본으로 만들고 지킨 기록을 남긴다

기관과 국가는 중요한 작업을 사람의 기억에 맡기지 않고 매뉴얼로 관리한다. 스위스 크립토밸리협회가 2020년에 낸 「신뢰할 수 있는 키 세리머니 지침」은 키를 만드는 절차를 대본으로 만들라고 권고한다. 대본에는 누가 어떤 역할을 맡아 수행하는지와 사용하는 장비의 버전과 일련번호까지 적고, 진행자와 조작자 외에 감사인이나 공증인 같은 독립된 입회인을 최소 한 명 둔다. 부품은 변조 방지 봉투에 넣어 봉인하고, 영상과 사진과 참가자 서명처럼 나중에 확인할 수 있는 증거를 남긴다. 눈에 띄는 것은 감사인이 일련번호가 제대로 적혔는지 확인한다는 식으로 겉으로 드러나지 않는 확인 행위까지 대본에 적으라고 한 대목과, 절차에서 벗어난 일은 전부 독립 입회인이 기록한다고 못박은 대목이다.[18]

기관이나 국가는 사고의 교훈을 매뉴얼에 반영하여 보안을 강화한다. 한국도 1장의 국세청 사고 뒤에 같은 과정을 거쳤다. 국세청은 사과와 함께 대외 공개 자료를 미리 심의하는 내부 통제를 두겠다고 밝혔고, 감사원은 검찰청과 경찰청, 국세청, 관세청 네 기관의 압수물 관리 실태를 들여다봤다. 2026년 4월에는 관계부처가 합동으로 개선방안을 내놨다. 압수한 자산은 현장에서 곧바로 인터넷과 끊긴 기관 지갑으로 옮기고, 개인키와 복구 구문은 두 사람 이상이 나눠 관리하며, 보관 장소에 금고와 폐쇄회로 화면을 두고, 유출 사고를 가정한 모의훈련을 해마다 한 번 이상 한다는 내용이다.[19]

개인 차원에서 위 사례를 다음과 같이 벤치마킹할 수 있다. 첫째, 대본을 만든다. 지갑을 만들 때, 백업을 꺼낼 때, 목돈을 옮길 때 무엇을 어떤 순서로 하는지 번호를 붙여 적어 두고 그대로만 한다. 둘째, 기록을 남긴다. 사용한 하드웨어 지갑의 종류와 사용 일자를 기록한다. 계획에서 벗어나는 일이 생기면 즉흥적으로 대응하지 말고 우선 무슨 일이 있었고 어떻게 대응할지 적는다.

교훈 5) 스스로의 판단이 아니라 외부 기준으로 검증한다

기관은 스스로 안전하다고 믿는 대신 바깥의 기준과 감사로 검증받는다. 대표적인 것이 암호자산 보안 표준(CryptoCurrency Security Standard, CCSS)이다. 비영리 기구인 암호화폐 인증 컨소시엄(CryptoCurrency Certification Consortium, C4)이 관리하는 이 표준은 셀프 커스터디 시스템에 41개 점검 항목을 적용한다. 키를 충분히 무작위로 만드는지, 생성부터 사용·폐기까지 절차와 기록을 남기는지, 키 유출에 대비하고 백업과 권한을 나눠 한 사람이나 한 지점의 실패가 자산 손실로 이어지지 않게 하는지 등을 살핀다. 여기에 암호 모듈을 검증하는 FIPS 140-2와 140-3, 재무보고 관련 통제를 다루는 SOC 1, 보안·가용성 등 서비스 조직의 통제를 다루는 SOC 2가 더해진다.[20] 적용 대상과 검증 범위는 서로 다르지만, 기관 내부의 판단에만 의존하지 않고 외부 기준에 따라 검증받는다는 점은 같다.

개인이 인증을 받을 수는 없다. 핵심은 자기 판단만 믿지 말고, 그 판단을 검증하는 절차를 따로 두라는 것이다. 지갑을 만들면 반드시 소액으로 입금과 송금을 해 보고, 백업만으로 복구가 되는지 시험해 보고, 기기가 보여주는 주소를 다른 도구로 다시 계산해 맞춰 보고, 펌웨어의 지문값(해시)을 제조사 공식 문서와 대조한다. 5장 내용 대부분이 이 원칙을 실제로 적용하는 방법이다.

04

개인 차원의 셀프 커스터디 원칙

2장의 사고와 3장의 교훈에서 개인이 지킬 원칙을 여덟 가지로 추렸다. 각 원칙이 어느 사고와 교훈에서 나와 5장의 어느 지침으로 이어지는지는 미주에 정리했다.[21] 이 원칙들이 5장의 실천 방안으로 이어진다.

원칙 1. 단일 실패지점을 만들지 않는다. 키 하나, 장소 하나, 제조사 하나, 기억 하나가 무너졌다고 전 재산이 사라지는 구성은 피한다.

원칙 2. 키를 만드는 환경을 통제한다. 신호를 내보내거나 촬영과 녹음이 되는 기기를 전부 끈 상태에서 키를 만들고, 주사위처럼 직접 만든 무작위성을 더한다.

원칙 3. 백업은 디지털 흔적을 남기지 않는다. 사진, 클라우드, 비밀번호 관리자, 이메일, 메신저 모두 실제로 대형 사고가 났던 경로다.

원칙 4. 백업과 키는 지리적으로 나눈다. 같은 시드를 지키는 기기와 백업을 한 장소에 두지 않는다.

원칙 5. 서명 전 확인은 하드웨어 지갑 화면에서 한다. 보낼 주소와 금액, 수수료 등 거래 내용을 기기에서 직접 확인한다.

원칙 6. 목돈을 옮기기 전에 소액으로 전체 절차를 검증한다. 복구까지 실제로 해 본 백업만 백업으로 친다.

원칙 7. 내 판단이 아니라 독립된 수단으로 검증한다. 기기가 보여주는 값은 다른 도구로 다시 계산해 맞춰 보고, 기준값은 제조사 공식 문서에서 가져온다.

원칙 8. 자신의 지갑 관리 수칙과 키 관리 매뉴얼을 작성한다. 지갑을 사용한 내역을 기록한다.

05

개인 차원의 실천 방안

이 장은 4장의 원칙을 구체화한 실천 방안이다. 키의 수명주기를 따라 생성, 백업, 보관, 서명, 복구 순서로 다루고, 끝에 인공지능을 검증 도우미로 쓰는 법과 유출 대응 절차를 추가했다. 구성을 고를 때는 보유 규모, 기기 숙련도, 복구·상속을 맡을 사람, 운영에 쓸 수 있는 시간을 함께 고려한다.

구성의 선택: 패스프레이즈와 멀티시그

학습·소액·초보자는 검증된 단일서명 지갑과 백업·복구 시험부터 시작할 수 있다. 단일 시드는 하나의 비밀 유출·분실에 취약하지만 복잡한 구성보다 관리가 쉽다. 니모닉 유출에 대비할 필요가 있는 숙련 사용자는 충분히 길고 고유한 BIP39 패스프레이즈를 추가하고, 고액·고급 운영은 충분히 시험하고 문서화한 2-of-3 멀티시그를 검토한다.

선택지는 단일서명, 패스프레이즈를 더한 단일서명, 멀티시그로 나뉜다.

FIGURE 03 단일 시드, 패스프레이즈, 멀티시그의 구조 차이

핵심 질문은 하나다. 어느 한 곳이 무너졌을 때 자산이 사라지는가.

패스프레이즈는 도난 저항을 높이지만 기억이라는 새로운 단일 실패지점을 만든다. 멀티시그는 도난과 분실을 함께 방어하지만 구성 정보까지 백업해야 한다. 5장에서 두 갈래의 선택 기준을 다룬다.

패스프레이즈 구성

패스프레이즈(BIP39 passphrase)는 12개 또는 24개 단어 시드에 본인만 아는 문자열을 하나 더 얹어 전혀 다른 지갑을 만들어 내는 방식이다. 흔히 25번째 단어라고 부른다. 시드가 적힌 금속판을 도둑맞아도 패스프레이즈를 모르면 돈에 손댈 수 없다.

장점은 단순함이다. 기기 한 대면 되고 추가 비용도 없다. 패스프레이즈 없이 열리는 지갑에 소액을 넣어 두면, 협박당했을 때 그 지갑만 보여주는 미끼로도 쓸 수 있다. 단점도 뚜렷하다. 자금에 접근하려면 두 요소가 모두 필요하다는 점에서 패스프레이즈 방식은 니모닉과 패스프레이즈 두 조각으로 이루어진 2-of-2 멀티시그와 비슷하다. 둘을 따로 보관하고 패스프레이즈를 충분히 강하게 정했다면, 하나만 유출돼서는 자산에 접근하기 어렵다. 반대로 둘 중 하나만 잃어도 같은 지갑을 복구할 수 없다. 도난에는 강한 대신 분실 쪽으로는 단일 실패지점이 하나 더 생기는 셈이다. 특히 패스프레이즈를 머릿속에만 두고 기록해 두지 않으면 기억 자체가 그 실패지점이 된다. 또 1장의 콜드카드 사고처럼 기기가 취약한 니모닉을 만들었다면, 짧거나 흔한 패스프레이즈를 더하는 것만으로는 충분히 방어하기 어렵다.

멀티시그 구성

멀티시그는 서로 독립된 키 세 개 가운데 두 개가 서명해야 자산이 움직이는 2-of-3 구성이 일반적이다. 키 하나를 도둑맞아도 그 키만으로는 자산을 옮길 수 없고, 키 하나를 잃어도 남은 두 키로 자산에 접근할 수 있다. 이처럼 도난과 분실에 함께 대비할 수 있다는 것이 장점이다.

세 키는 서로 다른 제조사 기기로 만드는 것이 중요하다. 같은 회사 기기 세 대로 채우면 그 회사 펌웨어에 결함이 나오는 순간 세 키가 같이 위험해진다. 뒤집어 말하면, 한 제조사의 결함이 다른 제조사 기기에서 만든 키까지 약하게 만들 구조적 이유는 없다. 1장의 콜드카드 사고에 대입하면, 다른 두 키가 안전한 2-of-3 구성에서는 콜드카드로 만든 취약한 키를 제외하고 새 키를 넣은 지갑을 만든 뒤 기존 자산을 옮길 수 있었다. 단점은 구성과 운영이 번거롭고, 키 말고도 지갑 구성 정보인 출력 디스크립터(output descriptor)까지 백업해야 한다는 것이다.

기기 선택의 여섯 가지 기준

구성을 정했으면 다음은 기기 고르기다. 하드웨어 지갑은 사양이 제각각이라 제조사 설명만 봐서는 비교가 어렵다. 앞에서 본 사고들이 무엇을 노렸는지를 기준으로 삼으면 여섯 가지로 정리된다.

첫째, 에어갭 운용. 서명기를 컴퓨터와 실시간 데이터 연결 없이 사용하고 QR 코드나 microSD 카드로 부분 서명 비트코인 거래(Partially Signed Bitcoin Transaction, PSBT)를 옮기는 방식이다. 직접 연결을 피함으로써 공격면을 줄이지만, 데이터 전달 경로와 서명기 자체의 위험까지 없애지는 않는다.

둘째, 물리적인 키 데이터 탈취 방어. 기기를 통째로 도둑맞았을 때 키를 지켜 내는 항목이다. 키를 밖으로 꺼낼 수 없게 설계된 전용 보안 칩(시큐어 엘리먼트)에 암호화해 가두는 방식과, 아예 키를 기기에 저장하지 않고 전원을 끄면 지워 버리는 방식 두 갈래가 있다. 어느 쪽이든 기기를 손에 넣어도 키를 뽑아낼 수 없게 만드는 것이 목적이다.

셋째, 공급망 공격 방어. 포장에 변조 방지 봉인이 있는지, 뜯긴 흔적을 감지하는 장치가 있는지, 분해가 감지되면 키를 스스로 지우는지를 본다. 배송 중 바꿔치기를 노린 항목이다.

넷째, 서명 방식에 맞는 안전한 논스 생성 절차와 구현 검증 여부. ECDSA에는 RFC6979와 같은 결정적 논스 생성 방식을 사용할 수 있고, 탭루트(Taproot)의 슈노르(Schnorr) 서명은 BIP340 절차를 따른다.

다섯째, 패스프레이즈 및 멀티시그 기능 지원. 에어갭 상태에서 여러 서명기가 차례로 서명하려면 PSBT를 지원해야 한다.

여섯째, 사용자가 직접 무작위성을 넣을 수 있는가. 주사위나 동전으로 만든 값을 시드에 반영할 수 있으면 기기의 난수 장치가 고장 나 있어도 내가 넣은 무작위성은 남는다. 1장의 콜드카드 사고는 앞의 다섯 항목으로 걸러지지 않는데, 이 항목이 그 공백을 메운다. 실제로 제조사도 주사위를 50회 이상 굴려 엔트로피를 직접 넣은 시드는 이번 결함의 영향을 받지 않는다고 밝혔다.

FIGURE 04 **위협 유형별 방어력 비교**

어느 쪽이 우월한 것이 아니라 막아주는 위협이 다르다. 자신이 두려운 위협이 무엇인지가 선택 기준이다.

	시드 + 패스프레이즈	2-of-3 멀티시그
백업 물리적 도난	🟢 강함	🟢 강함
기기 1대 분실·고장	🔴 취약	🟢 강함
기억 상실 · 사고	🔴 취약	🟢 강함
제조사 결함 · 공급망	🔴 취약	🟢 강함
물리적 강압(협박)	🟢 강함	🟠 보통
구성의 복잡성 부담	🟢 강함	🔴 취약

판단 기준

분실과 기기 결함이 더 두렵다면 멀티시그, 물리적 도난과 강압이 더 두렵고 단순함이 필요하다면 패스프레이즈를 택한다.

이 보고서가 참조한 공개 기능 비교 데이터베이스에서 여섯 가지 기능 기준을 모두 만족한 모델은 네 제조사의 여섯 모델이었다. 서로 다른 회사의 보안 칩을 두 개 겹쳐 넣은 콜드카드 Mk4와 Mk5, Q, 보안 칩 세 개와 제조사가 설명하는 PCI 수준의 변조 방지 기능을 갖춘 키스톤 3 Pro, EAL6+ 등급 보안 칩을 쓴 카드웨어(Cardware), 그리고 보안 칩 대신 키를 저장하지 않는 쪽을 택한 시드사이너(SeedSigner)다. 시드사이너는 전원을 끄면 시드가 지워지니 저장된 키가 없어 훔쳐 갈 키도 없고, 부품을 직접 사서 조립하는 기기라 값이 싸고 배송 중 바꿔치기 걱정도 적다.[22] 다만 이는 기능 기준에 따른 후보 목록이지 제품 전체의 안전성 인증은 아니다. 실제 선택에서는 공식 기능 문서, 독립 보안 검토, 공개된 취약점과 대응 이력, 펌웨어 검증 방법, 공급망·복구 위험을 함께 확인해야 한다.

목록 맨 앞의 콜드카드는 1장의 사고를 낸 바로 그 제조사다. 그런데도 목록에 남긴 이유는 두 가지다. 문제가 기기 자체의 침해가 아니라 펌웨어의 시드 생성 코드에서 발생했고, 제조사가 취약점 공개 다음 날 영향받은 모든 모델의 수정 펌웨어를 배포했기 때문이다. 다만 펌웨어를 업데이트해도 이미 만들어진 취약한 시드가 안전해지는 것은 아니다. 제조사는 새 시드를 만들어 자산을 옮기라고 권고했다. 공개 공지에는 별도의 보상이나 교환 절차가 안내되지 않았다.[23]

멀티시그를 구성한다면 서로 다른 제조사를 골라 섞는 것이 핵심이다. 콜드카드, 키스톤, 시드사이너처럼 만든 회사와 설계가 모두 다른 세 대로 2-of-3을 구성하면, 한 회사의 펌웨어에 구멍이 나도 남은 두 키로 자산을 옮길 수 있다. 1장의 콜드카드 사고에 대입하면, 세 키 가운데 하나만 콜드카드였다면 그 키가 뚫려도 남은 두 키의 보안은 뚫리지 않고 해커는 자금을 옮길 권한을 얻을 수 없다.

FIGURE 05 **여섯 가지 기준을 모두 충족한 하드웨어 지갑**

공개 데이터베이스의 기기 74종 가운데 여섯 기준을 모두 채운 것은 네 제조사의 여섯 모델이었다.

콜드카드 Mk4 · Mk5 · Q

서로 다른 회사의 보안 칩 두 개를 겹쳐 넣었다. 주사위 입력을 지원한다.

키스톤 3 Pro

결제 단말기 수준의 인증(PCI)을 받은 보안 칩을 세 개 넣었다. 오픈소스다.

카드웨어(Cardware)

EAL6+ 등급 보안 칩을 쓴다. 2025년에 나온 신형이라 소스는 아직 공개 전이다.

시드사이너(SeedSigner)

보안 칩 대신 키를 저장하지 않는다. 전원을 끄면 시드가 지워진다.

고르는 법

멀티시그를 짠다면 이 가운데 서로 다른 제조사를 골라 섞는다. 같은 회사 기기로 채우면 그 회사 펌웨어에 결함이 나오는 순간 키가 한꺼번에 위험해진다.

생성 단계: 키를 만드는 환경

키를 만들기 전에 공식 판매처인지 확인하고, 포장과 기기에 변조 흔적이 없는지 살핀다. 사전 초기화된 기기나 미리 적힌 시드는 거부한다. 공식 경로에서 최신 펌웨어를 받고 서명 또는 해시를 확인한 뒤 모델과 펌웨어 버전을 기록한다.

키는 인터넷에 연결된 적 없는 하드웨어 지갑에서 만든다. 일반 컴퓨터나 스마트폰 앱으로 만든 시드는 그 기기가 이미 감염돼 있었다면 만들어지는 순간부터 유출된 것으로 봐야 한다. 브라우저에서 시드를 만드는 방식이 특히 위험하다는 것은 2.1절의 랜드스톰 사례가 보여준다.

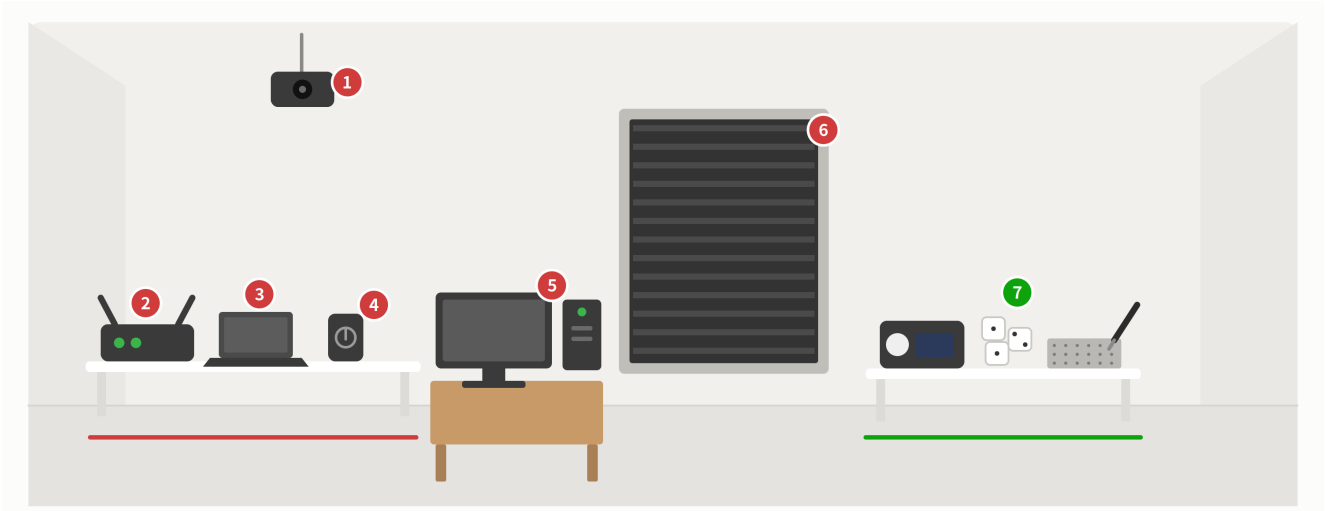
3.1절의 기관 절차를 개인 수준으로 줄이면 이렇게 된다. 키를 만들고 시드를 적는 동안, 주변에서 신호를 내보내거나 소리와 영상을 기록할 수 있는 기기를 전부 끈다.

여기에 하나 더, 기기가 지원한다면 주사위로 만든 무작위 값을 직접 넣는 기능을 쓰자. 주사위를 충분히 여러 번 굴려 그 결과를 시드에 반영하면, 기기의 난수 장치가 고장 나 있어도 내가 넣은 무작위성은 남는다. 콜드카드 사고에서도 이러한 방식으로 콜드카드를 사용한 사람은 해킹 위협과 무관했다.

니모닉 단어 수는 12개면 충분하다. 12단어는 무작위 값 128비트에 검산용 비트(체크섬) 4비트를 붙인 132비트를 11비트씩 자른 것이고, 24단어는 256비트에 8비트를 붙여 같은 방식으로 자른 것이다. 비트코인 서명에 쓰이는 암호의 실질 강도가 약 128비트라서, 24단어를 써도 개인키의 보안 수준은 12단어와 사실상 같다. 단어가 적을수록 받아 적고, 맞춰 보고, 복구할 때 실수할 여지가 줄어든다는 것은 12단어의 실속 있는 장점이다.

FIGURE 06 **키를 만드는 방: 무엇을 끄고 무엇을 남길 것인가**

기관이 전자기 차폐 텐트를 쓰는 이유를 개인 수준으로 옮기면, 신호를 내보내거나 기록하는 기기를 모두 없애는 것이다.



번호	대상	조치
1	**실내 CCTV · 홈캠**	초인종 카메라까지 전원을 차단한다.
2	**공유기 · 와이파이**	전원을 차단하거나 랜선을 분리한다.
3	**노트북 · 태블릿 · 웹캠**	다른 방으로 옮기는 것이 확실하다.
4	**스마트폰 · 스마트워치**	비행기 모드가 아니라 전원을 종료한다.
5	**데스크톱 · 모니터 · 스마트 스피커 · TV**	마이크가 있는 기기는 전부 종료한다.
6	**창문과 유리문**	블라인드를 끝까지 내린다. 가리지 않으면 밖에서 촬영이 가능하다.
7	**남길 것**	하드웨어 지갑, 주사위, 금속판과 각인 도구.

빨간 번호는 끄거나 치울 것, 초록 번호는 남길 것이다.

이 방에서만 시드를 만들고 손으로 기록한다. 촬영하지 않고, 컴퓨터에 입력하지 않고, 임시 메모지는 작업 직후 폐기한다.

실제 예시: 시드사이너에서 주사위로 시드를 만들어 스패로우에 등록하기

앞의 원칙이 실제로 어떻게 굴러가는지 시드사이너와 스패로우 지갑(Sparrow Wallet)으로 정리한다. 시드사이너를 고른 이유는 셋이다. 주사위 입력을 지원하고, 키를 저장하지 않아 몇 번이고 연습하기 좋고, 카메라와 QR만으로 움직여서 인터넷과 떨어져 있다는 것이 눈으로 확인된다.

먼저 그림 6의 환경을 만들고 기기를 켠다. ‘Tools’ 메뉴에서 주사위 아이콘이 붙은 ‘New seed’를 고르면 주사위 눈금 입력 화면이 나온다. 12단어는 50번, 24단어는 99번을 굴려서 나온 눈금을 그대로 넣는다. 주사위는 면이 고르게 나오는 것을 쓰고, 입력하는 동안 화면 촬영 등은 하면 안 된다.

FIGURE 07 **시드사이너에서 주사위로 시드를 만드는 과정**

기기의 난수 생성기를 신뢰하는 대신 사용자가 직접 무작위성을 넣는다. 콜드카드 사고에서 지갑을 바꿀 필요가 없던 집단이 이 방식을 썼다.

1단계 Tools → New seed

주사위 아이콘이 붙은 항목을 고른다. 카메라 항목은 사진 기반 방식이다.

2단계 주사위 99회 입력

12단어는 50회, 24단어는 99회를 굴려 눈금을 그대로 입력한다.

3단계 생성된 단어 기록

화면에 표시된 단어를 금속판에 옮긴다. 촬영하지 않는다.

4단계 Export Xpub 선택

상단의 지문값이 이후 대조 기준이 된다.

주의

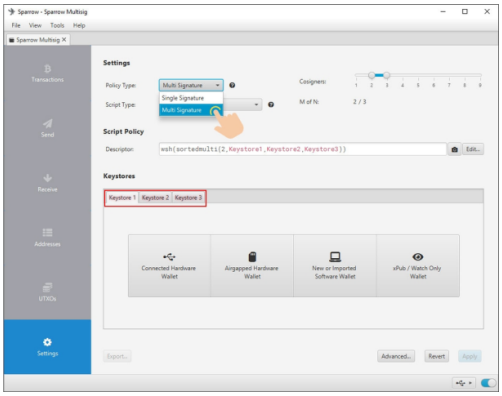
주사위 눈금을 입력하는 화면은 촬영 금지 대상이다. 이 그림의 값은 예시이며 실제 시드와 무관하다.

입력이 끝나면 단어들이 화면에 뜬다. 이 단어를 금속판에 새기고, 기기 화면과 다시 한 번 맞춰 본다. 이후 컴퓨터에는 개인키 원문을 전송하지 않고 공개키와 서명된 거래만 전달한다. 시드사이너는 전원을 끄면 시드가 지워지므로 다음 사용 때 백업한 니모닉을 다시 입력한다.

다음은 지갑 프로그램에 등록할 차례다. 기기에서 ‘Export Xpub’을 고르면 확장공개키(개인키 없이 그 지갑의 주소들만 만들어 볼 수 있는 열람용 키)가 QR로 표시된다. 이때 파생 경로(하나의 시드에서 주소를 만들어 갈 때 따라가는 갈래 표시)를 확인해야 한다. 비트코인 메인넷에서는 네이티브 세그윗 단일서명 지갑에 m/84'/0'/0', 네이티브 세그윗 멀티시그 지갑에 m/48'/0'/0'/2'를 쓰므로 구성에 맞는 경로를 고른다.[24] 컴퓨터에서 스패로우를 열고 ‘Keystore’의 ‘Airgapped Hardware Wallet’에서 ‘SeedSigner’의 ‘Scan’을 눌러 이 QR을 읽는다.

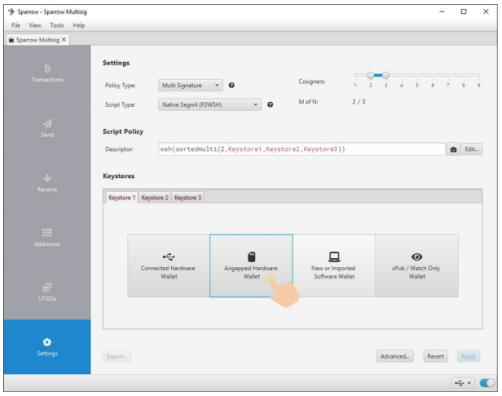
FIGURE 08 **시드사이너를 2-of-3 멀티시그로 스패로우에 등록하는 과정**

개인키는 기기 밖으로 나오지 않는다. QR로 넘어가는 것은 확장공개키와 파생 경로뿐이다.



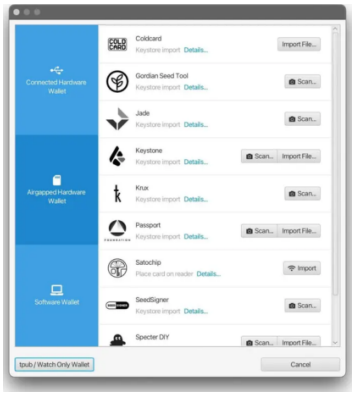
1단계 Policy Type을 Multi Signature로

Cosigners를 3으로 맞추면 M of N이 2/3이 되고 Keystore 탭이 세 개 생긴다.



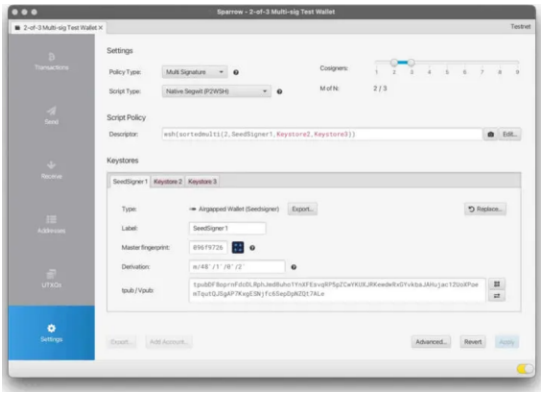
2단계 Keystore 1에서 에어갭 지갑 선택

스크립트 유형은 Native Segwit(P2WSH), 디스크립터는 wsh(sortedmulti(2,…))가 된다.



3단계 기기 목록에서 해당 기기의 Scan

기기마다 다른 제조사를 골라 세 번 반복한다. 같은 제조사로 채우지 않는다.



4단계 지문과 파생 경로 대조

멀티시그 경로는 m/48'로 시작한다. 화면의 값과 기기의 값이 같은지 확인한다.

확인

4단계 화면은 테스트넷 지갑이라 tpub과 coin type 1이 쓰였다. 본 자금용 지갑에서는 xpub과 coin type 0이 나온다.

등록이 끝나면 스패로우에 뜬 마스터 지문(키를 알아보는 여덟 자리 식별값)과 파생 경로가 기기 화면의 값과 같은지 꼭 맞춰 본다. 다르면 엉뚱한 기기나 계정을 등록한 것이니 바로 멈춘다. 이 대조가 5.5절 주소 검증의 출발점이다.

2-of-3 멀티시그를 만들려면 스패로우의 'Policy Type'을 'Multi Signature'로 바꾸고 2 of 3을 지정한 뒤, 위 과정을 서로 다른 제조사 기기 세 대로 반복한다. 지면상 다른 제조사 화면은 생략하지만, 앞 절의 여섯 기준을 채운 기기 가운데 회사가 서로 다른 것을 고르는 것이 핵심이다. 세 키가 다 등록되면 지갑 구성 정보인 디스크립터가 만들어지는데, 이것은 키와 별도로 꼭 백업한다. 디스크립터 없이 키 세 개만 있으면 복구가 어려울 수 있다. 스크립트 유형과 파생 경로, 서명자 순서를 전부 맞춰야 원래 주소가 나오기 때문에, 그 조합을 하나하나 시도하는 일은 사실상 별도의 복구 작업이 된다.

백업 단계: 니모닉을 어떻게 기록할 것인가

종이는 불과 물에 약하니 금속 기록이 기본이다. 그런데 금속이라고 다 같지 않다. 공개된 내구성 시험 가운데 반복 검증된 것은 제이미슨 로프(Jameson Lopp)가 75종 넘는 제품으로 진행한 스트레스 테스트다. 약 1,090도 화염을 10분 가한 뒤 물에 담그고, 염산에 부식시키고, 20톤 프레스로 눌러 봤다. 시험 온도는 일반 주택 화재의 평균인 약 590도의 두 배쯤으로 잡았다.

결과를 가른 것은 소재와 구조였다. 알루미늄 제품은 녹는점이 주택 화재 온도에 못 미쳐 대부분 녹아내렸다. 글자 타일을 레일에 끼우는 조립식은 열에 레일이 뒤틀리며 타일이 빠져 기록을 잃었다. 반면 스테인리스 판 한 장에 점을 찍거나 글자를 때려 박는 방식은 불, 산, 압착을 모두 버텼다. 부품이 적을수록 고장 날 곳도 적다는 원칙이 그대로 확인된 셈이다.[25]

소재는 304 스테인리스 이상이면 된다. 티타늄도 좋지만 현실적인 재난 대비에서 스테인리스와 의미 있는 차이는 없었다. 방식은 판 한 장에 직접 새기는 것을 권한다. 조립식 타일 구조는 위에서 본 대로 열에 약했고, 검은 코팅으로 글자를 드러내는 방식은 화염에서 30초를 못 버텼다. 각인 대행 서비스는 쓰지 않는다. 업체 직원, 주문 기록, 배송 경로 세 군데에 시드가 노출되기 때문이다. 참고로 BIP-39 단어들은 앞 네 글자만으로 서로 구분되게 만들어져 있어서, 자리가 모자라면 단어마다 앞 네 글자만 새겨도 된다.

패스프레이즈를 쓴다면 백업할 것은 니모닉만이 아니다. 니모닉과 패스프레이즈 중 하나라도 사라지면 같은 지갑을 복구할 수 없다. 기억은 백업이 아니다. 패스프레이즈도 니모닉과 똑같이 금속에 기록해 남긴다.

FIGURE 09 **금속 백업 방식별 내구성 시험 결과**

약 1,090도 화염, 염산 침지, 20톤 압착 시험. 방식에 따라 결과가 갈렸다.

센터펀치(점 타각)	● 우수	화재 · 산 · 압착 모두 손실 없음
전기화학 에칭	● 우수	산 노출 후 오히려 판독성 향상
스탬핑(글자 각인)	● 우수	변색과 휨은 있으나 판독 가능
코팅 표시 의존형	● 주의	검은 코팅이 화염에서 30초를 못 버팀
타일 · 글자판 조립식	● 부적합	열로 레일이 뒤틀려 타일 이탈
알루미늄 소재 전반	● 부적합	녹는점이 낮아 화재에서 녹아내림

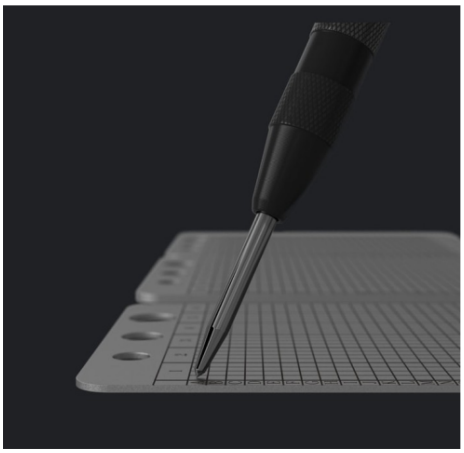
권고: 부품이 적은 단일 스테인리스 판에 직접 타각하는 방식이 가장 단순하면서 시험 결과가 일관되게 좋았다.

각인 대행 서비스는 제3자에게 시드를 노출시키므로 사용하지 않는다.

자료: Jameson Lopp, 금속 백업 스트레스 테스트(제품 75종 이상)

FIGURE 10 **금속 백업의 두 가지 기록 방식**

시험에서 화재와 산, 압착을 모두 통과한 방식이다. 어느 쪽이든 본인이 직접 새긴다.



점 타각(센터펀치)

격자판 위 해당 단어 번호 위치에 자동 센터펀치로 점을 찍는다. 부품이 가장 적다.

스탬핑(글자 각인)

글자 펀치와 망치로 단어를 판에 때려 박는다. 열에 변색되어도 판독된다.

주의

각인 대행 서비스는 쓰지 않는다. 업체 직원과 주문 기록, 배송 경로 세 곳에 시드가 노출된다.

샤미르 백업과 시드 XOR: 기관의 분산을 개인이 흉내 낼 때

기관이 키를 여러 조각으로 나누는 방식을 개인이 흉내 낼 수도 있다. 대표적인 것이 SLIP-39 샤미르 백업(Shamir backup)이다. 시드를 여러 조각으로 나누고 정해진 개수만 모이면 복원되게 하는 방식으로, 예를 들어 세 조각 중 두 조각으로 복원하게 짤 수 있다. 조각 하나가 새 나가도 아무 정보도 새지 않고, 하나를 잃어도 복구된다. 일부 하드웨어 지갑은 이것을 기본 백업 방식으로 쓴다.

다만 한계가 있다. 시드를 나누거나 다시 합칠 때에는 온전한 시드가 한 기기에 모인다. 이때 기기가 감염돼 있다면 조각을 나눠 보관한 효과가 사라질 수 있다. 반면 멀티시그는 각 키를 서로 다른 기기에서 생성·보관하면 여러 개인키를 한 기기에 모을 필요가 없다. SLIP39의 각 조각에는 입력 오류를 확인하는 RS1024 체크섬이 들어 있어, 기록하거나 입력하는 과정에서 단어 세 개 이하에 생긴 오류를 검출할 수 있다.[26] 그러나 이를 지원하는 기기와 소프트웨어가 제한적이고 생성·복구·상속 절차도 복잡하므로, 사용하려는 기기와 소프트웨어가 서로 호환되는지 미리 확인해야 한다.

콜드카드의 시드 XOR은 니모닉을 비트 단위 연산으로 여러 조각으로 나누는 방식이다. 특이한 점은 각 조각이 그 자체로 멀쩡한 지갑처럼 보인다는 것이다. 협박당했을 때 조각 하나를 내줘도 상대는 그것이 진짜 지갑인지 조각인지 구분하기 어렵다. 대신 조각을 전부 모아야만 복원되므로 하나만 잃어도 되돌릴 길이 없다. 즉, 분실 가능성을 낮춰주는 효과는 없다.[27]

직접 고안한 분할 방식은 쓰지 않는다

24단어를 앞뒤로 잘라 따로 보관하는 식으로 직접 방법을 고안하는 것은 금물이다. 검증된 적이 없을뿐더러, 단어 일부가 알려지면 공격자가 대입해야 할 조합도 그만큼 줄어든다. 나눠야 한다면 SLIP-39나 시드 XOR처럼 검증된 방식을 쓴다.

보관 단계: 어디에 둘 것인가

보관 원칙은 두 가지다. 하나, 같은 시드를 지키는 기기와 그 백업을 한 장소에 두지 않는다. 둘, 백업이나 멀티시그 키들을 서로 떨어진 여러 곳에 나눈다. 집 금고, 다른 지역의 은행 대여금고, 믿을 만한 가족의 보관처 같은 곳이 후보다. 패스프레이즈를 쓴다면 그 기록도 니모닉과 같은 원칙으로, 다만 니모닉과는 다른 장소에 둔다. 둘을 한곳에 두면 한 번에 함께 넘어가 패스프레이즈를 더한 의미가 사라진다.

은행 대여금고를 쓸 때는 알아 둘 것이 있다. 금고 안 내용물은 예금자 보호 대상이 아니고, 은행이 따로 보험을 들어 주지도 않는다. 임대료가 밀리면 은행이 금고를 열어 내용물을 옮길 수도 있다. 그래서 시드 하나짜리 구성의 전체 백업을 대여금고에 넣는 것은 권하지 않는다. 멀티시그 키 하나, 샤미르 조각 하나처럼 그것만으로는 돈을 못 움직이는 것을 넣는 편이 안전하다. 한 은행에 멀티시그 정족수를 채울 만큼의 키를 몰아넣는 것도 금물이다.[28]

대여금고든 가족이든 남의 공간에 백업을 맡긴다면 이렇게 하기를 권한다. 위변조 방지 봉투에 넣고 서명과 날짜를 적은 뒤 일련번호를 따로 기록해 둔다. 무엇을 왜 맡기는지 직원에게 말하지 않는다. 6개월에서 1년마다 실물과 봉인 상태를 살핀다. 상속까지 생각한다면 이용 은행에 사망 시 이용중지 여부와 상속인의 개봉·반환 절차, 필요서류를 미리 확인하고 비상 접근 계획에 반영한다.

서명 단계: 무엇을 확인할 것인가

돈을 받을 때는 하드웨어 지갑 화면에서 받는 주소를 직접 확인한다. 컴퓨터나 폰 화면의 주소는 악성코드가 바꿔치기할 수 있다. 보낼 때는 하드웨어 지갑 화면에서 보낼 주소와 금액, 수수료를 확인한다. 잔돈은 기기마다 표시하고 검증하는 방식이 다르므로, 예상하지 않은 전송 내역이나 잔돈 관련 경고가 없는지도 살핀다.

한 단계 더 확인하고 싶다면 기기가 보여주는 주소를 독립된 도구로 다시 계산해 맞춰 본다. 비트코인 코어(Bitcoin Core)의 주소 파생 기능이나 하드웨어 지갑 연동 도구를 쓰면 확장공개키에서 주소를 다시 만들어 낼 수 있다. 두 값이 같으면 그 주소가 정말 내 지갑 것임을 따로 확인한 셈이다.[29] 여러 제조사 기기로 멀티시그를 짰다면 기기마다 받는 주소를 교차 확인할 수 있어서, 한 대가 감염돼도 가짜 주소에 속지 않는다.

복구 단계: 소액 전체 절차 테스트

이 절이 이 보고서에서 가장 중요하다. 새로 만든 지갑에 목돈을 바로 보내지 않는다. 먼저 소액으로 지갑 운영의 전 과정을 처음부터 끝까지 돌려 보고, 전부 정상임을 확인한 다음에 목돈을 옮긴다. 복구까지 해 보기 전에는 백업이 아니고, 송금까지 해 보기 전에는 쓸 준비가 된 지갑이 아니다.

FIGURE 11 **목표 금액을 옮기기 전에 반드시 거치는 소액 전체 절차 테스트**

이 여섯 단계를 모두 통과한 뒤에야 본 자금을 이전한다.

1단계	2단계	3단계	4단계	5단계	6단계
키 생성 오프라인 환경	**백업 기록 금속에 각인**	**소액 수신 기기 화면에서 주소 확인**	**소액 송금 잔돈 주소와 수수료 확인**	**기기 초기화 백업만으로 복구**	**재송금 확인 복구된 지갑에서**

FIGURE 11 **목표 금액을 옮기기 전에 반드시 거치는 소액 전체 절차 테스트 (계속)**

통과 기준

6단계까지 통과 후 목표 금액 이전. 5단계를 건너뛰면 백업이 실제로 작동하는지 확인하지 못한 것이므로 테스트한 것이 아니다.

이 절차는 처음 한 번으로 끝내지 말고 기기나 펌웨어를 교체했을 때, 그리고 최소 연 1회 반복한다.

특히 5단계를 건너뛰면 안 된다. 원본 기기가 살아 있는 상태에서 하는 확인으로는 백업이 진짜 작동하는지 알 수 없다. 기기를 초기화하거나 다른 기기를 써서, 오직 보관 중인 백업만으로 지갑이 살아나는지 봐야 의미가 있다. 시드 단어를 잘못 적었거나, 패스프레이즈를 잘못 기억했거나, 멀티시그 구성 정보가 없거나, 파생 경로가 안 맞는 문제는 대부분 여기서 드러난다. 목돈을 넣은 뒤에 발견하면 손쓰기 어렵지만, 소액 단계에서 발견하면 고치면 그만이다.

이 테스트는 한 번으로 끝내지 않는다. 기기나 펌웨어를 바꿨을 때, 그리고 적어도 1년에 한 번은 다시 해서 백업이 여전히 살아 있는지 확인한다.

인공지능을 검증 보조 도구로 쓰는 법

인공지능을 검증에 쓰려면 전제부터 확실히 하자. 널리 쓰이는 인공지능 서비스는 인터넷 너머에 있는 남의 시스템이고, 입력한 내용은 보내는 순간 내 손을 떠난다. 대화 기록이 지워진다는 보장도 없다. 2025년에는 공유한 대화가 검색 엔진에 노출된 일이 있었고, 소송을 위해서 사용자가 삭제한 대화까지 보존하라는 요구를 받은 사례도 있었다.[30] 그래도 요즘 대규모 언어모델(Large Language Model, LLM)의 프로그래밍 실력은 눈에 띄게 늘어서, 최상위 모델은 셀프 커스터디의 여러 단계에서 실질적인 도움이 된다. 내 지갑 데이터는 넣지 않으면서 공개된 정보, 개념 질문, 민감한 값을 뺀 계획만 주면 인공지능은 꽤 쓸 만한 검증 파트너가 된다.

FIGURE 12 **셀프 커스터디에서 AI를 쓸 수 있는 곳과 쓰면 안 되는 곳**

기준은 하나다. AI는 내 지갑에 관한 데이터를 보지 않는다. 공개 정보와 개념 질문만 다룬다.

● 적합: 내 지갑 데이터가 없는 작업	● 금지: 내 지갑 데이터가 담긴 작업
· 모르는 개념과 용어 설명 · 공개된 오픈소스 펌웨어 코드 검토 · 제조사 공식 문서와 취약점 공고 해설 · 식별자를 뺀 절차서 검토와 위협 모델링 · 내가 오프라인에서 실행할 검증 도구 작성 · 복구 테스트 순서의 누락 점검	· 시드 · 니모닉 · 개인키 · 패스프레이즈 · 확장공개키(xpub)와 지갑 디스크립터 · 서명 전후 거래 데이터(PSBT) 원문과 해석 출력 · 내 주소 목록 · 보유 수량 · 백업 보관 위치 · 펌웨어 내려받기 주소 질의 · 주소 파생 · 키 계산 등 암호 연산 위임

가장 중요한 경고

시드 · 개인키 · 패스프레이즈는 일부만 가려서, 예시로 바꿔서도 입력하지 않는다. 확장공개키와 거래 데이터도 같다.
프론티어 AI는 인터넷에 연결된 외부 서비스이며, 입력한 내용이 사라진다는 보장이 없다.

방식 1. 이미 공개된 정보만 다룬다.

모르는 용어의 뜻, 파생 경로 표기가 무엇을 의미하는지, 재현 가능 빌드(공개된 코드로 똑같은 펌웨어를 다시 만들어 배포본과 맞춰 보는 방식)가 어떤 원리인지 같은 질문은 내 지갑과 무관한 일반 지식이라 마음껏 물어도 된다. 내가 쓰는 지갑의 공개된 펌웨어 코드를 같이 읽으며 취약점을 살펴보는 것도 가능하다. 제조사 공식 문서나 취약점 공지를 풀어 설명하게 하는 것도 같은 부류다. 실제로 콜드카드 사고 뒤 인공지능을 쓴 분석으로 여러 지갑에서 문제가 더 발견됐다는 보고가 있다.[31]

방식 2. 데이터를 주는 대신 도구를 만들게 한다.

가장 실속 있는 방식이다. 검증할 데이터를 인공지능에게 보내는 대신, 그 데이터를 검사하는 작은 프로그램(스크립트)을 짜 달라고 한 뒤 오프라인 기기에서 내가 직접 돌린다. 인공지능은 코드만 만들고 데이터는 못 보니, 지갑 정보가 내 기기 밖으로 나가지 않는다. 이 방식으로 할 수 있는 일 몇 가지를 소개한다. 다만 실제 자산에 쓰는 니모닉·개인키·패스프레이즈·확장개인키는 인공지능이 만든 스크립트에 넣지 않고, 해당 작업은 실제 지갑과 무관한 연습용 값으로만 수행한다.

- 연습용으로 주사위를 던져 만든 난수를 니모닉으로 변환하는 스크립트
- 거래 초안(PSBT)의 수취인 주소, 수량, 수수료가 사용자의 의도에 부합하는지 확인을 돕는 스크립트
- 확장공개키에서 주소를 파생해 하드웨어 지갑 화면에 표시된 주소와 대조하는 스크립트
- 내려받은 펌웨어의 해시를 계산해 제조사가 공개한 값과 비교하고 결과만 알려주는 스크립트
- 실제 자산과 무관한 연습용 니모닉의 체크섬이 올바른지 오프라인에서 확인하는 스크립트
- 공개키만 포함된 멀티시그 디스크립터를 읽어 공동 서명자 구성과 파생 경로를 사람이 읽기 쉬운 형태로 정리하는 스크립트

조심할 것이 세 가지 있다. 첫째, 받은 스크립트는 돌리기 전에 꼭 읽고 이해한다. 특히 데이터를 밖으로 보내는 코드가 있는지 본다. 네트워크 요청, 파일 업로드, 외부 서비스 호출이 있다면 그것이 곧 유출 통로다. 무엇을 봐야 할지 모르겠으면 같은 인공지능에게 이 코드에서 네트워크에 접근하는 부분을 전부 짚어 달라고 요청한다. 둘째, 실행은 네트워크를 끊은 기기에서 한다. 셋째, 스크립트가 요구하는 외부 라이브러리는 설치 전에 실재하는지 확인한다.

방식 3. 식별 정보를 뺀 설계와 절차를 검토하게 한다.

내 구성을 구체적인 식별 정보 없이 설명하고 검토를 부탁할 수도 있다. 예를 들어 서로 다른 세 제조사 기기로 2-of-3을 짰고, 세 곳에 나눠 보관하며, 디스크립터는 어디에 백업했다는 식으로, 주소나 확장공개키 없이 뼈대만 적는 것이다. 이러면 어떤 순서로 뚫릴 수 있는지, 빠진 단계는 없는지 안심하고 물을 수 있다.

묻는 방식도 중요하다. 내 구성이 안전하냐고 묻지 말고, 이 구성에서 어떤 공격이 가능하냐고 묻자. 앞의 질문은 듣기 좋은 답을 끌어내기 쉽다. 5.6절의 복구 테스트 순서를 설명하고 빠진 단계를 짚어 달라고 하는 것도 좋은 예다.

지갑 데이터를 반드시 분석해야 한다면

위 세 방식으로 해결되지 않아 실제 지갑 데이터를 꼭 분석해야 한다면, 인터넷이 끊긴 내 컴퓨터에서 가중치가 공개된 오픈소스 모델을 직접 돌리는 방법이 있다. 데이터가 외부로 전송되지 않으니 위험의 성격이 다르다.

그래도 안심할 방법은 아니다. 모델을 돌린 기기에 입력 내용이 기록으로 남을 수 있고, 그 기기가 나중에 인터넷에 연결되는 순간 그 기록이 노출 대상이 된다. 이 방법을 쓴다면 그 기기는 다른 용도로 쓰지 말고, 검증이 끝나면 기록을 지우거나 초기화하는 편이 낫다. 서명에 쓰는 기기와도 분리한다. 성능이 낮은 모델일수록 틀린 분석을 그럴듯하게 내놓는다는 점도 기억해 두자.

절대 입력하면 안 되는 정보

넣지 말아야 할 정보는 두 부류다. 하나는 넣는 순간 돈을 잃을 수 있는 정보, 다른 하나는 돈이 바로 빠져나가지는 않지만 한번 새면 되돌릴 수 없는 정보다.

입력 즉시 자산을 잃을 수 있는 정보

여기에는 니모닉(시드 구문), 개인키, 패스프레이즈, 확장개인키가 해당한다. 개인키나 확장개인키가 포함된 지갑 디스크립터도 같은 비밀정보다. 이런 정보는 일부를 가렸다고 해서 안전해지지 않는다. 특히 니모닉은 몇 단어만 다른 단어로 바꾼 상태로도 입력하지 않는다. 12단어 니모닉은 단어의 위치가 알려진 상태에서 아홉 단어가 노출되면 남은 세 단어의 조합을 컴퓨터로 모두 대입해 볼 수 있는 범위로 좁아진다. 시드가 찍힌 사진이나 백업 파일을 올리는 것도 마찬가지로 위험하다.

되돌릴 수 없는 프라이버시 손실을 낳는 정보

여기에는 확장공개키, 공개키만 포함된 지갑 디스크립터, 서명 전 거래 데이터의 원문과 해석 결과, 주소 목록, 보유 수량, 백업 보관 위치가 해당한다. 이 정보들만으로 자산을 직접 옮길 수는 없지만, 종류에 따라 잔액과 거래 내역, 신원이나 보관 위치가 드러나 물리적 위협의 표적이 될 수 있다. 한번 노출된 거래 이력과 신원의 연결은 주소를 바꿔도 되돌리기 어렵다.

FIGURE 13 **인공지능을 검증 보조 도구로 쓸 때의 프롬프트 예시**

기준은 하나다. 인공지능에는 내 지갑에 관한 데이터를 넣지 않고, 공개 정보와 개념 질문, 그리고 내가 직접 실행할 도구만 요청한다.

● 적합 · 개념 확인

비트코인 지갑의 파생 경로 m/48'/0'/0'/2' 에서 각 숫자가 무엇을 뜻하는지 설명해줘. 그리고 이 경로가 단일서명이 아니라 멀티시그에 쓰이는 이유도 알려줘.

왜 괜찮은가 내 지갑에 관한 값이 하나도 들어가지 않는 일반 지식 질문이다. 공개된 표준을 묻는 것이므로 유출될 것이 없다.

● 적합 · 검증 도구 작성 (가장 실용적)

PSBT 파일을 입력받아 다음을 점검하는 파이썬 스크립트를 작성해줘.
1) 모든 출력의 BIP32 파생 경로가 서로 일관되는지
2) 잔돈으로 표시된 출력이 실제로 내 계정 경로에 속하는지
3) 입력 합계와 출력 합계의 차이가 의도한 수수료와 맞는지
조건: 네트워크에 접속하는 코드는 넣지 마. 표준 라이브러리와 널리 검증된 오픈소스만 사용하고, 나는 이 스크립트를 오프라인 기기에서 직접 실행할 거야.

왜 괜찮은가 PSBT 자체는 입력하지 않고 그것을 읽는 도구만 요청한다. 데이터는 내 기기 밖으로 나가지 않는다.

● 적합 · 받은 코드의 교차 확인

방금 준 스크립트에서 데이터를 외부로 내보내는 부분이 있는지 전부 표시해줘. 네트워크 요청, 파일 업로드, 외부 API 호출, 원격 로그 전송을 포함해서 해당 줄을 하나씩 짚어줘.

왜 괜찮은가 실행 전에 반드시 거치는 단계다. 스스로 코드를 읽되, 놓친 부분이 없는지 같은 도구로 교차 확인한다.

● 적합 · 구성과 절차 검토

다음 구성에서 어떤 공격이 가능한지 알려줘. 서로 다른 세 제조사의 에어갭 기기로 2-of-3 멀티시그를 구성하고, 키는 집과 은행 대여금고와 가족의 집 세 곳에 나눠 두며, 디스크립터는 그중 두 곳에 사본을 둔다. 주소나 확장공개키는 알려주지 않겠다.

왜 괜찮은가 안전한지 묻지 않고 어떤 공격이 가능한지 묻는다. 앞의 질문은 동의하는 답을 끌어내기 쉽다.

● 금지 · 지갑 데이터를 그대로 입력

내 PSBT인데 이상한 점이 있는지 분석해줘.
cHNidP8BAHECAAAAAfuJ2K1cQ9pR… (이하 생략)
내 지갑이 안전한지 최종적으로 판단해줘. 확장공개키는 xpub6BfhisZn1pCf8WkEo4k9Y5… 야.

왜 위험한가 PSBT에는 확장공개키와 마스터 지문, 파생 경로가 함께 담긴다. 한 번 넘어가면 잔액과 거래 내역 전체가 드러나고 주소를 바꾸어도 회수되지 않는다. 해석 명령의 출력을 붙여넣는 것도 같다.

유출과 분실에 대비한 대응 계획

일이 틀어졌을 때 어떻게 움직일지를 평소에 글로 정해 둔다. 하드웨어 지갑 한 대를 잃어버리거나 도둑맞았다면, 패스프레이즈 구성은 곧바로 새 지갑으로 전액을 옮기고, 멀티시그는 뚫린 키만 바꾼 새 구성으로 자산을 옮긴다. 시드나 패스프레이즈가 새 나갔을 가능성이 조금이라도 있으면, 그 지갑은 이미 유출된 것으로 전제하고 움직인다.

제조사가 취약점을 발표했을 때의 대응도 미리 정해 둔다. 1장의 사고에서 봤듯, 펌웨어를 업데이트해도 이미 만들어진 시드는 고쳐지지 않는다. 취약한 시기에 만든 시드라면 새 시드를 만들어 자산을 옮기는 것 말고는 답이 없다. 평상시에 안전한 대피 지갑을 준비하고 5.6절의 소액 전체 절차 테스트를 마쳐 둔다. 공격이 진행 중이거나 시드 유출이 의심되면 검증된 대피 지갑의 주소를 확인하고 신속히 자산을 옮긴다. 긴급 상황에서는 전체 복구 시험을 끝내기 위해 위험한 지갑에 자산을 계속 두지 않는다.

마지막으로 상속과 비상 접근도 설계에 넣는다. 내가 접근할 수 없게 됐을 때 믿을 만한 사람이 자산에 닿을 길이 없다면, 그 구성은 언젠가 자산을 영영 잃는 방향으로 작동한다. 셀프 커스터디의 책임은 내 생애로 끝나지 않는다.

여기까지가 5장의 지침이다. 2장에서 본 사고가 어느 단계에서 났고, 그것을 막는 원칙이 4장의 몇 번이며, 실제 방법이 5장 어디에 있는지를 한 장에 모으면 아래와 같다.

FIGURE 14 **사고가 난 지점, 4장의 원칙, 5장의 지침이 어떻게 이어지는가**

한 열을 위에서 아래로 읽으면 무엇이 무너졌고 무엇으로 막는지가 한 줄로 이어진다.

	1. 생성	2. 백업	3. 보관	4. 서명	5. 복구
2장 사고가 난 지점	**2.1 난수 부족 랜드스톰 밀크새드**	**2.2 기록이 경로 국세청 사진 라스트패스**	**2.3 위치가 표적 실크로드 비트파이넥스**	**2.4 화면은 근거 아님 잔돈 주소 조작 논스 재사용**	**2.5 미검증 백업 조용한 영구 손실**
4장 개인이 지킬 원칙	원칙 2	원칙 3	원칙 4	원칙 5	원칙 6
5장 실천 지침	5.1 기기 선택 5.2 생성 환경	5.3 금속 기록	5.4 장소 분산	5.5 네 가지 확인	5.6 소액 테스트

전 단계 공통

2.6 사람과 물리 세계
원칙 1 · 원칙 7 · 원칙 8
5.1 구성 선택 · 5.7 인공지능 활용 · 5.8 대응 계획
단일 실패지점 제거, 독립 수단 검증, 절차 작성과 기록은 전 단계에 적용한다

읽는 법

3장 기관의 교훈 다섯 가지는 이 전체에 걸쳐 있다. 교훈 1은 생성, 교훈 3은 보관, 교훈 2와 4, 5는 구성과 검증 전반에 대응한다.

맺음말

믿을 만한 내부통제와 규제를 갖춘 거래소에 맡길 수 있고 그것으로 충분하다면, 이 문서는 필요 없다. 1장의 두 사고는 사용자가 아무 잘못을 하지 않아도 자산을 잃을 수 있고, 기록하는 방식 하나로 자산이 사라질 수 있다는 것을 보여줬다. 셀프 커스터디는 그 위험을 전부 본인이 떠안는 구조다.

그래도 이 길을 갈 수밖에 없는 사람들이 있다. 이 보고서가 내놓은 것은 모든 위험을 없애는 방법이 아니라, 알려진 실패 지점을 하나씩 지워 나가는 지침이다. 2장에서 봤듯 지금까지의 사고는 비트코인의 암호가 뚫려서 일어난 것이 아니었다. 난수가 모자랐거나, 기록이 새 나갔거나, 키가 엉뚱한 곳에 있었거나, 확인을 건너뛰었기 때문이었다. 뒤집어 말하면, 개인이 챙겨야 할 영역이 생각보다 넓다는 뜻이다.

마지막으로 한 번 더 강조할 것은 소액 전체 절차 테스트다. 평상시에 새 지갑을 준비할 때는 어떤 구성을 골랐든 목돈을 바로 옮기지 말자. 소액으로 생성, 백업, 수신, 송금, 그리고 백업만 가지고 하는 복구까지 전 과정을 완주하고, 복구된 지갑에서 다시 송금되는 것까지 본 다음에 목돈을 옮긴다. 하루도 안 걸리고 돈도 거의 안 들지만, 나중에 발견하면 손쓸 수 없는 문제를 자산이 걸리기 전에 미리 드러내 준다.

셀프 커스터디를 시작하기로 했다면, 한 번 설정하고 끝나는 일이 아니라 계속 가꿔야 하는 절차로 받아들이는 편이 좋다. 기기는 바뀌고, 취약점은 새로 나오고, 백업은 시간이 지나면 상태를 알 수 없게 된다. 1년에 한 번 점검과 복구 연습을 일정에 넣어 두고 나만의 매뉴얼을 조금씩 다듬어 가는 것. 그것이 어떤 값비싼 장비보다 확실한 방어다.

일러두기

본문의 수치는 보도와 온체인 분석, 제조사 공개 자료를 기준으로 하며 시점에 따라 달라질 수 있다. 본 보고서는 정보 제공을 목적으로 하며 투자, 법률, 세무 자문이 아니다.

미주

1. TRM Labs, 「The Largest Hardware Wallet Exploit of 2026: Inside the USD 116 Million Coldcard Hack」, https://www.trmlabs.com/resources/blog/the-largest-hardware-wallet-exploit-of-2026-inside-the-usd-116-million-coldcard-hack. 총액 집계는 주체와 시점에 따라 다르다. 2026년 8월 1일 The Hacker News 보도는 약 1,083 비트코인(약 7,020만 달러), 같은 기사 갱신본은 약 1,367 비트코인(약 8,860만 달러)이었고(https://thehackernews.com/2026/08/coldcard-hardware-wallet-flaw-linked-to.html), 2026년 8월 4일 TechCrunch는 Galaxy Research 집계를 인용해 1억 3,000만 달러를 넘는다고 보도했다(https://techcrunch.com/2026/08/04/hackers-steal-over-130-million-by-exploiting-bug-in-offline-hardware-wallets/). 본문의 달러 환산은 사고 당일인 2026년 7월 30일 시세(비트코인 한 개당 약 6만 4,500달러)를 기준으로 했다. 2026년 8월 18일 접속.

2. Block Engineering, 「Predictable RNG Fallback and 32-Bit Reseed in COLDCARD Firmware」, https://engineering.block.xyz/blog/predictable-rng-fallback-and-32-bit-reseed-in-coldcard-firmware; Coinkite, 「Coldcard Security Advisory」, https://blog.coinkite.com/coldcard-mk3-seed-generation-warning/. 제조사 공지 기준 영향 범위는 Mk2와 Mk3의 4.0.1(2021년 3월)부터 4.1.9까지, Mk4와 Mk5의 5.6.0 미만, Q의 1.5.0Q 미만이다. Block 보고서는 시작 버전을 4.0.0으로 적고 있다. 제조사는 시드 생성 시 50회 이상 공정하게 굴린 주사위로 엔트로피를 직접 넣은 경우에는 이 결함의 영향을 받지 않는다고 밝혔고, 2026년 7월 31일 영향받는 모든 모델의 수정 펌웨어를 배포하면서 취약한 시기에 만든 시드는 새로 만들어 자산을 옮기라고 권고했다. 2026년 8월 18일 접속.

3. 함지현, 「AI가 키운 보안 위협?…콜드카드 사태가 흔든 '셀프 커스터디' 신화(종합)」, 『블록미디어』, 2026년 8월 3일, https://www.blockmedia.co.kr/archives/1123428. 2026년 8월 18일 접속.

4. 『더피알』, 「국세청 보도자료가 부른 파장: 압류 성과 알리려다 코인 유출」, https://www.the-pr.co.kr/news/articleView.html?idxno=60674. PRTG(Pre-Retogeum)는 이더리움 기반 토큰이다. 약 69억 원은 400만 개 기준 개당 약 1,725원에 해당한다. 2026년 8월 18일 접속.

5. 『서울신문』, 「되찾자마자 또 털렸다: 국세청 압류 코인 2차 탈취」, 2026년 3월 2일, https://www.seoul.co.kr/news/society/accident/2026/03/02/20260302500147; 이데일리, 「환수한 400만개 탈취 코인 돌려준 국세청…가치평가 기준은 여전히 숙제」, 2026년 5월 6일, https://marketin.edaily.co.kr/News/ReadE?newsId=06123766645446296. 2026년 8월 18일 접속.

6. Babylon Labs, 「Bitcoin Staking」, https://babylonlabs.io/. 2026년 8월 18일 접속.

7. Unciphered, 「Randstorm: You Can't Patch a House of Cards」, https://www.unciphered.com/disclosure-of-vulnerable-bitcoin-wallet-library-2/. 2026년 8월 18일 접속.

8. Milk Sad 연구팀, 「Milk Sad: /bx full write-up」(CVE-2023-39910), https://milksad.info/disclosure.html. 2026년 8월 18일 접속.

9. Brian Krebs, 「Feds Link $150M Cyberheist to 2022 LastPass Hacks」, Krebs on Security, 2025년 3월, https://krebsonsecurity.com/2025/03/feds-link-150m-cyberheist-to-2022-lastpass-hacks/. 2026년 8월 18일 접속.

10. Kaspersky, 「SparkCat trojan stealer infiltrates App Store and Google Play, steals data from photos」, https://www.kaspersky.com/blog/ios-android-ocr-stealer-sparkcat/52980/. 2026년 8월 18일 접속.

11. Samuel Oakford, 「Prosecutors in Silk Road Trial Present Damning Evidence From Ross Ulbricht's Computer」, 『Vice』, 2015년 1월 21일 , https://www.vice.com/en/article/prosecutors-in-silk-road-trial-present-damning-evidence-from-ross-ulbrichts-computer/; 「Inside the FBI Takedown of the Mastermind Behind Website Offering Drugs, Guns and Murders for Hire」, CBS News, 2020년 11월 10일, https://www.cbsnews.com/news/ross-ulbricht-dread-pirate-roberts-silk-road-fbi/. 2026년 8월 18일 접속.

12. Namcios, 「How Law Enforcement Seized 94,000 Bitcoin Stolen From Bitfinex」, Bitcoin Magazine, 2022년 3월 8일, https://bitcoinmagazine.com/technical/how-authorities-found-bitfinex-bitcoin. 2026년 8월 18일 접속.

13. Bitcoin Project, 「Android Security Vulnerability」, 2013년 8월 11일 , https://bitcoin.org/en/alert/2013-08-11-android; RFC 6979, 「Deterministic Usage of the Digital Signature Algorithm」, https://www.rfc-editor.org/rfc/rfc6979; BIP340, 「Schnorr Signatures for secp256k1」, https://bips.dev/340/. 2026년 8월 18일 접속.

14. Coinkite, 「Troublesome Change Outputs」, http://blog.coinkite.com/troublesome-change/. 2026년 8월 18일 접속.

15. Pascal Gauthier, 「Message by Ledger's CEO: Update on the July Data Breach」, Ledger, 2020년 12월 21일, https://www.ledger.com/message-ledgers-ceo-data-leak. 2026년 8월 18일 접속.

16. NewsBTC, 「Coinbase Uses Electromagnetic Signal Blocking Tents to Secure Crypto」, https://www.newsbtc.com/news/ai-and-tech/coinbase-uses-electromagnetic-signal-blocking-tents-to-secure-cryptocurrencies/. 2026년 8월 18일 접속.

17. Reuters, 「El Salvador to transfer Bitcoin reserves to multiple addresses」, 2025년 8월 29일 , https://www.reuters.com/world/americas/el-salvador-transfer-bitcoin-reserves-multiple-addresses-2025-08-29/. 2026년 8월 18일 접속.

18. Crypto Valley Association Cybersecurity Working Group, 「Trusted Key Ceremony Guidelines」, 2020년 7월 , https://d1c2gz5q23tkk0.cloudfront.net/assets/uploads/2956620/asset/CVA_Trusted_Key_Ceremony_Guidelines.pdf. 2026년 8월 18일 접속.

19. 관계부처 합동, 「공공분야 가상자산 보유·관리체계 개선방안」, 2026년 4월 10일, https://www.korea.kr/news/policyNewsView.do?newsId=148962458. 국세청의 사과와 대외 공개 자료 사전 심의 도입은 『서울신문』, 2026년 3월 1일, https://www.seoul.co.kr/news/economy/2026/03/01/20260301500075; 감사원의 네 기관 압수물 관리 실태 점검은 『헤럴드경제』, 2026년 3월 4일, https://biz.heraldcorp.com/article/10686765 참고. 2026년 8월 18일 접속.

20. CryptoCurrency Certification Consortium(C4), 「CryptoCurrency Security Standard」, https://cryptoconsortium.org/standards-2/; NIST CMVP, 「FIPS 140-3 Standards」, https://csrc.nist.gov/projects/cryptographic-module-validation-program/fips-140-3-standards; AICPA, 「SOC 1」, https://www.aicpa-cima.com/topic/audit-assurance/audit-and-assurance-greater-than-soc-1; AICPA, 「SOC 2」, https://www.aicpa-cima.com/topic/audit-assurance/audit-and-assurance-greater-than-soc-2/. 2026년 8월 18일 접속.

21. 여덟 원칙과 2장 사고, 3장 교훈, 5장 지침의 대응 관계는 다음과 같다. 원칙 1은 2.5 복구 단계의 조용한 사고와 3.2 교훈 2, 3.3 교훈 3에서 나와 5.1 구성의 선택, 5.3 샤미르 백업과 시드 XOR, 5.4 보관 장소 분산으로 이어진다. 원칙 2는 1.1 콜드카드 난수 결함과 2.1 생성 단계, 3.1 교훈 1에서 나와 5.1 기기 선택, 5.2 생성 단계로 이어진다. 원칙 3은 1.2 국세청 보도자료와 2.2 백업 단계에서 나와 5.3 백업 단계, 5.7 인공지능에 입력하면 안 되는 정보로 이어진다. 원칙 4는 2.3 보관 단계와 2.6 렛저 고객정보 유출, 3.3 교훈 3에서 나와 5.4 보관 단계, 5.8 대응 계획으로 이어진다. 원칙 5는 2.4 서명 단계에서 나와 5.5 서명 단계의 네 가지 확인, 5.1 기기 선택 기준 중 결정적 논스로 이어진다. 원칙 6은 2.1 콜드카드 사고의 뒤늦은 발견과 2.5 복구 단계, 3.5 교훈 5에서 나와 5.6 소액 전체 절차 테스트와 맺음말로 이어진다. 원칙 7은 2.4 화면 신뢰 문제와 3.4 교훈 4, 3.5 교훈 5에서 나와 5.5 주소 재계산, 5.7 인공지능 활용, 5.8 대응 계획으로 이어진다. 원칙 8은 3.4 교훈 4에서 나와 5.8 대응 계획으로 이어진다. 이 대응 관계를 키 수명주기 축으로 그린 것이 5장 끝의 그림 14다.

22. The Bitcoin Hole, 「Hardware Wallets」 비교 데이터베이스, https://thebitcoinhole.com/hardware-wallets. 원본 데이터셋은 https://github.com/thebitcoinhole/database. 2026년 8월 18일 접속.

23. Coinkite, 「Coldcard Security Advisory」, https://blog.coinkite.com/coldcard-mk3-seed-generation-warning/; Coinkite, 「COLDCARD Security Update: Seed Generation, Transaction Integrity, and Data Isolation」, 2026년 8월 20일, https://blog.coinkite.com/coldcard-security-update-5.6.1-1.5.1q/. 보고서 기준일인 2026년 8월 30일 현재 표준 트랙의 권고 버전은 Mk4·Mk5 5.6.1과 Q 1.5.1Q였으며, Mk2·Mk3의 수정 버전은 4.2.0이다. Edge 트랙에는 별도의 버전 체계가 적용된다. 2026년 8월 30일 접속.

24. BIP84, 「Derivation scheme for P2WPKH based accounts」, https://bips.dev/84/; BIP48, 「Multi-Script Hierarchy for Multi-Sig Wallets」, https://bips.dev/48/. 2026년 9월 5일 접속.

25. Jameson Lopp, 「Metal Bitcoin Seed Storage Stress Test (Round IV)」, 2020년 8월 2일, https://blog.lopp.net/metal-bitcoin-seed-storage-stress-test-iv/. 2026년 8월 18일 접속.

26. SatoshiLabs, 「SLIP-0039: Shamir's Secret-Sharing for Mnemonic Codes」, https://github.com/satoshilabs/slips/blob/master/slip-0039.md. 2026년 8월 18일 접속.

27. Coldcard Firmware Docs, 「Seed XOR」, https://github.com/Coldcard/firmware/blob/master/docs/seed-xor.md. 2026년 8월 18일 접속.

28. Unchained, 「How to Store a Bitcoin Seed Phrase in a Safe Deposit Box」, https://www.unchained.com/blog/how-to-store-seed-phrase-safe-deposit-box; FDIC, 「Five Things to Know About Safe Deposit Boxes, Home Safes and Your Valuables」, https://www.fdic.gov/consumer-resource-center/five-things-know-about-safe-deposit-boxes-home-safes-and-your-valuables. 금융기관별로 약관은 다르므로 이용 전 확인이 필요하다. 2026년 8월 18일 접속.

29. Bitcoin Core, 「deriveaddresses」 RPC 문서, https://bitcoincore.org/en/doc/25.0.0/rpc/util/deriveaddresses/; Bitcoin Core, 「HWI」, https://github.com/bitcoin-core/HWI. 2026년 8월 18일 접속.

30. Sonya Moisset, 「When 'Private' Isn't: The Security Risk of GPT Chats Leaking to Search Engines」, Snyk, 2025년 8월 1일, https://snyk.io/blog/chatgpt-chat-google/; OpenAI, 「How We're Responding to The New York Times' Data Demands in Order to Protect User Privacy」, 2025년 6월 5일, https://openai.com/index/response-to-nyt-data-demands/. 2026년 8월 18일 접속.

31. Rony Roy, 「Coldcard's RNG Flaw Is Still Draining Wallets, and an AI Audit Just Found 85 More Critical Bugs Across Bitcoin」, crypto.news, 2026년 8월 7일, https://crypto.news/coldcard-rng-flaw-bitcoin-wallet-ai-audit/. 2026년 8월 18일 접속.

비트코인 셀프 커스터디 가이드

사고 사례와 모범 사례에서 도출한 셀프 커스터디 지침

Researcher

이지환 Jeehwan Lee jeehwanlee27@gmail.com

Corresponding Author

오태민 Taemin Oh

Editor

박상현 Sanghyun Park