24 slov: čo je to kryptografická identita

Kryptografická identita nie je heslo: neukladá ju žiaden server a nedá sa obnoviť. Didaktické vysvetlenie mechanizmu BIP39, prečo presne dvadsaťštyri slov a aká reálna váha spočíva na tom, kto ich vlastní.

Aby sme si rozumeli: Ak zabudnete heslo ku Gmailu, Google vám ho resetuje. Ak stratíte 24 slov, ktoré tvoria kryptografickú identitu, nemáte koho požiadať o ich vrátenie. Nejde o to, že by bol postup prísny — ide o to, že na druhom konci nikto nie je. Tento rozdiel je úplne zásadný.

Rozdiel medzi heslom a identitou

Heslo v klasickom modeli internetu nie je identitou používateľa. Je to doklad. Používateľ má identitu — meno, e-mail, zákaznícke číslo — a aby serveru dokázal, že je tým, za koho sa vydáva, predloží heslo, ktoré server porovná s odtlačkom, ktorý má uložený. Ak sa odtlačky zhodujú, server povolí reláciu. Ak sa heslo stratí, používateľ zostáva rovnakým používateľom; to, čo stráca, je doklad, a existuje postup na obnovu — e-mail na registrovanú adresu, bezpečnostná otázka — ako ho získať späť.

Kryptografická identita funguje inak. Nie je to poverenie, ktoré by niekto porovnával s uloženým odtlačkom; *je to* úplné matematické tajomstvo samo o sebe. Je jedno, kde sa nachádza — na papieri, v zariadení, alebo dokonca na cudzom serveri — identita existuje vďaka svojej matematike, nie vďaka tomu, kto ju overuje. Tu sa objavuje vlastnosť podobná tej, ktorú sme videli v článku «Čo je SHA-256 v skutočnosti»: vlastníctvo sa nedokazuje predložením tajomstva, ale jeho použitím na podpísanie. Takto vytvorený podpis môže ktokoľvek overiť pomocou verejnej hodnoty, ktorá je matematicky odvodená zo samotného tajomstva, bez toho, aby tajomstvo musel poznať a bez sprostredkovania tretej strany pri overovaní. Kto má tajomstvo, je identitou; kto ho stratí, prestáva ňou byť. Rozsudok je kategorický: **neexistuje nikto, koho by ste mohli požiadať o vrátenie identity. Taký niekto neexistuje, pretože ju v prvom rade vôbec nemal.**

Čo predstavuje dvadsaťštyri slov

Kryptografická identita je zvyčajne reprezentovaná matematickým tajomstvom s dĺžkou tridsaťdva bajtov — dvestopäťdesiatšesť bitov. Číslo, ktoré je ťažké si zapamätať a ešte ťažšie ho bezchybne opísať. Kryptografický priemysel vyriešil tento problém v roku 2013 malým a elegantným štandardom zvaným BIP39: spôsobom, ako reprezentovať týchto dvestopäťdesiatšesť bitov ako sekvenciu dvadsaťštyri slov vybraných z oficiálneho zoznamu dvetisícštyridsaťosem slov. Aritmetika v pozadí do seba elegantne zapadá; kto ju chce vidieť podrobne, nájde ju v poznámke na okraji.

Počítanie začína od konca. Chceme reprezentovať dvestopäťdesiatšesť bitov tajomstva a pridať osem bitov kontrolného súčtu: celkovo dvestošesťdesiatštyri bitov. Ak ich rozdelíme do dvadsaťštyri slov — čo je zvládnuteľný počet pre zápis aj diktovanie bezo strát — musí každé slovo niesť presne jedenásť bitov informácie. A jedenásť bitov je dva na jedenástu možností, teda dvetisícštyridsaťosem. Preto má oficiálny slovník BIP39 práve túto veľkosť: zoznam existuje na mieru problému, nie naopak.

Počítanie nie je dekoratívne. Ak niekto opíše dvadsaťtri slov správne a v dvadsiatom štvrtom sa pomýli, kontrolný súčet to zistí: softvér mu povie „táto sekvencia nie je platná“. Ak niekto opíše všetkých dvadsaťštyri

slov správne, softvér jednoznačne odvodí rovnakú identitu. Voľba zoznamu slov je tiež zámerná: slová zo slovníka BIP39 sú krátke, vzájomne odlišné, bez diakritiky, zvolené tak, aby sa minimalizovali fonetické a pravopisné zámeny. Je to slovník navrhnutý tak, aby si ho ľudia zapamätali, zapísali a nadiktovali bezo strát.

Od frázy ku kľúču

Tých dvadsaťštyri slov nie je kryptografický kľúč, ktorým sa podpisujú správy. Sú obnoviteľnou reprezentáciou pôvodnej entropie, ktorá sa pomocou deterministického procesu nazývaného PBKDF2 transformuje na šesťdesiatštyribajtový seed. Z tohto seedu sa takisto deterministicky odvodzujú konkrétne kryptografické kľúče, ktoré používateľ používa: súkromný kľúč na podpisovanie a zodpovedajúci verejný kľúč, ktorý sa zverejňuje na overovanie podpisov. Rovnaký mechanizmus v rôznych systémoch: kryptomeny používajú krivku secp256k1; protokol Signal a mnoho moderných systémov používajú Ed25519 na krivke Curve25519. Pre konkrétnu krivku, ako je Ed25519, berú štandardy BIP32 a SLIP-0010 onen šesťdesiatštyribajtový seed a deterministicky odvodzujú tridsaťdva bajtov, ktoré tvoria efektívny podpisový kľúč — tých istých tridsaťdva bajtov, ktorými začína príklad kódu v nasledujúcej časti.

Toto je štandardný spôsob, akým celý priemysel prezentuje mechanizmus používateľovi —kryptomenové peňaženky, správcovia decentralizovanej identity, Signal vo svojej časti pre trvalú identitu, Solo2 medzi nimi—: používateľ v praxi nikdy nevidí seed ani odvodené kľúče. Vidí oných dvadsaťštyri slov pri vytváraní svojej identity a voliteľne si ich zapíše na papier. Slová potom cestujú medzi jeho zariadeniami, keď chce identitu migrovať: zadá ich do novej aplikácie, aplikácia odvodí rovnaký seed, rovnaké kľúče, rovnakú identitu. Je to prenosný, kryptograficky solídny a v medziach rozumného zapamätateľný mechanizmus.

Ako sa podpisuje kľúčom (náčrt v Zig)

V Zig, hneď ako máte tridsaťdvabajtový seed odvodený z dvadsiatich štyroch slov, podpísanie správy pomocou Ed25519 sa zmestí na pár riadkov:

```
const std = @import("std");
const Ed25519 = std.crypto.sign.Ed25519;

// 'semilla' son los 32 bytes derivados de las 24 palabras.
const par = Ed25519.KeyPair.create(semilla);

// Firmar un mensaje con la clave privada:
const mensaje = "Este mensaje lo escribí yo.";
const firma = try par.sign(mensaje, null);

// Cualquiera con la clave pública del par puede verificar:
try Ed25519.Signature.verify(firma, mensaje, par.public_key);
```

Operácia podpisovania vytvorí šesťdesiatštyri bajtov —nazývaných podpis— ktoré mohli byť vygenerované iba z zodpovedajúceho súkromného kľúča. Overenie je verejné: ktokoľvek s verejným kľúčom môže skontrolovať, či podpis zodpovedá správe. Bez súkromného kľúča nemôže nikto vytvoriť platný podpis pre danú správu; s verejným kľúčom môže ktokoľvek zistiť, či je podpis platný. Táto asymetria je to, čo umožňuje podpisujúcemu preukázať autorstvo bez zdieľania tajomstva.

Predchádzajúci príklad je minimálna verzia príručky. V skutočnom kóde Solo2 reťazec prechádza dvoma súbormi, jedným v JavaScript, ktorý žije v prehliadači používateľa a rekonštruuje entrópiu z dvadsiatich štyroch slov, druhým v Zig v rámci knižnice *zcatcrypto*, ktorý túto entrópiu preberá a odvodzuje konkrétne kryptografické kľúče. Počnúc stranou prehliadača:

```
// solo2/web-app/js/lib/bip39.js
async function mnemonicToEntropy(mnemonic, lang) {
    const validation = await validateMnemonic(mnemonic, lang);
    if (!validation.valid) {
        return { entropy: null, valid: false, error: validation.error };
    }
    const wordlist = WORDLISTS[lang || 'en'];
    const words = mnemonic.trim().split(/\s+/);

    // Cada palabra aporta 11 bits (su índice en la lista de 2048).
    let bits = '';
    for (let i = 0; i < words.length; i++) {
        bits += wordlist.indexOf(words[i]).toString(2).padStart(11, '0');
    }

    // 24 palabras = 264 bits. Los primeros 256 son la entropía.
    const entropyBytes = new Uint8Array(32);
    for (let j = 0; j < 32; j++) {
        entropyBytes[j] = parseInt(bits.slice(j * 8, (j + 1) * 8), 2);
    }
    return { entropy: entropyBytes, valid: true };
}
```

Týchto tridsaťdva bajtov entrópie spolu s ďalšími tridsiatimi dvoma odvodenými v rovnakom kroku putuje do modulu WebAssembly v Zig, ktorý generuje samotné kľúče Ed25519. Kompletná funkcia s konečným vyčistením pamäte sa zmestí na jednu obrazovku:

```
// zcatcrypto/wasm/bindings/identity.zig
const Ed25519 = std.crypto.sign.Ed25519;
const X25519  = std.crypto.dh.X25519;

export fn identity_generate() ?*IdentityHandle {
    var seed: [64]u8 = undefined;
    if (!common.getRandomBytes(&seed)) return null;

    const handle = common.wasm_allocator.create(IdentityHandle) catch return null;

    // Bytes 0..31: semilla determinista del par Ed25519 (firma).
    const sign_kp = Ed25519.KeyPair.generateDeterministic(seed[0..32].*) catch {
        common.wasm_allocator.destroy(handle);
        return null;
    };
    handle.sign_secret = sign_kp.secret_key.toBytes();
    handle.sign_public = sign_kp.public_key.toBytes();

    // Bytes 32..63: secreto X25519 (para acordar claves de cifrado con el otro).
    handle.exchange_secret = seed[32..64].*;
    handle.exchange_public = X25519.recoverPublicKey(handle.exchange_secret) catch {
        common.wasm_allocator.destroy(handle);
        return null;
    };

    @memset(&seed, 0);  // Borra la semilla de la memoria.
```

```
    return handle;
}
```

Za zmienku stoja dva detaily. Prvý: ten istý seed vždy produkuje ten istý pár kľúčov — práve to umožňuje obnovu identity zadaním dvadsiatich štyroch slov do nového zariadenia. Druhý: seed sa v poslednom riadku explicitne vymaže z pamäte. Po tomto bode by ani samotná funkcia nemohla kľúče rekonštruovať; slová používateľa by boli jediným zdrojom.

Pre tých, ktorí si to chcú overiť na malých číslach. Schému podpisu možno prejsť celú s číslami dostatočne malými na to, aby sa výpočty dali robiť ručne. Kto radšej nechce zachádzať do aritmetiky, môže tento blok vynechať bez straty nite článku; kto chce vidieť mechanizmus fungujúci krok za krokom, nájde ho tu. **Verejné pravidlá**, ktoré si môže prečítať ktokoľvek: prvočíslo $p = 23$ (v skutočnom Ed25519 má asi sedemdesiatsedem číslic; používame dvadsaťtri, aby sa výpočty zmestili na jednu stranu), základ $g = 2$, ktorého rád v tejto grupe je $q = 11$, a konvencia, že všetka aritmetika s g sa vykonáva *módulo p* a všetky exponenty sa redukujú *módulo q*. **Súkromná voľba**, jediná a nikdy nezdieľaná: tajomstvo $x = 6$. To je identita.

Krok 1 — Verejná časť identity. Vypočíta sa raz a otvorene sa zverejní.

$$y = g^x \bmod p$$

$$y = 2^6 \bmod 23 = 64 \bmod 23 = 18$$

Verejná časť identity je **18**. Ktokoľvek ju môže vziať a použiť na overenie podpisov vytvorených s touto identitou. Nikto, pozorujúc iba 18, nemôže obnoviť tajomstvo 6: to je problém diskrétneho logaritmu, ku ktorému sa vrátime na konci.

Krok 2 — Podpísanie správy. Držiteľ identity chce podpísať správu $m = 7$. Začne výberom novej náhodnej hodnoty $k = 4$, ktorá sa použije iba raz a nikdy sa nebude zdieľať (v skutočnom Ed25519 sa k odvodzuje deterministicky zo správy a tajomstva, aby sa predišlo nebezpečenstvu jeho opätovného použitia, ale rola, ktorú hrá, je presne táto). Potom vypočíta tri čísla:

$$r = g^k \bmod p = 2^4 \bmod 23 = 16$$

$$e = H(r, m) \bmod q = (16 + 7) \bmod 11 = 1$$

$$s = (k + x \cdot e) \bmod q = (4 + 6 \cdot 1) \bmod 11 = 10$$

Podpis je dvojica **(r, s) = (16, 10)**. Putuje otvorene spolu so správou. Ktokoľvek ho môže prečítať. Didaktická poznámka: v skutočnom Ed25519 je funkcia H SHA-512, kryptograficky robustná; tu používame zjednodušenie $e = (r + m) \bmod q$, aby si čitateľ mohol prejsť kroky bez potreby výpočtu hashu. Štruktúra algoritmu je rovnaká.

Krok 3 — Overenie podpisu. Overovateľ má verejnú časť $y = 18$, správu $m = 7$ a podpis $(r, s) = (16, 10)$. Rekonštruuje e rovnakým spôsobom — $e = (16 + 7) \bmod 11 = 1$ — a skontroluje, či táto rovnosť platí:

$$g^s \bmod p \stackrel{?}{=} r \cdot y^e \bmod p$$

Vypočíta obe strany zvlášť:

Izquierda: $2^{10} \bmod 23 = 1024 \bmod 23 = 12$

Derecha: $16 \cdot 18^1 \bmod 23 = 288 \bmod 23 = 12$

Obe strany dávajú **12**. Podpis je platný. Ktokoľvek s verejnou časťou 18 môže prísť k tomuto záveru bez toho, aby kedy vedel, že tajomstvo bolo 6.

A čo tretia strana, ktorá by sa pokúsila o falšovanie? Eva videla všetko verejné prechádzajúce kanálom: $p = 23$, $g = 2$, $q = 11$, $y = 18$, $m = 7$, $r = 16$, $s = 10$. Aby mohla podpísať *inú* správu v mene tejto identity, musela by poznať x. Jej jedinou cestou je opýtať sa samej seba: „pre aký exponent x platí $2^x \bmod 23 = 18$?“. S $p = 23$ môže skúsiť 0, 1, 2, 3, … a nájsť ho v priebehu niekoľkých sekúnd. Ale pri nahradení 23 prvočíslom reálnych rozmerov Ed25519 priestor možných exponentov presahuje počet atómov v pozorovateľnom vesmíre. **Dnes neexistuje žiadny ľudstvu známy algoritmus, ktorý by dokázal prejsť tento priestor za menej ako miliardy rokov.** Je to ten istý problém diskrétneho logaritmu, na ktorom stavia Diffie-Hellman v predchádzajúcom článku, aplikovaný tu na schému podpisu.

To, čo sme práve prešli, je *presne* Schnorr, schéma podpisu, ktorej je Ed25519 variantom prispôsobeným eliptickej krivke. V skutočnom Ed25519 sa všetky operácie vykonávajú nad bodmi konkrétnej krivky (Curve25519) namiesto celých čísel modulo prvočíslo a funkcia H je SHA-512 namiesto nami použitého zjednodušeného súčtu. Obe substitúcie sú úpravy implementácie — získanie kryptografickej odolnosti voči hrubej sile, získanie ďalších bezpečnostných vlastností pre k. Algoritmická štruktúra, tri operácie a dôvod asymetrie sú rovnaké.

Tu je vhodné sa krátko zastaviť, pretože celý reťazec si možno pri bežnom pohľade spliesť s iným primitívom z trojice: hashom. Nie je ním. Hash je unikátna funkcia, ktorá komprimuje — vstupuje mnoho bajtov, vystupuje krátky odtlačok a tam cesta končí. Kryptografická identita je matematicky doplnková dvojica: tajomstvo zostáva a podpisuje; jeho verejný protipól sa zverejňuje a overuje. Zatiaľ čo hash kolabuje informácie v jednom smere, identita vytvára asymetriu medzi dvoma polovicami. Hash potvrdzuje, čo bolo povedané; identita potvrdzuje, kto to povedal.

Čím fráza nie je

Je vhodné vyjasniť tri časté omyly. Fráza nie je heslo v pravom zmysle slova: neporovnáva sa s odtlačkom uloženým na serveri; zadáva sa do zariadenia používateľa za účelom matematickej rekonštrukcie identity. Fráza sa neobnovuje: ak sa stratí, nie je nikto, koho by ste o ňu mohli požiadať; ak sa duplikuje, duplikuje sa aj identita. Fráza nie je poverenie oddeliteľné od identity: fráza *je* identita. Kto ju má, môže za ňu konať, bez ďalšieho povolenia, bez procesu autorizácie, bez možnosti obnovy.

Práve táto tretia vlastnosť mení váhu celej veci. Stratené heslo je administratívna nepríjemnosť. Stratená kryptografická identita je identita sama. Papier s frázou nájdený treťou stranou nie je riziko krádeže účtu: je to odovzdanie celej identity. Prísľub systému — aby vám nikto nemohol identitu odobrať alebo vás svojvoľne zablokovať — je neoddeliteľne sprevádzaný zodpovednosťou — že vy ste jediným strážcom niečoho, čo za vás nikto nemôže obnoviť.

Prísľub a váha

Model kryptografickej identity sa zvyčajne označuje ako *samosuverénna* —self-sovereign v anglosaskej literatúre—. Voľba slova je zámerná a popisuje stav dosť presne. Používateľ je suverénom nad svojou identitou v takmer stredovekom zmysle: neudeľuje ju žiadny kráľ, žiadny vydavateľ, žiadna centrálna autorita; ani ju žiadny z vyššie uvedených nemôže odobrať. Ale rovnako ako stredoveký monarcha nesie používateľ všetky následky svojich chýb: neexistuje žiadny regent, ktorý by rozhodoval za neho, ak stratí pečať.

Voľba medzi identitou spravovanou treťou stranou a samosuverénnou identitou nemá jedinú univerzálne správnu odpoveď. Pre účet na nepodstatnom fóre je spravovaná identita pravdepodobne úmerná riziku. Pre profesijnú identitu, ktorá podpisuje právne záväzné dokumenty, pre ekonomickú identitu, ktorá stráži vlastné úspory, pre identitu profesijnej komunikácie s klientmi, ktorí zverili citlivé informácie, sa situácia mení. Tam otázka prestáva byť „je to pohodlné?“ a stáva sa „kto okrem mňa má moc konať mojím menom a za akých okolností?“.

Kde sa tento mechanizmus objavuje v reálnych systémoch

BIP39 sa zrodil vo svete Bitcoin v roku 2013 a rýchlo sa rozšíril do celého kryptomenového ekosystému: každá seriózna peňaženka dnes prijíma dvanásť- alebo dvadsaťštyrislovnú frázu BIP39 ako zálohu ekonomickej identity svojho držiteľa. Mimo kryptomien sa rovnaký základný koncept — kryptografický pár preukazujúci autorstvo bez sprostredkovateľa — objavuje v iných systémoch s odlišnou syntaxou. SSH kľúče, ktoré správca systému používa na prístup k svojim serverom, sú klasickým prípadom: súkromný kľúč, ktorý si správca ukladá na svojom stroji, a verejný, ktorý sa kopíruje na každý server; nezasahuje žiadny subjekt porovnateľný s centralizovanou službou. Protokol Signal používa Ed25519 s perzistentným materiálom kľúča v zariadení; európske eIDAS sa vo svojej časti o kvalifikovanom podpise opierajú o rovnaký kryptografický princíp s tým rozdielom, že kľúč opatruje kvalifikovaný poskytovateľ dôveryhodných služieb namiesto užívateľa.

Solo2, vydavateľská platforma tejto publikácie, používa dvadsaťštyrislovnú frázu BIP39 ako identitu každého užívateľa. Užívateľ pri vytváraní svojho účtu vidí slová raz. Neukladajú sa na žiadnom serveri Solo2 ani nikoho iného: ak si ich užívateľ poznamená a opatruje, uchová si svoju identitu navždy. Ak ich stratí, stratí ich. Je to logický dôsledok architektúry bez operátora uprostred: keby Solo2 mohla vrátiť identitu užívateľovi, ktorý ju stratil, mohla by ju tiež dať komukoľvek, kto na Solo2 zatlačí, aby ju vydala.

Pre profesionálneho čitateľa

Štyri úvahy pre tých, ktorí zvažujú prijatie kryptografickej samovrstennej (autosoberana) identity v profesionálnom kontexte:

1. Fráza je identita. Fyzické opatrovanie — papier, niekoľko kópií na rôznych miestach, prípadne kov s gravírovaním pre dlhodobé použitie — ponúka viac záruk než digitálne opatrovanie, ktoré zväčšuje útočnú plochu bez toho, aby znižovalo riziko straty.
2. Neexistuje žiadna obnova. Navrhnúť proces za predpokladu, že jedného dňa dôjde k strate primárnej kópie, je oveľa vhodnejšie než to zistiť v deň, keď k strate dôjde. Druhá geograficky oddelená kópia rieši takmer všetky scenáre.
3. Nie je to to isté čo kvalifikovaný certifikát eIDAS. Pre kvalifikovaný podpis v Únii — notárske zápisnice, určité úkony s úradmi — legislatíva vyžaduje kvalifikovaného poskytovateľa, ktorý kľúč opatruje. Kryptografická samovrstenná identita slúži pre profesionálnu komunikáciu a podpisovanie dokumentov s dôkaznou hodnotou, ale nenahrádza automaticky kvalifikovaný certifikát v prípadoch, keď to norma vyžaduje.
4. Ak má byť identita prevedená — dedičstvo, profesijné nástupníctvo, ukončenie činnosti — je vhodné pripraviť postup vopred, nie až potom. Formálne postupy s obálkami zapečatenými pečatným voskom (lacre), inštrukcie pre vykonávateľa závetu, uloženie u notára, sú klasické dojednania dokonale zlučiteľné s kryptografickou povahou aktíva.

Tento článok uzatvára konceptuálne trio, ktoré cyklus otvorilo — hash, šifrovanie, identita —. Tieto tri myšlienky sa stavajú jedna na druhej: hash dáva nemenný odtlačok, šifrovanie dáva dôvernosť bez dôveryhodnej tretej strany, identita dáva autorstvo bez poskytujúcej tretej strany. Všetky tri zdieľajú vlastnosť, ktorá tiež nie je ideologická: prenášajú od toho, kto spravuje službu, na toho, kto ju používa, technické možnosti, ktoré tradične spočívali na operátorovi. Spolu s nimi prenášajú aj zodpovednosť. Hovoriť čestne o ktorejkoľvek z týchto troch vyžaduje hovoriť aj o zvyšných dvoch.

Zdroje a ďalšie čítanie

- Palatinus, M.; Rusnak, P.; Voisine, A.; Bowe, S. — *BIP-0039: Mnemonic code for generating deterministic keys*, návrh na vylepšenie Bitcoin z roku 2013. De facto štandard pre frázy pre obnovu v kryptopriemysle.
- RFC 8032 — Edwards-Curve Digital Signature Algorithm (EdDSA), vrátane Ed25519. IETF, január 2017. Normatívna špecifikácia schémy podpisu používanej vo veľkej časti súčasného priemyslu.

- RFC 2898 — PKCS #5: Password-Based Cryptography Specification, verzia 2.0. IETF, september 2000. Definuje algoritmus PBKDF2 použitý v derivácii BIP39 z frázy na seed.
- Nariadenie (EÚ) 910/2014 (eIDAS) a jeho vývoj nariadením (EÚ) 2024/1183 (eIDAS 2) — európsky rámec pre elektronickú identitu a kvalifikovaný podpis. Iný režim než samovrstenný, ale konceptuálne sa opierajúci o rovnaké kryptografické primitíva.
- Allen, C. — *The Path to Self-Sovereign Identity* (2016). Kanonický text o princípoch a záväzkoch samovrstenného modelu, starší, ale relevantný pre pochopenie rodiny súčasných riešení.

← PredchádzajúciObchodný model ako signál dôveryNasledujúci →Self-hosting ako profesionálna prax

Nedávne čítanie

- Úvaha · 29. júna 2026 Nie si anonymný
- Úvaha · 27. mája 2026 To, čo podpis nemôže vyriešiť
- Analýza · 26. mája 2026 Skutočné vs. zdanlivé súkromie: otázky, ktoré je vhodné si položiť

Vezmite si tento článok tam, kam potrebujete.

↓ Markdown ↓ Čistý text ↓ PDF

Súbor sa stiahne do vášho zariadenia. Odtiaľ si ho môžete uložiť, importovať do Solo2 alebo zdieľať, kdekoľvek chcete. Cuadernos za vás o cieli nerozhoduje.

Vosková pečať · SHA-256 4cb049921afad58eb7090afc722b9ae5db557cfd94e298b47beded91ad0839ac

Cuadernos Lacre · Publikácia spoločnosti Menzuri Gestión S.L. ·
napísal R.Eugenio · edituje tím Solo2.