Koncept · 23. svibnja 2026.

← Prethodno·Sljedeće →

24 riječi: što je to kriptografski identitet

Kriptografski identitet nije lozinka: niti jedan ga poslužitelj ne pohranjuje i ne može se vratiti. Didaktičko objašnjenje BIP39 mehanizma, zašto točno dvadeset i četiri riječi i kakva stvarna težina pada na onoga tko ih posjeduje.

Da se razumijemo: Ako zaboravite lozinku za Gmail, Google će vam je poništiti. Ako izgubite 24 riječi koje čine kriptografski identitet, nemate ih od koga tražiti. Nije stvar u tome da je procedura stroga — stvar je u tome da na drugoj strani nema nikoga. Ta razlika je ključna.

Razlika između lozinke i identiteta

Lozinka, u klasičnom modelu interneta, nije identitet korisnika. Ona je dokaz. Korisnik ima identitet — ime, e-mail, broj klijenta — i kako bi poslužitelju dokazao da je onaj za kojeg se izdaje, prilaže lozinku koju poslužitelj uspoređuje s pohranjenim otiskom. Ako se otisci podudaraju, poslužitelj odobrava sesiju. Ako se lozinka izgubi, korisnik ostaje isti korisnik; ono što gubi je dokaz, a postoji postupak oporavka — e-mail na registriranu adresu, sigurnosno pitanje — kako bi se on ponovno uspostavio.

Kriptografski identitet funkcionira drugačije. To nije vjerodajnica koju netko uspoređuje s pohranjenim otiskom; to *jest* potpuna matematička tajna sama po sebi. Nije važno gdje se nalazi — na papiru, u uređaju, čak i na tuđem poslužitelju — identitet postoji zbog svoje matematike, a ne zbog onoga tko ga potvrđuje. Ovdje se pojavljuje svojstvo slično onome koje smo vidjeli u «Što je zapravo SHA-256»: posjedovanje se ne dokazuje pokazivanjem tajne, već njezinim korištenjem za potpisivanje. Tako proizveden potpis svatko može provjeriti pomoću javne vrijednosti koja je matematički izvedena iz same tajne, bez potrebe za poznavanjem same tajne i bez posredovanja treće strane u provjeri. Onaj tko ima tajnu, jest identitet; onaj tko je izgubi, prestaje to biti. Presuda je kategorična: **nema nikoga koga biste mogli zamoliti da vam vrati identitet. Ta osoba ne postoji jer ga uopće nije ni imala.**

Što predstavlja dvadeset i četiri riječi

Kriptografski identitet obično se predstavlja matematičkom tajnom od trideset dva bajta — dvjesto pedeset i šest bita. Broj koji je teško upamtiti, a još teže prepisati bez greške. Kriptografska industrija riješila je ovaj problem 2013. godine malim i elegantnim standardom zvanim BIP39: način predstavljanja tih dvjesto pedeset i šest bita kao niza od dvadeset i četiri riječi preuzetih sa službene liste od dvije tisuće četrdeset i osam riječi. Aritmetika u pozadini uklapa se s elegancijom; onaj tko je želi vidjeti u detalje, naći će je na margini.

Brojanje počinje od kraja. Želimo predstaviti dvjesto pedeset i šest bita tajne dodavanjem osam bita kontrolnog zbroja (checksum): ukupno dvjesto šezdeset i četiri bita. Ako ih rasporedimo u dvadeset i četiri riječi — broj kojim je lako upravljati pri zapisivanju i diktiranju bez gubitaka — svaka riječ mora doprinijeti s točno jedanaest bita informacija. A jedanaest bita je dvije na jedanaestu potenciju mogućnosti, odnosno dvije tisuće četrdeset i osam. Otuda službeni BIP39 vokabular ima upravo tu veličinu: lista postoji prema mjeri problema, a ne obrnuto.

Brojanje nije dekorativno. Ako netko prepiše dvadeset i tri riječi točno, a pogriješi u dvadeset i četvrtoj, kontrolni zbroj će to otkriti: softver će mu reći "ovaj niz nije valjan". Ako netko prepiše sve dvadeset i četiri

riječi točno, softver će nedvosmisleno izvesti isti identitet. Izbor popisa riječi također je namjeran: riječi BIP39 vokabulara su kratke, međusobno različite, bez dijakritičkih znakova, odabrane kako bi se minimizirale fonetske i pravopisne zabune. To je vokabular dizajniran da ga ljudi pamte, zapisuju i diktiraju bez gubitaka.

Od fraze do ključa

Dvadeset i četiri riječi nisu kriptografski ključ koji potpisuje poruke. One su povratni prikaz izvorne entropije koja se deterministčkim procesom zvanim PBKDF2 transformira u sjeme (seed) od šezdeset i četiri bajta. Iz tog sjemena također se deterministički izvode konkretni kriptografski ključevi koje korisnik koristi: privatni ključ za potpisivanje i odgovarajući javni ključ koji se objavljuje za provjeru potpisa. Isti mehanizam u različitim sustavima: kriptovalute koriste krivulju secp256k1; Signal protokol i mnogi moderni sustavi koriste Ed25519 na krivulji Curve25519. Za konkretnu krivulju poput Ed25519 standardi BIP32 i SLIP-0010 uzimaju to sjeme od šezdeset i četiri bajta i deterministički izvode trideset i dva bajta koji čine učinkovit ključ za potpisivanje — istih onih trideset i dva bajta s kojima počinje primjer koda u sljedećem odjeljku.

Ovo je standardni način na koji cijela industrija predstavlja mehanizam korisniku —novčanici za kriptovalute, upravitelji decentraliziranog identiteta, Signal u svom dijelu za trajni identitet, Solo2 među njima—: korisnik u praksi nikada ne vidi sjeme niti izvedene ključeve. Vidi dvadeset i četiri riječi prilikom stvaranja svog identiteta i, po želji, zapisuje ih na papir. Riječi zatim putuju između njegovih uređaja kada želi migrirati identitet: unosi ih u novu aplikaciju, aplikacija izvodi isto sjeme, iste ključeve, isti identitet. To je prijenosan, kriptografski solidan i, u granicama razumnog, pamtljiv mehanizam.

Kako se potpisuje ključem (potez kistom u Zig-u)

U Zig-u, nakon što imate sjeme od trideset i dva bajta izvedeno iz dvadeset i četiri riječi, potpisivanje poruke pomoću Ed25519 stane u nekoliko redaka:

```
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);
```

Operacija potpisivanja proizvodi šezdeset i četiri bajta —zvanih potpis— koji su se mogli generirati samo iz odgovarajućeg privatnog ključa. Provjera je javna: svatko s javnim ključem može provjeriti odgovara li potpis poruci. Bez privatnog ključa nitko ne može proizvesti važeći potpis za tu poruku; s javnim ključem svatko može otkriti je li potpis važeći. Ta asimetrija je ono što omogućuje potpisniku da dokaže autorstvo bez dijeljenja tajne.

Prethodni primjer je minimalna verzija priručnika. U stvarnom Solo2 kodu, lanac prolazi kroz dvije datoteke, jednu u JavaScriptu koja živi u pregledniku korisnika i rekonstruira entropiju iz dvadeset i četiri riječi, drugu u Zigu unutar *zcatcrypto* knjižnice koja uzima tu entropiju i izvodi konkretne kriptografske ključeve. Počevši od strane preglednika:

```
// 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 };
}
```

Tih trideset i dva bajta entropije, zajedno s još trideset i dva izvedena u istom koraku, putuju do Zig WebAssembly modula koji generira same Ed25519 ključeve. Kompletna funkcija, sa svojim konačnim čišćenjem memorije, stane na jedan ekran:

```
// 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;
}
```

Vrijedi istaknuti dva detalja. Prvi: isti seed uvijek proizvodi isti par ključeva — upravo to omogućuje oporavak identiteta unosom dvadeset i četiri riječi u novi uređaj. Drugi: seed se eksplicitno briše iz memorije u zadnjem retku. Nakon te točke, čak ni sama funkcija ne bi mogla rekonstruirati ključeve; korisnikove riječi bile bi jedini izvor.

Za one koji to žele provjeriti malim brojevima. Shema potpisa može se u potpunosti proći s brojkama dovoljno malim da se izračuni obave ručno. Oni koji ne žele ulaziti u aritmetiku mogu preskočiti ovaj blok bez gubljenja niti članka; oni koji žele vidjeti mehanizam kako radi korak po korak, naći će ga ovdje. **Javna pravila**, koja svatko može pročitati: prost broj $p = 23$ (u stvarnom Ed25519 on ima oko sedamdeset i sedam znamenki; koristimo dvadeset i tri kako bi izračuni stali na jednu stranicu), baza $g = 2$ čiji je red u ovoj grupi $q = 11$, te konvencija da se sva aritmetika s g izvodi *módulo p* i svi se eksponenti reduciraju *módulo q*. **Privatni izbor**, jedan jedini i nikada dijeljen: tajna $x = 6$. To je identitet.

Korak 1 — Javni dio identiteta. Izračunava se jednom i otvoreno objavljuje.

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

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

Javni dio identiteta je **18**. Svatko ga može uzeti i koristiti za provjeru potpisa napravljenih s tim identitetom. Nitko, promatrajući samo 18, ne može vratiti tajnu 6: to je problem diskretnog logaritma na koji ćemo se vratiti na kraju.

Korak 2 — Potpisivanje poruke. Vlasnik identiteta želi potpisati poruku $m = 7$. Počinje odabirom nove nasumične vrijednosti $k = 4$, koja će se upotrijebiti samo jednom i nikada se neće dijeliti (u stvarnom Ed25519, k se izvodi deterministički iz poruke i tajne kako bi se izbjegla opasnost od ponovne uporabe, ali uloga koju igra je upravo ova). Zatim izračunava tri broja:

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

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

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

Potpis je par **(r, s) = (16, 10)**. Putuje otvoreno zajedno s porukom. Svatko ga može pročitati. Didaktička napomena: u stvarnom Ed25519 funkcija H je SHA-512, kriptografski robusna; ovdje koristimo pojednostavljenje $e = (r + m) \bmod q$ kako bi čitatelj mogao proći korake bez potrebe za izračunavanjem hasha. Struktura algoritma je ista.

Korak 3 — Provjera potpisa. Provjeritelj ima javni dio $y = 18$, poruku $m = 7$ i potpis $(r, s) = (16, 10)$. Rekonstruira e na isti način — $e = (16 + 7) \bmod 11 = 1$ — i provjerava je li ova jednakost zadovoljena:

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

Izračunava obje strane zasebno:

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

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

Obje strane daju **12**. Potpis je valjan. Svatko s javnim dijelom 18 može doći do ovog zaključka, a da nikada nije znao da je tajna bila 6.

A treća strana koja bi pokušala krivotvoriti? Eva je vidjela sve javno što prolazi kanalom: $p = 23$, $g = 2$, $q = 11$, $y = 18$, $m = 7$, $r = 16$, $s = 10$. Da bi potpisala *drugačiju* poruku u ime ovog identiteta, trebala bi znati x. Njezin jedini put je zapitati se: „za koji eksponent x vrijedi $2^x \bmod 23 = 18$?”. Uz $p = 23$ može isprobati 0, 1, 2,

3, … i pronaći ga u sekundi. Ali zamjenom 23 prostim brojem stvarnih dimenzija Ed25519, prostor mogućih eksponenata premašuje broj atoma u vidljivom svemiru. **Danas čovječanstvu nije poznat nijedan algoritam koji bi mogao proći taj prostor u manje od milijardi godina.** To je isti problem diskretnog logaritma koji podupire Diffie-Hellman iz prethodnog članka, primijenjen ovdje na shemu potpisa.

Ovo što smo upravo prošli je *točno* Schnorr, shema potpisa čija je Ed25519 varijanta prilagođena eliptičkoj krivulji. U stvarnom Ed25519, sve se operacije izvode na točkama određene krivulje (Curve25519) umjesto na cijelim brojevima modulo prost broj, a funkcija *H* je SHA-512 umjesto pojednostavljenog zbroja koji smo gore koristili. Dvije zamjene su prilagodbe implementacije — dobivanje kriptografske otpornosti na brute force napade, dobivanje dodatnih sigurnosnih svojstava za *k*. Algoritamska struktura, tri operacije, razlog asimetrije, isti su.

Ovdje je prikladan kratki zastoj, jer se cijeli lanac na brz pogled može zamijeniti s drugom primitivom iz trija: hashom. On to nije. Hash je jedinstvena funkcija koja komprimira — ulazi mnogo bajtova, izlazi kratki otisak, tu put završava. Kriptografski identitet je matematički komplementaran par: tajna ostaje i potpisuje; njezin javni pandan se objavljuje i provjerava. Tamo gdje hash urušava informacije u jednom smjeru, identitet uspostavlja asimetriju između dvije polovice. Hash svjedoči o tome što je rečeno; identitet svjedoči o tome tko je to rekao.

Što fraza nije

Potrebno je razjasniti tri česte zablude. Fraza nije lozinka u pravom smislu: ne uspoređuje se s otiskom pohranjenim na poslužitelju; unosi se u korisnikov uređaj kako bi se matematički rekonstruirao identitet. Fraza se ne vraća: ako se izgubi, nema je koga pitati; ako se duplicira, duplicira se i identitet. Fraza nije vjerodajnica odvojiva od identiteta: fraza *jest* identitet. Tko je posjeduje može djelovati kao taj identitet, bez dodatnog dopuštenja, bez postupka autorizacije, bez mogućnosti oporavka.

Upravo to treće svojstvo mijenja težinu cijele stvari. Izgubljena lozinka je administrativna neugodnost. Izgubljeni kriptografski identitet je sam identitet. Papir s frazom koji pronađu treće strane nije rizik od krađe računa: to je predaja cijelog identiteta. Obećanje sustava — da vam nitko ne može opozvati identitet niti vas samovoljno blokirati — neraskidivo je popraćeno odgovornošću — da ste vi jedini čuvar nečega što nitko ne može obnoviti umjesto vas.

Obećanje i težina

Model kriptografskog identiteta obično se naziva *samosuverena* —self-sovereign u anglosaksonskoj literaturi—. Izbor riječi je namjeran i prilično točno opisuje stanje. Korisnik je suveren nad svojim identitetom u gotovo srednjovjekovnom smislu: ne dodjeljuje ga nijedan kralj, nijedan izdavatelj, nijedno središnje tijelo; niti ga itko od navedenih može povući. Ali također, poput srednjovjekovnog monarha, korisnik snosi punu posljedicu svojih pogrešaka: nema regenta koji bi donosio odluke umjesto njega ako izgubi pečat.

Izbor između identiteta kojim upravlja treća strana i samosuverenog identiteta nema jedinstven točan odgovor. Za korisnički račun na nevažnom forumu, upravljani identitet vjerojatno je razmjeran riziku. Za profesionalni identitet koji potpisuje pravno obvezujuće dokumente, za ekonomski identitet koji čuva vlastitu štednju, za identitet profesionalne komunikacije s klijentima koji su povjerili osjetljive informacije, pitanje se mijenja. Tamo pitanje prestaje biti „je li to zgodno?” i postaje „tko, osim mene, ima moć djelovati kao ja i pod kojim okolnostima?”.

Gdje se ovaj mehanizam pojavljuje u stvarnim sustavima

BIP39 je rođen u svijetu Bitcoina 2013. godine i brzo se proširio na cijeli ekosustav kriptovaluta: svaki ozbiljan novčanik danas prihvaća BIP39 frazu od dvanaest ili dvadeset i četiri riječi kao sigurnosnu kopiju ekonomske osobnosti svog vlasnika. Izvan kriptovaluta, isti temeljni koncept — kriptografski par koji dokazuje autorstvo

bez posrednika — pojavljuje se u drugim sustavima s drugačijom sintaksom. SSH ključevi koje administrator sustava koristi za pristup svojim poslužiteljima klasičan su slučaj: privatni ključ koji administrator čuva na svom računalu i javni koji se kopira na svaki poslužitelj; ne intervenira nikakav subjekt usporediv s centraliziranom uslugom. Protokol Signal koristi Ed25519 s perzistentnim materijalom ključa na uređaju; europski eIDAS se u svom dijelu o kvalificiranom potpisu oslanjaju na isto kriptografsko načelo, s tom razlikom što ključ čuva kvalificirani pružatelj usluga povjerenja umjesto korisnika.

Solo2, izdavačka platforma ove publikacije, koristi BIP39 frazu od dvadeset i četiri riječi kao osobnost svakog korisnika. Korisnik prilikom kreiranja svog računa vidi riječi jednom. One se ne pohranjuju ni na jednom poslužitelju Solo2 niti bilo koga drugoga: ako ih korisnik zabilježi i čuva, zadržava svoju osobnost zauvijek. Ako ih izgubi, izgubio ih je. To je logična posljedica arhitekture bez posrednika: kada bi Solo2 mogao vratiti osobnost korisniku koji ju je izgubio, mogao bi je dati i bilo kome tko pritisne Solo2 da mu je da.

Za profesionalnog čitatelja

Četiri razmatranja za one koji procjenjuju usvajanje kriptografske samovlasne (autosoberana) osobnosti u profesionalnom kontekstu:

1. Fraza je osobnost. Fizičko čuvanje — papir, nekoliko kopija na različitim mjestima, na kraju ugravirani metal za dugotrajnu upotrebu — nudi više jamstava od digitalnog čuvanja, koje povećava površinu napada bez smanjenja rizika od gubitka.
2. Nema oporavka. Dizajniranje procesa uz pretpostavku da će jednog dana primarna kopija biti izgubljena mnogo je razumnije nego to otkriti na dan gubitka. Druga geografski odvojena kopija rješava gotovo sve scenarije.
3. To nije isto što i eIDAS kvalificirani certifikat. Za kvalificirani potpis u Uniji — javnobilježnički akti, određeni postupci s upravom — zakonodavstvo zahtijeva kvalificiranog pružatelja koji čuva ključ. Kriptografska samovlasna osobnost služi za profesionalnu komunikaciju i dokumentarno potpisivanje s dokaznom vrijednošću, ali ne zamjenjuje automatski kvalificirani certifikat u slučajevima kada to norma zahtijeva.
4. Ako će se osobnost prenositi — nasljeđivanje, profesionalno nasljeđivanje, prestanak djelatnosti — preporučljivo je pripremiti postupak prije, a ne poslije. Formalni postupci s omotnicama zapečaćenim pečatnim voskom (lacre), upute izvršitelju oporuke, polog kod javnog bilježnika, klasični su aranžmani savršeno kompatibilni s kriptografskom prirodom imovine.

Ovaj članak zatvara konceptualni trio koji je otvorio ciklus — hash, šifriranje, osobnost —. Tri ideje grade se jedna na drugoj: hash daje nepromjenjiv otisak, šifriranje daje povjerljivost bez povjerljive treće strane, osobnost daje autorstvo bez treće strane koja ga dodjeljuje. Sva tri dijele svojstvo koje također nije ideološko: prenose s onoga tko upravlja uslugom na onoga tko je koristi tehničke sposobnosti koje su tradicionalno pripadale operateru. S njima prenose i odgovornosti. Iskren razgovor o bilo kojem od ova tri zahtijeva razgovor i o preostala dva.

Izvori i dodatno štivo

- Palatinus, M.; Rusnak, P.; Voisine, A.; Bowe, S. — *BIP-0039: Mnemonic code for generating deterministic keys*, prijedlog poboljšanja Bitcoina iz 2013. De facto standard za fraze za oporavak u kripto industriji.
- RFC 8032 — Edwards-Curve Digital Signature Algorithm (EdDSA), uključujući Ed25519. IETF, siječanj 2017. Normativna specifikacija sheme potpisa koja se koristi u velikom dijelu suvremene industrije.
- RFC 2898 — PKCS #5: Password-Based Cryptography Specification, verzija 2.0. IETF, rujan 2000. Definira PBKDF2 algoritam koji se koristi u BIP39 izvođenju iz fraze u seed.
- Uredba (EU) 910/2014 (eIDAS) i njezina evolucija Uredbom (EU) 2024/1183 (eIDAS 2) — europski okvir za elektroničku osobnost i kvalificirani potpis. Režim drugačiji od samovlasnog, ali konceptualno

podržan istim kriptografskim primitivima.
- Allen, C. — *The Path to Self-Sovereign Identity* (2016). Kanonski tekst o načelima i obvezama samovlasnog modela, raniji, ali relevantan za razumijevanje obitelji suvremenih rješenja.

← PrethodnoPoslovni model kao signal povjerenjaSljedeće →Self-hosting kao profesionalna praksa

Nedavna čitanja

- Razmišljanje · 29. lipnja 2026. Nisi anoniman
- Razmišljanje · 27. svibnja 2026. Ono što potpis ne može popraviti
- Analiza · 26. svibnja 2026. Stvarna naspram prividne privatnosti: pitanja koja si trebate postaviti

Ponesite ovaj članak sa sobom gdje god vam zatreba.

↓ Markdown ↓ Obični tekst ↓ PDF

Datoteka će se preuzeti na vaš uređaj. Od tamo je možete spremiti, uvesti u Solo2 ili dijeliti gdje god želite. Cuadernos ne odlučuje o odredištu umjesto vas.

Voštani pečat · SHA-256 926683cf7395e50c3586082cd79e74f931959c7478bed62dd891e49e16aa0bf1

Cuadernos Lacre · Publikacija Menzuri Gestión S.L. ·
napisao R.Eugenio · uredio tim Solo2.