24 Kata: Apakah Identiti Kriptografi

Identiti kriptografi bukanlah kata laluan: tiada pelayan yang menyimpannya dan ia tidak boleh dipulihkan. Penjelasan didaktik tentang mekanisme BIP39, mengapa tepat dua puluh empat kata, dan beban sebenar yang dipikul oleh mereka yang memilikinya.

Untuk kita saling memahami: Jika anda terlupa kata laluan Gmail anda, Google akan menetapkannya semula untuk anda. Jika anda kehilangan 24 kata yang membentuk identiti kriptografi, tiada sesiapa yang boleh diminta untuk mendapatkannya semula. Bukannya prosedurnya ketat — masalahnya ialah tiada sesiapa di hujung sana. Perbezaan itulah yang menjadi segalanya.

Perbezaan Antara Kata Laluan dan Identiti

Kata laluan, dalam model internet klasik, bukanlah identiti pengguna. Ia adalah bauca. Pengguna mempunyai identiti — nama, e-mel, nombor pelanggan — dan, untuk membuktikan kepada pelayan bahawa mereka adalah orang yang mereka dakwa, mereka mengemukakan kata laluan yang dibandingkan oleh pelayan dengan jejak yang telah disimpan. Jika jejak tersebut sepadan, pelayan membenarkan sesi tersebut. Jika kata laluan hilang, pengguna tetap menjadi pengguna yang sama; apa yang hilang ialah bauca, dan terdapat prosedur pemulihan — e-mel ke alamat berdaftar, soalan keselamatan — untuk memulihkannya.

Identiti kriptografi berfungsi dengan cara yang berbeza. Ia bukan kredensial yang dibandingkan oleh seseorang dengan jejak yang disimpan; ia *adalah* rahsia matematik yang lengkap dalam dirinya sendiri. Tidak kira di mana ia berada — pada kertas, dalam peranti, malah pada pelayan asing — identiti itu wujud kerana matematikanya, bukan kerana siapa yang mengesahkannya. Di sini muncul ciri yang serupa dengan yang kita lihat dalam «Apa itu SHA-256 sebenarnya»: pemilikan tidak dibuktikan dengan menunjukkan rahasia, sebaliknya dengan menggunakannya untuk menandatangani. Tandatangan yang dihasilkan dengan cara ini boleh diperiksa oleh sesiapa sahaja dengan nilai awam yang diturunkan secara matematik dari rahsia itu sendiri, tanpa perlu mengetahui rahsia tersebut, dan tanpa pihak ketiga yang menengahi pemeriksaan tersebut. Sesiapa yang mempunyai rahsia tersebut, dialah identitinya; sesiapa yang kehilangannya, berhenti menjadi identiti tersebut. Keputusannya adalah mutlak: **tiada sesiapa yang boleh diminta untuk mengembalikan identiti tersebut kepada anda. Orang itu tidak wujud, kerana mereka tidak memilikinya pada mulanya.**

Apa yang Direpresentasikan oleh Dua Puluh Empat Kata

Identiti kriptografi biasanya direpresentasikan oleh rahsia matematik sebanyak tiga puluh dua bait — dua ratus lima puluh enam bit. Nombor yang sukar diingat dan lebih sukar lagi untuk ditulis tanpa ralat. Industri kriptografi menyelesaikan masalah ini pada tahun 2013 dengan standard kecil dan elegan yang dipanggil BIP39: cara untuk merepresentasikan dua ratus lima puluh enam bit tersebut sebagai urutan dua puluh empat kata yang diambil daripada senarai rasmi dua ribu empat puluh lapan kata. Aritmetik di sebaliknya sangat sesuai; mereka yang ingin melihatnya secara terperinci boleh menemuinya di nota sampingan.

Pengiraan bermula dari akhir. Kita mahu merepresentasikan dua ratus lima puluh enam bit rahsia dengan menambah lapan bit checksum: jumlah dua ratus enam puluh empat bit. Jika kita membahagikannya kepada dua puluh empat kata — jumlah yang boleh diuruskan untuk dicatat dan didekte tanpa kehilangan data — setiap kata

mesti memberikan tepat sebelas bit maklumat. Dan sebelas bit adalah dua kuasa sebelas kemungkinan, iaitu dua ribu empat puluh lapan. Itulah sebabnya kosa kata rasmi BIP39 mempunyai saiz tepat sebanyak itu: senarai tersebut wujud mengikut saiz masalahnya, bukan sebaliknya.

Pengiraan tersebut bukan sekadar hiasan. Jika seseorang menulis dua puluh tiga kata dengan betul dan salah pada kata kedua puluh empat, checksum akan mengesannya: perisian akan memberitahu mereka "urutan ini tidak sah". Jika seseorang menulis kedua-dua puluh empat kata dengan betul, perisian akan menurunkan identiti yang sama tanpa kekaburan. Pemilihan senarai kata juga disengajakan: kata-kata dalam kosa kata BIP39 adalah pendek, berbeza antara satu sama lain, tanpa tanda diakritik, dipilih untuk meminimumkan kekeliruan fonetik dan ejaan. Ia adalah kosa kata yang direka untuk diingat, ditulis, dan didekte oleh manusia tanpa kehilangan data.

Daripada frasa kepada kunci

Dua puluh empat patah perkataan tersebut bukanlah kunci kriptografi yang menandatangani mesej. Ia adalah representasi yang boleh dipulihkan daripada entropi asal yang, melalui proses deterministik yang dipanggil PBKDF2, ditukarkan menjadi benih (seed) enam puluh empat bait. Daripada benih itu diterbitkan, juga secara deterministik, kunci kriptografi konkrit yang digunakan oleh pengguna: kunci peribadi untuk menandatangani dan kunci awam sepadan yang diterbitkan untuk mengesahkan tandatangan. Mekanisme yang sama dalam sistem yang berbeza: mata wang kripto menggunakan lengkung secp256k1; protokol Signal dan banyak sistem moden menggunakan Ed25519 pada lengkung Curve25519. Untuk lengkung konkrit seperti Ed25519, piawaian BIP32 dan SLIP-0010 mengambil benih enam puluh empat bait itu dan menerbitkan secara deterministik tiga puluh dua bait yang membentuk kunci tandatangan berkesan — tiga puluh dua bait yang sama dengan contoh kod dalam bahagian seterusnya bermula.

Ini adalah cara standard di mana seluruh industri membentangkan mekanisme tersebut kepada pengguna — dompet mata wang kripto, pengurus identiti terdesentralisasi, Signal dalam bahagian identiti berterusan mereka, Solo2 antaranya—: pengguna, dalam praktiknya, tidak pernah melihat benih atau kunci terbitan. Dia melihat dua puluh empat patah perkataan semasa mencipta identitinya dan, secara pilihan, mencatatnya pada sehelai kertas. Perkataan tersebut kemudiannya berpindah antara perantinya apabila dia ingin memindahkan identiti: dia memasukkannya ke dalam aplikasi baharu, aplikasi itu menerbitkan benih yang sama, kunci yang sama, identiti yang sama. Ia adalah mekanisme yang mudah alih, kukuh secara kriptografi dan, dalam had yang munasabah, boleh diingat.

Cara menandatangani dengan kunci (sentuhan Zig)

Dalam Zig, sebaik sahaja anda mempunyai benih tiga puluh dua bait yang diterbitkan daripada dua puluh empat patah perkataan, menandatangani mesej dengan Ed25519 hanya memerlukan beberapa baris:

```
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);
```

Operasi menandatangani menghasilkan enam puluh empat bait —dipanggil tandatangan— yang hanya boleh dihasilkan daripada kunci peribadi yang sepadan. Pengesahan adalah awam: sesiapa sahaja yang mempunyai

kunci awam boleh menyemak bahawa tandatangan tersebut sepadan dengan mesej. Tanpa kunci peribadi, tiada siapa yang boleh menghasilkan tandatangan yang sah untuk mesej tersebut; dengan kunci awam, semua orang boleh mengesan jika tandatangan adalah sah. Asimetri inilah yang membolehkan penandatangan membuktikan pemilikan tanpa berkongsi rahsia.

Contoh sebelumnya ialah versi manual minimum. Dalam kod Solo2 yang sebenar, rantaian ini melalui dua fail, satu dalam JavaScript yang berada dalam pelayar pengguna dan membina semula entropi daripada dua puluh empat perkataan, satu lagi dalam Zig dalam pustaka *zcatcrypto* yang mengambil entropi tersebut dan menerbitkan kunci kriptografi khusus. Bermula dari sisi pelayar:

```
// 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 };
}
```

Tiga puluh dua bait entropi itu, bersama-sama dengan tiga puluh dua bait lain yang diterbitkan dalam langkah yang sama, dihantar ke modul WebAssembly Zig yang menghasilkan kunci Ed25519 yang sebenar. Fungsi lengkapnya, dengan pembersihan memori terakhirnya, muat dalam satu skrin:

```
// 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;
}
```

Dua butiran layak dicatatkan. Pertama: seed yang sama sentiasa menghasilkan pasangan kunci yang sama — inilah yang membolehkan pemulihan identiti dengan memasukkan dua puluh empat perkataan ke dalam peranti baharu. Kedua: seed dipadamkan secara eksplisit daripada memori pada baris terakhir. Selepas titik itu, fungsi itu sendiri pun tidak dapat membina semula kunci; perkataan pengguna akan menjadi satu-satunya sumber.

Bagi sesiapa yang ingin menyemaknya dengan nombor kecil. Skema tandatangan boleh ditelusuri sepenuhnya dengan angka yang cukup kecil untuk melakukan pengiraan secara manual. Sesiapa yang memilih untuk tidak masuk ke dalam aritmetik boleh melangkau blok ini tanpa kehilangan alur artikel; sesiapa yang ingin melihat mekanisme berfungsi langkah demi langkah akan menemuinya di sini. **Peraturan awam**, yang boleh dibaca oleh sesiapa sahaja: nombor perdana $p = 23$ (dalam Ed25519 sebenar ia adalah sekitar tujuh puluh tujuh digit; kami menggunakan dua puluh tiga supaya pengiraan muat dalam satu halaman), asas $g = 2$ yang tertibnya dalam kumpulan ini ialah $q = 11$, dan konvensyen bahawa semua aritmetik dengan g dilakukan *módulo p* dan semua eksponen dikurangkan *módulo q*. **Pilihan peribadi**, satu-satunya dan tidak pernah dikongsi: rahsia $x = 6$. Itulah identitinya.

Langkah 1 — Bahagian awam identiti. Ia dikira sekali dan diterbitkan secara terbuka.

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

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

Bahagian awam identiti ialah **18**. Sesiapa sahaja boleh mengambilnya dan menggunakannya untuk mengesahkan tandatangan yang dibuat dengan identiti ini. Tiada sesiapa, dengan hanya memerhati angka 18, dapat memulihkan rahsia 6: itulah masalah logaritma diskret yang akan kita bincangkan semula pada penghujungnya.

Langkah 2 — Menandatangani mesej. Pemegang identiti ingin menandatangani mesej $m = 7$. Dia bermula dengan memilih nilai rawak baharu $k = 4$, yang akan digunakan sekali sahaja dan tidak akan dikongsi (dalam Ed25519 sebenar, k diterbitkan secara deterministik daripada mesej dan rahsia untuk mengelakkan bahaya penggunaan semula, tetapi peranan yang dimainkannya adalah tepat seperti ini). Kemudian dia mengira tiga nombor:

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

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

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

Tandatangannya ialah pasangan **(r, s) = (16, 10)**. Ia dihantar secara terbuka bersama-sama mesej. Sesiapa sahaja boleh membacanya. Nota didaktik: dalam Ed25519 sebenar fungsi H ialah SHA-512, yang teguh secara kriptografi; di sini kami menggunakan pemudahan $e = (r + m) \bmod q$ supaya pembaca dapat mengikuti langkah-langkah tanpa perlu mengira hash. Struktur algoritma adalah sama.

Langkah 3 — Mengesahkan tandatangan. Pengesah mempunyai bahagian awam $y = 18$, mesej $m = 7$, dan tandatangan $(r, s) = (16, 10)$. Dia membina semula e dengan cara yang sama — $e = (16 + 7) \bmod 11 = 1$ — dan

menyemak sama ada persamaan ini dipenuhi:

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

Mengira kedua-dua belah secara berasingan:

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

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

Kedua-dua belah memberikan hasil **12**. Tandatangan adalah sah. Sesiapa sahaja dengan bahagian awam 18 dapat mencapai kesimpulan ini tanpa pernah tahu bahawa rahsianya adalah 6.

Bagaimana pula dengan pihak ketiga yang cuba memalsukan? Eva telah melihat semua yang awam melalui saluran: $p = 23$, $g = 2$, $q = 11$, $y = 18$, $m = 7$, $r = 16$, $s = 10$. Untuk menandatangani mesej yang *berbeza* bagi pihak identiti ini, dia perlu mengetahui x. Satu-satunya caranya ialah dengan bertanya kepada diri sendiri: "untuk eksponen x yang manakah $2^x \bmod 23 = 18$ dipenuhi?". Dengan $p = 23$ dia boleh mencuba 0, 1, 2, 3, … dan menemuinya dalam beberapa saat. Tetapi apabila menggantikan 23 dengan nombor perdana daripada dimensi sebenar Ed25519, ruang eksponen yang mungkin melebihi bilangan atom di alam semesta yang boleh diperhatikan. **Sehingga hari ini tiada algoritma yang diketahui manusia yang dapat menelusuri ruang tersebut dalam masa kurang daripada berbilion tahun.** Ini adalah masalah logaritma diskret yang sama yang mendasari Diffie-Hellman daripada artikel sebelumnya, yang digunakan di sini pada skema tandatangan.

Apa yang baru sahaja kita lalui ialah *tepat* Schnorr, skema tandatangan di mana Ed25519 ialah varian yang diadaptasi kepada lengkung elips. Dalam Ed25519 sebenar, semua operasi dilakukan pada titik-titik lengkung tertentu (Curve25519) dan bukannya pada integer modulo nombor perdana, dan fungsi H ialah SHA-512 berbanding hasil tambah mainan yang kami gunakan di atas. Kedua-dua penggantian tersebut ialah pelarasan pelaksanaan — memperoleh rintangan kriptografi terhadap brute force, memperoleh sifat keselamatan tambahan untuk k. Struktur algoritma, tiga operasi, sebab asimetri, adalah sama.

Sangat sesuai untuk berhenti seketika di sini, kerana keseluruhan rantaian boleh disalah anggap dalam pandangan sepintas lalu dengan primitif lain daripada trio tersebut: hash. Ia bukan itu. Hash ialah fungsi unik yang memampatkan — banyak bait masuk, kesan pendek keluar, di situlah jalannya berakhir. Identiti kriptografi ialah pasangan matematik yang saling melengkapi: rahsia kekal dan menandatangani; rakan awamnya diterbitkan dan mengesahkan. Di mana hash meruntuhkan maklumat dalam satu arah, identiti mewujudkan asimetri antara dua bahagian. Hash membuktikan apa yang dikatakan; identiti membuktikan siapa yang mengatakannya.

Apa yang bukan frasa

Tiga salah faham yang kerap berlaku perlu diperjelaskan. Frasa tersebut bukanlah kata laluan dalam erti kata yang sebenarnya: ia tidak dibandingkan dengan cap jari yang disimpan pada pelayan; ia dimasukkan ke dalam peranti pengguna untuk membina semula identiti secara matematik. Frasa tersebut tidak boleh dipulihkan: jika ia hilang, tiada sesiapa yang boleh diminta untuk mendapatkannya semula; jika ia diduplikasi, identiti juga diduplikasi. Frasa tersebut bukanlah kelayakan yang boleh dipisahkan daripada identiti: frasa tersebut *ialah* identiti. Sesiapa yang memilikinya boleh bertindak sebagai identiti tersebut, tanpa kebenaran tambahan, tanpa proses kebenaran, tanpa kemungkinan pemulihan.

Sifat ketiga inilah yang mengubah berat perkara tersebut. Kata laluan yang hilang adalah kesulitan pentadbiran. Identiti kriptografi yang hilang adalah identiti itu sendiri. Sehelai kertas dengan frasa yang ditemui oleh pihak ketiga bukanlah risiko kecurian akaun: ia adalah penyerahan keseluruhan identiti. Janji sistem —bahawa tiada siapa yang boleh membatalkan identiti anda atau menyekat anda secara sewenang-wenangnya— disertakan secara tidak dapat dipisahkan dengan tanggungjawab —bahawa anda adalah satu-satunya penjaga sesuatu yang tidak boleh dipulihkan oleh sesiapa pun untuk anda.

Janji dan berat

Model identiti kriptografi biasanya menerima gelaran *berdaulat sendiri* —self-sovereign dalam literatur Inggeris —. Pemilihan perkataan ini adalah sengaja dan menggambarkan keadaan dengan agak tepat. Pengguna adalah berdaulat ke atas identitinya dalam erti kata yang hampir seperti zaman pertengahan: ia tidak diberikan oleh mana-mana raja, mana-mana pengeluar, mana-mana pihak berkuasa pusat; dan tiada seorang pun daripada mereka boleh menariknya balik. Tetapi juga, seperti raja zaman pertengahan, pengguna menanggung keseluruhan akibat daripada kesilapannya: tiada pemangku raja yang akan membuat keputusan bagi pihaknya jika dia kehilangan meterai.

Pilihan antara identiti yang diuruskan oleh pihak ketiga dan identiti berdaulat sendiri tidak mempunyai satu jawapan universal yang betul. Untuk akaun forum yang tidak penting, identiti terurus mungkin seimbang dengan risiko. Untuk identiti profesional yang menandatangani dokumen yang mengikat secara undang-undang, untuk identiti ekonomi yang menjaga simpanan sendiri, untuk identiti komunikasi profesional dengan pelanggan yang telah mempercayakan maklumat sensitif, perkara itu berubah. Di sana soalan itu bukan lagi «adakah ia selesa?» dan menjadi «siapa, selain saya, yang mempunyai kuasa untuk bertindak sebagai saya, dan dalam keadaan apa?».

Di mana mekanisme ini muncul dalam sistem sebenar

BIP39 dilahirkan dalam dunia Bitcoin pada tahun 2013 dan tersebar luas ke seluruh ekosistem mata wang kripto dengan pantas: mana-mana dompet yang serius hari ini menerima frasa BIP39 dua belas atau dua puluh empat perkataan sebagai sandaran identiti ekonomi pemegangnya. Di luar mata wang kripto, konsep asas yang sama — pasangan kriptografi yang membuktikan kepengarangan tanpa perantara — muncul dalam sistem lain dengan sintaks yang berbeza. Kekunci SSH yang digunakan oleh pentadbir sistem untuk mengakses pelayan mereka adalah kes klasik: kekunci peribadi yang disimpan oleh pentadbir pada mesin mereka dan kekunci awam yang disalin ke setiap pelayan; tiada entiti yang setanding dengan perkhidmatan berpusat yang campur tangan. Protokol Signal menggunakan Ed25519 dengan bahan kekunci berterusan pada peranti; eIDAS Eropah, dalam bahagian tandatangan berkelayakan, bergantung pada prinsip kriptografi yang sama, dengan perbezaan bahawa kekunci itu disimpan oleh pembekal perkhidmatan amanah berkelayakan dan bukannya pengguna.

Solo2, platform penerbitan penerbitan ini, menggunakan frasa BIP39 dua puluh empat perkataan sebagai identiti setiap pengguna. Pengguna, semasa mencipta akaun mereka, melihat perkataan tersebut sekali. Ia tidak disimpan dalam mana-mana pelayan Solo2 atau milik sesiapa pun: jika pengguna mencatat dan menjaganya, mereka mengekalkan identiti mereka selama-lamanya. Jika mereka kehilangannya, mereka kehilangannya. Ia adalah akibat yang konsisten dengan seni bina tanpa pengendali di tengah: jika Solo2 boleh mengembalikan identiti kepada pengguna yang kehilangannya, ia juga boleh memberikannya kepada sesiapa sahaja yang menekan Solo2 untuk memberikannya.

Untuk pembaca profesional

Empat pertimbangan bagi mereka yang menilai penggunaan identiti kriptografi kedaulatan kendiri (autosoberana) dalam konteks profesional:

1. Frasa tersebut adalah identiti. Jagaan fizikal — kertas, beberapa salinan di tempat yang berbeza, akhirnya logam yang diukir untuk kegunaan jangka panjang — menawarkan lebih banyak jaminan daripada jagaan digital, yang menambah permukaan serangan tanpa mengurangkan risiko kehilangan.
2. Tiada pemulihan. Merangka proses dengan mengandaikan bahawa suatu hari nanti salinan utama akan hilang adalah jauh lebih bijak daripada menemuinya pada hari ia hilang. Salinan kedua yang dipisahkan secara geografi menyelesaikan hampir semua senario.
3. Ia tidak sama dengan sijil berkelayakan eIDAS. Untuk tandatangan berkelayakan dalam Kesatuan — surat ikatan notari, prosedur tertentu dengan Pentadbiran — undang-undang memerlukan pembekal berkelayakan yang menyimpan kekunci tersebut. Identiti kriptografi kedaulatan kendiri berfungsi untuk

komunikasi profesional dan tandatangan dokumentari dengan nilai pembuktian, tetapi tidak menggantikan sijil berkelayakan secara automatik dalam kes di mana norma memerlukannya.

4. Jika identiti akan dipindahkan — warisan, penggantian profesional, penutupan aktiviti — adalah dinasihatkan untuk menyediakan prosedur sebelum ini, bukan selepasnya. Prosedur formal dengan sampul surat yang dimeterai dengan lak (lacre), arahan kepada pelaksana wasiat, deposit di pejabat notari, adalah pengaturan klasik yang sangat serasi dengan sifat kriptografi aset tersebut.

Artikel ini menutup trio konseptual yang membuka kitaran — hash, penyulitan, identiti —. Ketiga-tiga idea tersebut dibina di atas satu sama lain: hash memberikan cap jari yang tidak boleh diubah, penyulitan memberikan kerahsiaan tanpa pihak ketiga yang dipercayai, identiti memberikan kepengarangan tanpa pihak ketiga yang memberikan. Ketiga-tiga mereka berkongsi sifat yang juga bukan ideologi: mereka memindahkan, daripada pihak yang menguruskan perkhidmatan kepada pihak yang menggunakannya, keupayaan teknikal yang secara tradisinya terletak pada pengendali. Mereka juga memindahkan tanggungjawab bersama mereka. Bercakap jujur tentang mana-mana daripada ketiga-tiga ini memerlukan bercakap juga tentang dua yang lain.

Sumber dan bacaan lanjut

- Palatinus, M.; Rusnak, P.; Voisine, A.; Bowe, S. — *BIP-0039: Mnemonic code for generating deterministic keys*, cadangan penambahbaikan Bitcoin pada tahun 2013. Piawaian de facto untuk frasa pemulihan dalam industri kripto.
- RFC 8032 — Edwards-Curve Digital Signature Algorithm (EdDSA), termasuk Ed25519. IETF, Januari 2017. Spesifikasi normatif skema tandatangan yang digunakan dalam sebahagian besar industri kontemporari.
- RFC 2898 — PKCS #5: Password-Based Cryptography Specification, versi 2.0. IETF, September 2000. Menentukan algoritma PBKDF2 yang digunakan dalam terbitan BIP39 daripada frasa kepada seed.
- Peraturan (EU) 910/2014 (eIDAS) dan evolusinya oleh Peraturan (EU) 2024/1183 (eIDAS 2) — rangka kerja Eropah untuk identiti elektronik dan tandatangan berkelayakan. Rejim yang berbeza daripada kedaulatan kendiri, tetapi secara konseptual disokong oleh primitif kriptografi yang sama.
- Allen, C. — *The Path to Self-Sovereign Identity* (2016). Teks kanonik tentang prinsip dan komitmen model kedaulatan kendiri, sebelum ini tetapi relevan untuk memahami keluarga penyelesaian kontemporari.

← SebelumnyaModel perniagaan sebagai isyarat kepercayaanSeterusnya →Self-hosting sebagai amalan profesional

Bacaan terkini

- Refleksi · 29 Jun 2026 Anda bukan tanpa nama
- Refleksi · 27 Mei 2026 Apa yang Tandatangan Tidak Dapat Selesaikan
- Analisis · 26 Mei 2026 Privasi nyata vs semu: soalan yang perlu anda tanya diri sendiri

Bawa artikel ini bersama anda ke mana sahaja anda memerlukannya.

↓ Markdown ↓ Teks biasa ↓ PDF

Fail akan dimuat turun ke peranti anda. Dari sana anda boleh menyimpannya, mengimportnya ke Solo2, atau berkongsinya di mana sahaja anda mahu. Cuadernos tidak memutuskan destinasi untuk anda.

Mohor lilin · SHA-256 139995216b3b18fd210e22c5812619b1645574829a5888655e3d0ed91e44a124

Laman web ini tidak menggunakan kuki. Segala yang dimuatkan oleh pelayar anda ditulis atau diselia oleh kami dan dihoskan pada pelayan Eropah kami: pembilang lawatan tanpa nama (Umami, dihoskan sendiri) dan JavaScript minimum yang diperlukan untuk pemilih bahasa serta pilihan tema terang/gelap anda, yang disimpan pada peranti anda sendiri. Tiada sumber daripada syarikat luar, tiada penjejak, tiada pemprofilan, tiada perkongsian data. Jika anda ingin mengikuti kami: RSS.