Enkripcija od kraja do kraja, stvarno objašnjena

Što pružatelji usluga kažu kada kažu E2EE, a što prešućuju. Didaktičko objašnjenje mehanizma i njegovih ograničenja, bez reklamnog omota.

Da budemo jasni: WhatsApp kaže da su vaše poruke end-to-end kriptirane. To je istina — i to nije dovoljno. Ako sigurnosna kopija ide na iCloud ili Google Drive bez dodatne enkripcije, enkripcija se lomi na vašem vlastitom telefonu. Operativno pitanje nije je li kriptirano, već gdje borave ključevi.

Što enkripcija doista znači

Kriptiranje poruke znači pretvaranje poruke u nešto što izgleda kao šum svakome tko ne posjeduje određenu informaciju zvanu ključ. Operacija se izvodi na uređaju pošiljatelja i, s ispravnim ključem, poništava se na uređaju primatelja. Između toga, poruka putuje kao niz bajtova bez očitog značenja. To je jednostavna ideja. Ostatak članka bavi se nijansama koje je, ovisno o slučaju, pretvaraju u stvarno jamstvo ili u marketinšku oznaku.

Pridjev *od kraja do kraja* — na engleskom *end-to-end*, skraćeno E2EE — dodaje preciznost. Kriptiranje se ne radi kako bi ga posredni poslužitelj mogao pročitati i isporučiti. Radi se tako da samo dva kraja — uređaj pošiljatelja i uređaj primatelja — posjeduju ključ. Svaki poslužitelj kroz koji poruka prolazi vidi šum, a ne poruku. To je tehnička razlika u odnosu na kriptiranje *u tranzitu*, gdje sadržaj putuje kriptiran s jednog poslužitelja na drugi, ali ga svaki poslužitelj kroz koji prolazi dekriptira kako bi ga proslijedio, privremeno vraćajući tekst u čitljiv oblik.

Paradoks zajedničke tajne

Postoji očigledan problem. Kako bi dvije osobe mogle međusobno kriptirati i dekriptirati poruke, objema je potreban isti ključ. Ali, kako se dogovoriti oko tog ključa ako sve što jedna drugoj šalju, po definiciji, prolazi kroz kanal na kojem bi netko mogao slušati? Dogovaranje ključa na istom kanalu na kojem će ga kasnije koristiti čini se nemogućim: ako ga napadač čuje prilikom dogovaranja, moći će dekriptirati sve što slijedi. Desetljećima je klasična kriptografija to rješavala na težak način: ključevi su se predavali osobno, prije početka korištenja, na fizičkim sastancima. Veleposlanici su nosili torbe s ključevima ušivene u podstavu kaputa.

U suvremenoj e-pošti to rješenje nije skalabilno. Kad bismo morali fizički ići u kuću svake osobe s kojom namjeravamo kriptirano komunicirati, ne bismo stigli ni s kim razgovarati. Pitanje koje je kriptografska zajednica postavila prije pedeset godina bilo je ovo: je li moguće da se dvije osobe koje se ne poznaju i koje dijele samo javni kanal dogovore, na tom istom javnom kanalu, o tajni koju nitko tko sluša kanal ne može saznati?

Elegancija Diffie-Hellmana

Godine 1976. dva matematičara po imenu Whitfield Diffie i Martin Hellman dokazala su nešto naizgled nemoguće: da se dvije osobe, razgovarajući samo putem javnog kanala — kanala na kojem svatko može čuti sve što govore — mogu dogovoriti o tajnoj lozinki a da je nijedan slušatelj ne može otkriti. Zvuči kao magija. Nije: to je matematika. Razmjena ključeva Diffie-Hellman, kako je od tada poznata, osnova je praktički cjelokupne kriptirane komunikacije na internetu, a pola stoljeća intenzivne uporabe i svjetskog akademskog nadzora potvrđuju njezinu solidnost. Tko želi vidjeti vizualnu intuiciju ili matematiku, može nastaviti čitati. Tko radije vjeruje da to radi, također može nastaviti bez gubljenja niti članka.

Za one koji to žele zamisliti u slici, postoji poznata analogija s bojama. Zamislite da se Alice i Bruno javno dogovore oko osnovne boje — recimo žute — pred očima Eve koja ih sluša. Svatko privatno odabere drugu tajnu boju i pomiješa svoju tajnu sa žutom. Alice dobije određenu narančastu; Bruno dobije određenu zelenu. Razmijene rezultate pred očima Eve. Sada svatko pomiješa primljenu boju sa svojom tajnom i oboje dođu do iste konačne boje, jer redoslijed miješanja nije važan. Eva je vidjela žutu i dvije međumješavine, ali ne i tajne; bez neke od tajni ne može doći do konačne boje. Stvarna matematika zamjenjuje boje potenciranjem u modularnim grupama ili eliptičkim krivuljama, ali ideja je ista: zajednička tajna gradi se javno a da je nitko na kanalu ne može rekonstruirati.

U aritmetici, za one koji radije vide mehanizam: Alice odabire tajni broj a, Bruno odabire b. Razmjenjuju g^a i g^b javno preko kanala. Alice izračunava $(g^b)^a$ a Bruno izračunava $(g^a)^b$; oboje dolaze do istog g^{ab}. Eva vidi g, g^a i g^b kako prolaze kanalom, ali povrat a iz g^a — takozvani problem diskretnog logaritma — zahtijeva astronomsko vrijeme izračuna veće od starosti svemira kada se g odabere u prikladnoj matematičkoj grupi.

Para quien quiera comprobarlo con números pequeños. El intercambio Diffie-Hellman se puede recorrer entero con cifras lo bastante reducidas como para hacer las cuentas a mano. Quien prefiera no entrar en aritmética puede saltarse este bloque sin perder el hilo del artículo; quien quiera ver el mecanismo funcionando paso a paso lo encontrará aquí. **Las reglas públicas**, que cualquiera puede leer: un primo $p = 11$ (en el Diffie-Hellman real es de unas trescientas cifras; usamos once para que las cuentas quepan en una página), una base $g = 2$, y la convención de

que toda la aritmética se hace *módulo p* — se calcula, se divide entre *p*, y se conserva el resto, como un reloj de once posiciones que vuelve al cero al rebasar el diez. **Las elecciones privadas**, una cada uno y jamás compartidas: Alicia elige $a = 4$. Bruno elige $b = 7$.

Paso 1. Alicia calcula $2^4 = 16$, luego $16 \bmod 11 = \mathbf{5}$. Envía el cinco. Eva lo anota.

Paso 2. Bruno calcula $2^7 = 128$, luego $128 \bmod 11 = \mathbf{7}$. Envía el siete. Eva también lo anota. Tras los dos envíos, la libreta de Eva contiene cuatro datos: $p = 11$, $g = 2$, A = 5, B = 7. Le falta el número compartido que Alicia y Bruno están a punto de derivar — y que Eva no podrá reconstruir.

Paso 3. Alicia toma el siete que Bruno le envió y lo eleva a su exponente privado $a = 4$. Para evitar manejar $7^4 = 2401$, se calcula por partes aplicando el módulo en cada paso:

$7^2 = 49$

$49 \bmod 11 = 5$

$7^4 = (7^2)^2 = 5^2 = 25$

$25 \bmod 11 = 3$

Alicia obtiene el número **3**.

Paso 4. Bruno toma el cinco que Alicia le envió y lo eleva a su exponente privado $b = 7$. De nuevo por partes:

$5^2 = 25 \bmod 11 = 3$

$5^4 = (5^2)^2 = 3^2 = 9 \bmod 11 = 9$

$5^6 = 5^4 \times 5^2 = 9 \times 3 = 27 \bmod 11 = 5$

Finalmente $5^7 = 5^6 \times 5 = 5 \times 5 = 25 \bmod 11 = \mathbf{3}$.

Bruno obtiene también **3**.

Los dos han llegado al mismo número, 3, trabajando en paralelo. Ninguno envió su exponente privado en ningún momento. Alicia no sabe que $b = 7$; Bruno no sabe que $a = 4$. Cada cual usó el valor público que el otro envió combinado con su propio exponente privado, y se encontraron en el mismo destino. **¿Por qué llegan al mismo número?** Lo que calculó cada uno: Alicia, $(g^b)^a = 2^{7\times4} = 2^{28} \bmod 11$. Bruno, $(g^a)^b = 2^{4\times7} = 2^{28} \bmod 11$. Es la misma cantidad porque el orden de multiplicación de exponentes no importa ($7 \times 4 = 4 \times 7$). Cada cual llegó por un camino distinto al mismo destino.

¿Y Eva? Tiene en su libreta $p = 11$, $g = 2$, A = 5, B = 7, y quisiera el 3. Para calcularlo necesitaría conocer a o b — pero ninguno ha viajado por el canal. Su única vía es preguntarse: «¿para qué exponente a se cumple $2^a \bmod 11 = 5$?». Con p tan pequeño puede probar 0, 1, 2, 3, 4… y encontrarlo en menos de un minuto. Pero al sustituir 11 por un primo de trescientas cifras, el espacio de exponentes posibles tiene más elementos que átomos hay en el universo observable. **No existe a día de hoy ningún algoritmo conocido por la humanidad que pueda recorrer ese espacio en menos de miles de millones de años.** Es el llamado *problema del logaritmo discreto*: fácil hacia adelante, computacionalmente imposible hacia atrás. Y es la razón por la que el cifrado resiste aunque Eva haya seguido toda la conversación letra por letra.

Tres ingredientes simples —aritmética sobre un reloj, exponenciación, y conmutatividad de la multiplicación ($a \cdot b = b \cdot a$)— combinados producen un protocolo del que media humanidad depende cada día para sus comunicaciones privadas. Ninguna de las tres piezas, por separado, parece especial. Lo decisivo es el ensamblaje.

Od Diffie-Hellmana do protokola Signal

Enkripcija od kraja do kraja koju danas koriste profesionalne aplikacije za razmjenu poruka oslanja se, gotovo bez iznimke, na elegantnu i ojačanu verziju razmjene Diffie-Hellman. Protokol Signal, koji su dizajnirali Trevor Perrin i Moxie Marlinspike između 2013. i 2016., referenca je. Kombinira dvije ključne ideje. Prva je razmjena ključeva u eliptičkim krivuljama (X25519), koja proizvodi početnu zajedničku tajnu između dva uređaja. Druga je takozvani Double Ratchet — dvostruki zupčanik — koji automatski obnavlja ključeve sa svakom porukom, tako da kompromitiranje uređaja danas ne dopušta dekriptiranje prošlih poruka, niti budućih poruka nakon što se zupčanik okrenuo.

U Zigu, razmjena X25519 koja proizvodi zajedničku tajnu između dva uređaja stane u šest redaka, koristeći standardnu biblioteku:

```
const std = @import("std");
const X25519 = std.crypto.dh.X25519;

// Alicia y Bruno generan cada uno un par (privada, pública).
const par_alicia = X25519.KeyPair.generate(io);
const par_bruno  = X25519.KeyPair.generate(io);

// Cada parte recibe la clave pública de la otra y deriva el mismo secreto.
const secreto_alicia = X25519.scalarmult(par_alicia.secret_key, par_bruno.public_key) catch unreachable;
const secreto_bruno  = X25519.scalarmult(par_bruno.secret_key,  par_alicia.public_key) catch unreachable;
// secreto_alicia == secreto_bruno  (32 bytes)
```

Što se događa u tih šest redaka: Javni ključevi putuju otvoreno. Privatni ključevi nikada ne napuštaju odgovarajući uređaj. Svaka strana izvodi, iz svog privatnog i javnog ključa druge strane, istu tajnu od trideset i dva bajta koju nitko u kanalu ne može vratiti. Ta tajna kasnije služi kao sjeme za kriptiranje razmijenjenih poruka. Double Ratchet protokola Signal dodaje stalnu rotaciju tog materijala kako kompromitiranje jednog trenutka ne bi ugrozilo ostatak razgovora.

A što je točno unutar `std.crypto.dh.X25519`? Nema skrivene magije. To su dvije kratke funkcije koje se mogu u cijelosti pročitati u samoj standardnoj biblioteci Ziga. Prva izvodi javni ključ iz privatnog — «g^a» razmjene:

```
pub fn recoverPublicKey(secret_key: [secret_length]u8) IdentityElementError![public_length]u8 {
    const q = try Curve.basePoint.clampedMul(secret_key);
    return q.toBytes();
}
```

Rječnikom iz članka: privatni ključ se «množi» — u eliptičkom smislu, ne osnovnoaritmetičkom — osnovnom točkom Curve25519 krivulje, a rezultat se serijalizira u trideset i dva bajta. Operacija `clampedMul` ojačana je verzija tog skalarnog množenja: uključuje zaštitne mjere koje je kriptografska zajednica dodavala godinama kako bi se oduprla poznatim obiteljima napada. Dva retka tijela funkcije.

Druga funkcija kombinira vaš privatni ključ s javnim ključem koji vam šalje druga strana. To je «$(g^b)^a$» razmjene, koje proizvodi dijeljenu tajnu od trideset i dva bajta koju nijedno od vas nikada nije prenijelo:

```
pub fn scalarmult(secret_key: [secret_length]u8, public_key: [public_length]u8) IdentityElementError![shared_length]u8 {
    const q = try Curve.fromBytes(public_key).clampedMul(secret_key);
    return q.toBytes();
}
```

Još dva retka. Primljeni javni ključ tumači se kao točka na krivulji i «množi» se vlastitim privatnim ključem. Zbog komutativnosti operacije na krivulji — analogne komutativnosti množenja eksponenata koju smo vidjeli u numeričkom primjeru — obje strane završavaju s istom serijaliziranom točkom: upravo onom dijeljenom tajnom o kojoj članak govori.

To je sve. Ono što u aplikaciji izgleda kao magija, u stvarnosti su dvije funkcije od po tri retka. Tehnička složenost koncentrirana je u jednoj operaciji, `clampedMul`, koja je napisana dalje u istoj standardnoj biblioteci, desetljećima pregledavana od strane međunarodne kriptografske zajednice, te dostupna svima koji je žele čitati slovo po slovo. Nema crne kutije ni u našoj aplikaciji ni u Zigovoj standardnoj biblioteci. Postoji kod otvorenog koda koji čovjek može razumjeti, birajući tempo kojim želi ulaziti u njega.

Što enkripcija od kraja do kraja štiti

Ono što E2EE dobro štiti, pod pretpostavkom ispravne implementacije, jest sadržaj poruke u tranzitu. Posredni poslužitelj koji primi i proslijedi kriptirane podatke vidjet će niz nerazumljivih bajtova. Napadač s pristupom kabelu, usmjerivaču (routeru), wifi pristupnoj točki vidjet će isto. Pružatelj usluge koji čuva kopije prometa neće ih moći pročitati naknadno. Vlada koja naredi operateru usluge da preda sadržaj primit će iste nerazumljive bajte koje je poslužitelj imao na prvom mjestu.

To je, u praktičnom smislu, puno. To je razlika između pisanja pisma unutar neprozirne omotnice i pisanja na razglednici. Obje stižu. Samo jedna čuva sadržaj pred poštarom.

Što enkripcija od kraja do kraja ne štiti

Vrijedi to znati jednako dobro. E2EE ne štiti metapodatke: poslužitelj i dalje zna da korisnik A šalje podatke korisniku B, u koliko sati, kojom učestalošću i odakle, iako ne zna što kaže. Ti metapodaci, kao što smo već tvrdili u *Kriptirati ne znači biti privatan*, često su rječitiji od sadržaja. Znati da je netko nazvao odvjetnički ured specijaliziran za razvode u petak u 22:00 na trideset minuta priča priču koju sadržaj poziva nikada nije ispričao. To je ista situacija kao vidjeti osobu kako nekoliko puta ulazi i izlazi iz onkološke klinike: ne treba čuti ništa od onoga o čemu se unutra razgovara da bi se zamislilo što se događa. Jedan samostalni metapodatak možda ne znači ništa; nekoliko međusobno ukrštenih crtaju nešto previše slično istini. E2EE ne štiti krajeve: ako je uređaj primatelja kompromitiran zlonamjernim programom, poruka se normalno dekriptira za tog primatelja i zlonamjerni program je čita. E2EE ne štiti od identiteta samog sugovornika: ako Alice vjeruje da razgovara s Brunom, ali se napadač umetnuo na početku (*man in the middle*) i protokol ne uključuje neovisnu provjeru, dvije strane završe razgovarajući s uljezom misleći da razgovaraju međusobno.

Postoji i četvrta stvar koju vrijedi formulirati bez dvosmislenosti. E2EE ne sprječava pružatelja koji tvrdi da ga nudi da dodatno zadrži kopiju nekriptirane poruke u vlastitim sustavima. Tvrdnja „moje su poruke kriptirane od kraja do kraja“ i tvrdnja „pružatelj ne čuva moj sadržaj“ nisu iste. Aplikacija može ispunjavati prvu dok krši drugu; vidjeli smo to u novinskim naslovima više puta od 2018. Korisnik, osim ako kod klijenta nije provjerljiv, nema tehničkog načina da razlikuje jedan slučaj od drugog bez stručne istrage. Najpoznatiji slučaj u širokoj javnosti: WhatsApp kriptira poruke od kraja do kraja u tranzitu, ali ako korisnik aktivira sigurnosnu kopiju na iCloudu ili Google Driveu bez dodatnog kriptiranja, ta se kopija pohranjuje čitljiva u infrastrukturi treće strane, a kriptiranje se prekida na kraju samog korisnika.

Pitanje koje operater ne želi čuti

Aplikacija koja tvrdi da kriptira od kraja do kraja može tehnički učiniti jednu od tri stvari u vezi s ključevima:

1. **Ključevi borave samo na uređajima.** Generiraju se i borave isključivo na uređajima korisnika; operater ih ne poznaje niti ih pohranjuje. To je optimalan slučaj.

2. **Operater može pristupiti ako želi.** Operater posjeduje ključeve korisnika (ili ih može generirati po želji) i pohranjuje ih u svoje baze podataka. Ako želi ili je prisiljen, može pročitati sadržaj. To je slučaj kod većine „cloud” usluga.
3. **Operater ne može pristupiti po dizajnu, ali kontrolira pristup.** Operater nema ključeve, ali ima kontrolu nad aplikacijom koja ih generira. Ako je prisiljen, može poslati zlonamjerno ažuriranje koje snima ključeve ili sadržaj prije enkripcije. To je slučaj kod mnogih komercijalnih E2EE usluga.

Operativno pitanje stoga nije je li nešto kriptirano, već tko ima kontrolu nad uređajem i softverom koji upravlja ključevima. U Solo2 ključevi se nalaze isključivo u vašem Trezoru (IndexedDB kriptiran vašom lozinkom), a softver je provjerljiv otvoreni kod.

Za profesionalne čitatelje

Enkripcija od kraja do kraja alat je za digitalni suverenitet. No, kao i svaki alat, njegova učinkovitost ovisi o ruci koja njime rukuje i o tlu na kojem se oslanja.

1. Gdje se generiraju kriptografski ključevi i gdje fizički borave? Ako im operater može pristupiti (čak i privremeno, čak i pod izgovorom oporavka), E2EE je samo nominalan.
2. Postoji li neovisna provjera sugovornika (sigurnosni brojevi, QR kodovi, out-of-band usporedba) koja sprječava man-in-the-middle napad tijekom uspostave razgovora?
3. Je li kod klijenta moguće revidirati — otvoren, objavljen, ponovljiv — ili zahtijeva povjerenje u riječ pružatelja o tome što klijent zapravo radi?
4. Koje metapodatke usluga generira i čuva, i na koliko dugo? Čak i ako je sadržaj neproziran, metapodaci mogu rekonstruirati dobar dio osjetljivih informacija.

Ova četiri pitanja ne traže napredne tehničke informacije; ona traže informacije na koje svaki pošten operater može odgovoriti u svojoj javnoj dokumentaciji. Kvaliteta i preciznost odgovora govori o proizvodu jednako koliko i sam odgovor.

Enkripcija od kraja do kraja, ako se izvede ispravno, jedna je od najfinijih konstrukcija koje je suvremena kriptografija podarila svakodnevnoj praksi. Izvorna ideja — da se dvije osobe mogu dogovoriti o tajni putem javnog kanala — pripada Whitfield Diffieu i Martin Hellmanu iz 1976. godine; pola stoljeća kasnije i dalje živimo u njezinim posljedicama. No, kao i kod svakog tehničkog obećanja, njezina vrijednost ovisi o stvarnom ispunjenju, a ne o oznaci. Pitanje poštenog profesionalca nije „je li kriptirano?“, već „tko ima ključeve?“. Odgovori imaju različite posljedice. Vrijedi ih znati.

Izvori i dodatno štivo

- Diffie, W.; Hellman, M. — *New Directions in Cryptography*, IEEE Transactions on Information Theory, studeni 1976. Temeljni članak o kriptografiji javnog ključa.
- Perrin, T.; Marlinspike, M. — *The Double Ratchet Algorithm*, javna specifikacija Open Whisper Systemsa, revizija iz 2016. Osnova Signal protokola i njegovih industrijskih derivata.
- RFC 7748 — Elliptic Curves for Security (IETF, siječanj 2016.). Normativna specifikacija krivulja X25519 i X448 koje se koriste u modernim razmjenama ključeva.
- Ferguson, N.; Schneier, B.; Kohno, T. — *Cryptography Engineering: Design Principles and Practical Applications* (Wiley, 2010). Poglavlja o razmjeni ključeva i protokolima za autentificiranu enkripciju.
- Uredba (EU) 2024/1183 o europskom okviru za digitalni identitet (eIDAS 2) — uspostavlja okvire u kojima neovisna provjera sugovornika stječe institucionalnu potporu i gdje razlikovanje između nominalne i stvarne enkripcije ima različite pravne posljedice.

← PrethodnoKill switch i institucionalno zarobljavanjeSljedeće →Poslovni model kao signal povjerenja

Nedavna čitanja

- Analiza · 18. svibnja 2026. Stvarna naspram prividne privatnosti: pitanja koja si trebate postaviti
- Analiza · 18. svibnja 2026. Self-hosting kao profesionalna praksa
- Koncept · 18. svibnja 2026. 24 riječi: što je kriptografski identitet

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 a561287caa0ad6629b1a7c67c9f22e73c524440f0ce857594301b189bf39a4d1

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