Криптиране от край до край, обяснено истински

Какво казват доставчиците, когато говорят за E2EE, и какво премълчават. Дидактическо обяснение на механизма и неговите граници, без рекламната опаковка.

За да се разберем: WhatsApp казва, че вашите съобщения са криптирани от край до край. Това е вярно — и не е достатъчно. Ако резервното копие отива в iCloud или Google Drive без допълнително криптиране, криптирането се нарушава в собствения ви телефон. Оперативният въпрос не е дали е криптирано, а къде се намират ключовете.

Какво всъщност означава криптирането

Криптирането на съобщение означава превръщането му в нещо, което изглежда като шум за всеки, който не притежава определена информация, наречена ключ. Операцията се извършва на устройството на изпращача и с правилния ключ се отменя на устройството на получателя. По средата съобщението пътува като поредица от байтове без видимо значение. Това е простата идея. Останалата част от статията се занимава с нюансите, които я превръщат, в зависимост от случая, в реална гаранция или в маркетингов етикет.

Прилагателното *от край до край* — на английски *end-to-end*, съкратено E2EE — добавя точност. Криптирането не се прави, за да може междинен сървър да го прочете и достави. Прави се така, че само двата края — устройството на изпращача и устройството на получателя — да притежават ключа. Всеки сървър, през който преминава съобщението, вижда шума, а не съобщението. Това е техническата разлика с криптирането *при транзит*, при което съдържанието се предава криптирано от един сървър на друг, но всеки сървър, през който преминава, го декриптира, за да го препрати, възстановявайки временно текста в явен вид.

Парадоксът на споделената тайна

Има очевиден проблем. За да могат двама души да криптират и декриптират съобщения помежду си, и двамата се нуждаят от един и същ ключ. Но как се договарят за този ключ, ако всичко, което си изпращат, по дефиниция преминава през канал, където някой може да слуша? Договарянето на ключа в същия канал, където по-късно ще го използват, изглежда невъзможно: ако нападателят го чуе при договарянето му, ще може да декриптира всичко следващо. В продължение на десетилетия класическата криптография решаваше това по трудния начин: ключовете се предаваха лично, преди да започнат да се използват, при физически срещи. Посланиците носеха куфарчета с ключове, зашити в подплатата на палтото им.

В съвременната електронна поща това решение не е мащабируемо. Ако трябваше физически да ходим до дома на всеки човек, с когото възнамеряваме да общуваме по криптиран начин, нямаше да стигнем до разговор с никого. Въпросът, поставен преди петдесет години от криптографската общност, беше следният: възможно ли е двама души, които не се познават и споделят само публичен канал, да се договорят в същия този публичен канал за тайна, която никой, който слуша канала, не може да разбере?

Елегантността на Diffie-Hellman

През 1976 г. двама математици на име Whitfield Diffie и Martin Hellman демонстрираха нещо на пръв поглед невъзможно: че двама души, разговарящи само по публичен канал — канал, в който всеки може да чуе всичко, което казват — могат да се договорят за тайна парола, без никой слушател да може да я разкрие. Звучи като магия. Не е: това е математика. Обменът на ключове на Diffie-Hellman, както е известен оттогава, е основата на практически цялата криптирана комуникация в интернет и половин век интензивна употреба и световен академичен контрол доказват неговата солидност. Всеки, който иска да види визуалната интуиция или математиката, може да продължи да чете. Който предпочита да се довери, че работи, също може да продължи, без да губи нишката на статията.

За тези, които искат да си го представят, има известна аналогия с цветове. Представете си, че Алиса и Бруно се споразумяват открито за основен цвят — да кажем жълт — пред очите на Ева, която ги слуша. Всеки избира насаме втори таен цвят и смесва своята тайна с жълтото. Алиса получава особен оранжев цвят; Бруно получава особен зелен цвят. Те обменят резултатите пред очите на Ева. Сега всеки смесва получения цвят със своята собствена тайна и двамата стигат до един и същ краен цвят, тъй като редът на смесване няма значение. Ева е видяла жълтото и двете междинни смеси, но не и тайните; без някоя от тайните тя не може да стигне до крайния цвят. Реалната математика заменя цветовете с експоненциации в модулни групи или елиптични криви, но идеята е същата: споделената тайна се изгражда публично, без никой в канала да може да я възстанови.

В аритметиката, за тези, които предпочитат да видят механизма: Алиса избира тайно число a, Бруно избира b. Те обменят g^a и g^b открито по канала. Алиса изчислява $(g^b)^a$, а Бруно изчислява $(g^a)^b$; и двамата стигат до едно и също g^{ab}. Ева вижда g, g^a и g^b да преминават по канала, но възстановяването на a от g^a — така нареченият проблем с дискретния логаритъм — изисква астрономическо време за изчисление, превъзхождащо възрастта на вселената, когато g е избрано в подходяща математическа група.

За тези, които искат да го проверят с малки числа. Обменът на Diffie-Hellman може да бъде проследен изцяло с числа, достатъчно малки, за да се направят изчисленията на ръка. Който предпочита да не навлиза в аритметика, може да прескочи този блок, без да губи нишката на статията; който иска да види как механизмът работи стъпка по стъпка, ще го намери тук. **Публичните правила**, които всеки може да прочете: просто число $p = 11$ (в истинския Diffie-Hellman е от около триста цифри; използваме единадесет, за да се съберат сметките на една страница), основа $g = 2$ и конвенцията, че цялата аритметика се извършва по *модул p* — изчислява се, разделя се на p и се запазва остатъкът, като часовник с единадесет позиции, който се връща на нула, когато надвиши десет. **Частните избори**, по един за всеки и никога несподеляни: Алиса избира $a = 4$. Бруно избира $b = 7$.

Стъпка 1. Алиса изчислява $2^4 = 16$, след това $16 \bmod 11 = \mathbf{5}$. Изпраща петицата. Ева я записва.

Стъпка 2. Бруно изчислява $2^7 = 128$, след това $128 \bmod 11 = \mathbf{7}$. Изпраща седмицата. Ева също я записва. След двете изпращания тефтерът на Ева съдържа четири данни: $p = 11$, $g = 2$, A = 5, B = 7. Липсва й споделеното число, което Алиса и Бруно са на път да изведат — и което Ева няма да може да реконструира.

Стъпка 3. Алиса взема седмицата, която Бруно й изпрати, и я повдига на своя частен степенен показател $a = 4$. За да се избегне работа с $7^4 = 2401$, се изчислява на части, като се прилага модулът на всяка стъпка:

$7^2 = 49$

$49 \bmod 11 = 5$

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

$25 \bmod 11 = 3$

Алиса получава числото **3**.

Стъпка 4. Бруно взема петицата, която Алиса му изпрати, и я повдига на своя частен степенен показател $b = 7$. Отново на части:

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

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

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

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

Бруно също получава **3**.

И двамата достигнаха до едно и също число, 3, работейки паралелно. Никой от тях не изпрати своя частен показател в нито един момент. Алиса не знае, че $b = 7$; Бруно не знае, че $a = 4$. Всеки използва публичната стойност, която другият изпрати, комбинирана със собствения му частен показател, и се срещнаха на същата дестинация. **Защо стигат до едно и също число?** Какво изчисли всеки един: Алиса, $(g^b)^a = 2^{7\times4} = 2^{28} \bmod 11$. Бруно, $(g^a)^b = 2^{4\times7} = 2^{28} \bmod 11$. Това е едно и също количество, защото редът на умножение на показателите няма значение ($7 \times 4 = 4 \times 7$). Всеки стигна по различен път до една и съща дестинация.

А Ева? В нейния тефтер са $p = 11$, $g = 2$, A = 5, B = 7 и тя би искала 3. За да го изчисли, ще трябва да знае a или b — но нито едно от тях не е пътувало по канала. Единственият й начин е да се запита: «за кой показател a е изпълнено $2^a \bmod 11 = 5$?». При толкова малко p тя може да тества 0, 1, 2, 3, 4... и да го намери за по-малко от минута. Но когато заменим 11 с просто число от триста цифри, пространството на възможните показатели има повече елементи, отколкото са атомите във видимата вселена. **Към днешна дата не съществува алгоритъм, познат на човечеството, който да може да обходи това пространство за по-малко от милиарди години.** Това е така наречените *проблем на дискретния логаритъм*: лесно напред, изчислително невъзможно назад. И това е причината криптирането да устои, дори ако Ева е проследила целия разговор буква по буква.

Три прости съставки — аритметика на часовник, степенуване и комутативност на умножението ($a \cdot b = b \cdot a$) — комбинирани създават протокол, от който половината човечество зависи всеки ден за личната си комуникация. Нито една от трите части сама по себе си не изглежда специална. Решаващо е сглобяването.

От Diffie-Hellman до протокола Signal

Криптирането от край до край, използвано днес от професионалните приложения за съобщения, се основава, почти без изключение, на елегантна и подсилена версия на обмена на Diffie-Hellman. Референтен е протоколът Signal, разработен от Trevor Perrin и Moxie Marlinspike между 2013 и 2016 г. Той съчетава две ключови идеи. Първата е обменът на ключове в елиптични криви (X25519), който създава първоначалната споделена тайна между две устройства. Втората е така наречените Double Ratchet — двоен механизъм — който подновява ключовете автоматично с всяко съобщение, така че компрометирането на устройството днес не позволява декриптиране на минали съобщения, нито на бъдещи съобщения, след като механизмът е бил завъртян.

В Zig обменът X25519, който създава споделената тайна между две устройства, се побира в шест реда, като се използва стандартната библиотека:

```
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)
```

Какво се случва в тези шест реда: Публичните ключове пътуват открито. Частните ключове никога не напускат съответното устройство. Всяка страна извежда от своя частен ключ и публичния ключ на другата страна същата тайна от тридесет и два байта, която никой в канала не може да възстанови. Тази тайна служи по-късно като основа за криптиране на обменените съобщения. Double Ratchet на протокола Signal добавя постоянно завъртане на този материал, така че компрометирането на един момент да не компрометира останалата част от разговора.

И какво точно има вътре в `std.crypto.dh.X25519`? Няма скрита магия. Това са две кратки функции, които могат да бъдат прочетени изцяло в самата стандартна библиотека на Zig. Първата извежда публичния ключ от частния — това е «g^a» в обмена:

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

На езика на статията: частният ключ се «умножава» — в елиптичния, а не в елементарния аритметичен смисъл — по базовата точка на кривата Curve25519 и резултатът се сериализира в тридесет и два байта. Операцията `clampedMul` е подсилената версия на това скаларно умножение: тя включва предпазните мерки, които криптографската общност е добавяла през годините, за да устои на известни семейства атаки. Два реда в тялото на функцията.

Втората функция комбинира вашия частен ключ с публичния ключ, който другата страна ви изпраща. Това е «$(g^b)^a$» в обмена, което създава споделената тайна от тридесет и два байта, която никой от вас не е предал:

```
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();
}
```

Още два реда. Полученият публичен ключ се интерпретира като точка върху кривата и се «умножава» по собствения частен ключ. Поради комутативността на операцията върху кривата — аналогична на комутативността при умножението на показателите, която видяхме в числовия пример — и двете страни завършват с една и съща сериализирана точка: точно споделената тайна, за която се говори в статията.

Това е всичко. Това, което в едно приложение изглежда като магия, в действителност са две функции от по три реда всяка. Техническата сложност е концентрирана в една-единствена операция, `clampedMul`, която е написана по-нататък в същата стандартна библиотека, преглеждана е десетилетия наред от международната криптографска общност и е достъпна за всеки, който иска да я прочете буква по буква. Няма черна кутия нито в нашето приложение, нито в стандартната библиотека на Zig. Има отворен код, който всеки човек може да разбере, избирайки темпото, с което иска да навлезе в него.

Какво защитава криптирането от край до край

Това, което E2EE защитава добре, при условие че е реализирано правилно, е съдържанието на съобщението по време на транзит. Междинен сървър, който получава и препраща криптираните данни, ще види поредица от неразбираеми байтове. Нападател с достъп до кабела, рутера, точката за достъп до Wi-Fi ще види същото. Доставчик на услуги, който съхранява копия от трафика, няма да може да го прочете впоследствие. Правителство, което нареди на оператора на услугата да предаде съдържанието, ще получи същите неразбираеми байтове, които сървърът е имал първоначално.

Това, в практически план, е много. Това е разликата между писането на писмо в непрозрачен плик и писането му на пощенска картичка. И двете пристигат. Само едната запазва съдържанието пред пощальона.

Какво не защитава криптирането от край до край

Добре е да го знаете също толкова добре. E2EE не защитава метаданните: сървърът продължава да знае, че потребител А изпраща данни на потребител Б, в колко часа, с каква честота и откъде, дори и да не знае какво казва. Тези метаданни, както вече аргументирахме в *Да криптираш не означава да бъдеш защитен*, често са по-показателни от съдържанието. Знанието, че някой се е обадил в адвокатска кантора, специализирана в разводи, в петък в 22:00 ч. за тридесет минути, разказва история, която съдържанието на разговора никога не е разказвало. Това е същата ситуация като да видиш човек да влиза и излиза няколко пъти от онкологична клиника: не е нужно да чуваш нищо от това, което се говори вътре, за да си представиш какво се случва. Един-единствен отделен метаданък може да не означава нищо; няколко кръстосани помежду си очертават нещо твърде подобно на истината. E2EE не защитава крайните точки: ако устройството на получателя е компрометирано от зловредна програма, съобщението се декриптира нормално за този получател и зловредната програма го прочита. E2EE не защитава срещу самоличността на самия събеседник: ако Алиса вярва, че разговаря с Бруно, но нападател се е намесил в началото (*man in the middle*) и протоколът не включва независима проверка, двете страни в крайна сметка разговарят с натрапника, мислейки, че разговарят помежду си.

Има четвърто нещо, което е добре да се формулира без двусмислие. E2EE не пречи на доставчик, който твърди, че го предлага, да пази освен това копие от некриптираното съобщение в собствените си системи. Твърдението „моите съобщения са криптирани от край до край“ и твърдението „доставчикът не запазва съдържанието ми“ не са едно и също. Приложението може да изпълнява първото, докато нарушава второто; виждали сме го в заглавията на пресата многократно от 2018 г. насам. Потребителят, освен ако кодът на клиента не е проверим, няма технически начин да разграничи единия случай от другия без експертно разследване. Най-известният случай в широката общественост: WhatsApp криптира съобщенията от край до край при транзит, но ако потребителят активира архивирането в iCloud или Google Drive без допълнително криптиране, това копие се съхранява четливо в инфраструктурата на трета страна и криптирането се прекъсва в края на самия потребител.

Въпросът, който операторът не иска да чуе

Приложение, което твърди, че криптира от край до край, може технически да прави едно от три неща по отношение на ключовете:

1. **Ключовете се намират само в устройствата.** Те се генерират и се намират изключително в устройствата на потребителите; операторът не ги познава и не ги съхранява. Това е оптималният случай.
2. **Операторът може да има достъп, ако желае.** Операторът притежава ключовете на потребителите (или може да ги генерира по свое желание) и ги съхранява в своите бази данни. Ако желае или бъде принуден, той може да прочете съдържанието. Това е случаят с повечето „облачни“ услуги.
3. **Операторът не може да има достъп по проект, но контролира достъпа.** Операторът не разполага с ключовете, но контролира приложението, което ги генерира. Ако бъде принуден, той може да изпрати зловредна актуализация, която да улови ключовете или съдържанието преди криптирането. Това е случаят с много търговски E2EE услуги.

Следователно оперативният въпрос не е дали нещо е криптирано, а кой контролира устройството и софтуера, който управлява ключовете. В Solo2 ключовете се съхраняват единствено във вашия Сейф (IndexedDB, криптирана с вашата парола), а софтуерът е с отворен код и може да бъде проверен.

За професионалния читател

Криптирането от край до край е инструмент за дигитален суверенитет. Но както всеки инструмент, неговата ефективност зависи от ръката, която го държи, и от почвата, на която стъпва.

1. Къде се генерират криптографските ключове и къде се намират физически? Ако операторът може да има достъп до тях (дори временно, дори под формата на възстановяване), E2EE е само номинално.
2. Съществува ли независима проверка на събеседника (номера за сигурност, QR кодове, сравнение извън канала), която да предотврати атака „човек по средата“ по време на установяване на разговора?
3. Кодът на клиента подлежи ли на одит — отворен, публикуван, възпроизводим — или изисква да се доверим на думата на доставчика за това какво реално прави клиентът?
4. Какви метаданни генерира и съхранява услугата и за колко време? Дори ако съдържанието е непрозрачно, метаданните могат да реконструират голяма част от чувствителната информация.

Тези четири въпроса не изискват напреднала техническа информация; те изискват информация, на която всеки честен оператор може да отговори в своята публична документация. Качеството и точността на отговора говорят за продукта толкова, колкото и самият отговор.

Криптирането от край до край, направено правилно, е една от най-фините конструкции, които съвременната криптография е предоставила на ежедневната практика. Оригиналната идея — двама души да се споразумеят за тайна през публичен канал — принадлежи на Whitfield Diffie и Martin Hellman, 1976 г.; половин век по-късно продължаваме да живеем с последствията от нея. Но, както при всяко техническо обещание, стойността му зависи от реалното изпълнение, а не от етикета. Въпросът на честния професионалист не е „криптирано ли е?“, а „кой притежава ключовете?“. Отговорите имат различни последствия. Добре е да ги знаете.

Източници и допълнително четиво

- Diffie, W.; Hellman, M. — *New Directions in Cryptography*, IEEE Transactions on Information Theory, ноември 1976 г. Основополагаща статия за криптографията с публичен ключ.
- Perrin, T.; Marlinspike, M. — *The Double Ratchet Algorithm*, публична спецификация на Open Whisper Systems, ревизия 2016 г. Основа на протокола Signal и неговите индустриални производни.
- RFC 7748 — Elliptic Curves for Security (IETF, януари 2016 г.). Нормативна спецификация на кривите X25519 и X448, използвани в съвременния обмен на ключове.
- Ferguson, N.; Schneier, B.; Kohno, T. — *Cryptography Engineering: Design Principles and Practical Applications* (Wiley, 2010). Глави за обмен на ключове и протоколи за удостоверено криптиране.
- Регламент (ЕС) 2024/1183 за рамката за европейска цифрова самоличност (eIDAS 2) — установява рамки, в които независимата проверка на събеседника придобива институционална подкрепа и където разграничението между номинално и реално криптиране има различни правни последици.

← ПредишнаKill switch и институционалното овладяванеСледваща →Бизнес моделът като сигнал за доверие

Скорошни четива

- Анализ · 18 май 2026 г. Реална срещу привидна поверителност: въпросите, които е добре да си зададете
- Анализ · 18 май 2026 г. Self-hosting като професионална практика
- Концепция · 18 май 2026 г. 24-те думи: какво е криптографска идентичност

Вземете тази статия със себе си навсякъде, където ви е необходима.

↓ Markdown ↓ Обикновен текст ↓ PDF

Файлът ще бъде изтеглен на вашето устройство. Оттам можете да го запазите, импортирате в Solo2 или споделите където пожелаете. Cuadernos не решава дестинацията вместо вас.

Восъчен печат · SHA-256 df28f6f35295c5a1774b8714305c9a32f8a1ebc9506adc4b52aa9365803f3699

Cuadernos Lacre · Публикация на Menzuri Gestión S.L. ·
написана от R.Eugenio · редактирана от екипа на Solo2.