Аналіз · 16 травня 2026 р.

← Попередній·Наступний →

Коли посередині нікого немає

Шифрування того, що проходить через сервер, захищає вміст. Відсутність сервера посередині усуває це питання. Це не одне й те саме.

Дві людини, одна розмова

Коли двоє людей розмовляють віч-на-віч у кімнаті, ніхто не повинен обіцяти, що нічого не чув. Він не чув, бо його там не було. Коли двоє людей передають папірець з рук у руки, нікому посередині не потрібно присягати, що він його не читав. Посередині нікого немає.

Більшість речей у повсякденному житті працюють саме так. Ми не підписуємо угоди про конфіденційність із повітрям, яке передає наш голос, або з папером, який ми тримаємо. Конфіденційність розмови не спирається на обіцянку посередника, тому що посередника немає. Це одна з найсильніших існуючих форм приватності: не тому, що хтось або щось поводиться добре, а тому, що цього когось або чогось немає.

Коли розмова переноситься в цифровий канал, це змінюється за замовчуванням. Звичайна модель така: двоє людей підключаються до сервера, сервер отримує повідомлення, шифрує його або зберігає в зашифрованому вигляді і доставляє одержувачу. Сервер знаходиться посередині. Сервер може бути чесним. Він може бути перевіреним. Він може працювати у сприятливій юрисдикції та за суворої політики конфіденційності. Усе це може бути правдою. Але сервер знаходиться посередині.

Різниця між шифруванням і незбиранням (друга частина)

У попередній статті цієї ж серії ми стверджували, що шифрування вмісту і незбирання метаданих — це не одне й те саме. Є ще один крок, який слід чітко сформулювати: шифрування того, що проходить через сервер, і відсутність сервера — це також не одне й те саме.

Перша модель — сервер посередині, зашифрований вміст — захищає вміст від оператора сервера, від його обслуговуючого персоналу, від зовнішнього зловмисника, який може зламати систему. І це важливо. Але це не усуває сервер. Сервер усе ще там. Він продовжує обробляти метадані. Він продовжує залишатися точкою, яка може отримати судовий запит, законне втручання, політичний тиск або порушення безпеки. Він продовжує залишатися точкою, яка вимагає довіри до когось.

Друга модель — відсутність сервера між двома кінцями — не захищає зашифрований вміст краще: якщо криптографія надійна, вміст захищений в обох випадках. Змінюється не вміст. Змінюється те, що питання *«що відбувається із сервером?»* втрачає сенс, бо немає сервера, про який можна було б запитати.

Довіра, відсутність і різниця між ними

Довіра може бути виправданою. Чесні компанії існують. Суворі аудитори існують. Сприятливі для користувача закони існують. Серйозні сервіси, які скрупульозно дотримуються всього вищезазначеного,

існують. Довіра, коли вона надається оператору, який на неї заслуговує, не є поганим рішенням.

Але довіра, хоч би якою сильною вона була, залишається довірою. Це соціальне рішення, а не технічне. Компанія може змінити власника. Юрисдикція може змінити уряд. Судовий наказ може надійти завтра. Нова вразливість може бути виявлена наступного місяця. Нічого з цього не відбувається через злий намір. Це відбувається тому, що оператор існує, і все, що існує, піддається непередбачуваним обставинам світу.

Відсутність оператора не піддається цим самим обставинам. Судовий наказ не може вимагати даних від сервера, якого не існує. Зловмисник не може зламати сервер, якого не існує. Зміна в політики компанії не може вплинути на дані, яких ця компанія ніколи не мала. Ключова фраза проста: дані, яких не існує, неможливо втратити.

Про законний аргумент на боці сервера

Той, хто пропонує послугу професійного обміну повідомленнями з сервером посередині, зазвичай наводить три цілком слушні аргументи. По-перше, що сервер необхідний для гарантування доставки, коли одержувач перебуває в офлайні. По-друге, що шифрування вмісту надійне і тому оператор не може його прочитати. По-третє, що сервіс відповідає європей законодавству і що дані захищені законом.

Усі три аргументи вірні. Жоден з них не змінює суті справи. Це правда, що сервер дозволяє зберігати повідомлення для відкладеної доставки; також правда, що відкладена доставка може бути вирішена іншим шляхом, за допомогою протоколів прямого зв'язку між пристроями, вдосконаленими протягом десятиліть і діючими сьогодні. Це правда, що шифрування вмісту в транзиті є надійним у серйозних сервісах. І це правда, що європейське законодавство захищає користувачів більше, ніж у багатьох інших місцях.

Питання не в тому, чи є послуги з сервером посередині законними, ні в тому, чи є вони безпечними, ні в тому, чи захищають вони вміст. Вони можуть бути такими, є законними і зазвичай безпечними. Питання полягає в тому, що наявність сервера посередині — це архітектурний вибір, а не технічна необхідність. І кожен вибір має наслідки. Архітектура з сервером посередині неминуче створює суб'єкта, якому потрібно довіряти. Архітектура без сервера посередині — ні.

Що каже закон і що робить архітектура

Загальний регламент про захист даних (RGPD) не вимагає конкретної архітектурної моделі. Він вимагає результатів: мінімізації даних, обмеженої мети, захисту на стадії проектування та за замовчуванням, здатності продемонструвати відповідність. Сервіс із сервером посередині може відповідати всім цим вимогам. Сервіс без сервера посередині відповідає багатьом з них конструктивно, а не декларативно. Абсолютна мінімізація — не збирати нічого, що не є суворо необхідним для доставки повідомлення, — є тривіальною, коли немає сервера, який міг би щось збирати.

Для неконфіденційного повсякденного використання архітектура із сервером є цілком розумною, і довіра до серйозного оператора є прийнятним рішенням. Для інших цілей — тих, що пов'язані з регульованою професійною таємницею, деонтологічною відповідальністю, особливо чутливою інформацією — відсутність точки довіри є не розкішшю, а структурною перевагою.

Для професійного читача

Питання, які варто поставити щодо послуги професійного зв'язку, вже знайомі з попередніх статей цієї серії, доповнюються лише одним архітектурним запитанням:

1. Чи шифрується вміст під час передачі? (Ймовірно, так.)
2. Чи створюються та зберігаються метадані про те, з ким я говорю і коли? (Ймовірно, так.)

3. Чи є сервер на шляху між моїм пристроєм і пристроєм одержувача?
4. Якщо є: хто ним керує, в якій юрисдикції, і що має статися, щоб він передав дані про мене?
5. Якщо немає: попередні питання втрачають сенс.

Різниця між двома категоріями — не в ступені, а в типі. Коли приходить час пояснювати це клієнту, пацієнту або колезі, найчесніше формулювання є водночас і найпростішим: в одній системі хтось є посередині; в іншій — ні.

Ця стаття завершує початковий цикл Cuadernos Lacre. Після розмови про шифрування, метадані та професійну таємницю ми доповнюємо архітектурну картину: шифрування вмісту і відсутність сервера посередині — це різні речі. Обидві можуть бути законними; лише одна усуває точку довіри.

Джерела та додаткова література

- Saltzer, J. H.; Reed, D. P.; Clark, D. D. — *End-to-end arguments in system design*, ACM TOCS, 1984. Основоположний текст принципу, згідно з яким гарантії системи повинні реалізовуватися на кінцях, а не в проміжному каналі.
- Регламент (ЄС) 2016/679, ст. 25 — захист даних на стадії проектування та за замовчуванням.
- Регламент (ЄС) 2016/679, ст. 5.1.c — принцип мінімізації даних.
- Schneier, B. — *Data and Goliath: the hidden battles to collect your data and control your world* (2015), W. W. Norton. Розділи про архітектури, які мінімізують збір даних за своєю конструкцією.

← ПопереднійGDPR та професійний обмін повідомленнями: чому більшість порушує правила, не знаючи про цеНаступний →CUADERNOS_LIST_SCHREMS_TITLE

Останні матеріали

- CUADERNOS_LIST_PREGUNTAS_TITLE
- CUADERNOS_LIST_SELFHOST_TITLE
- CUADERNOS_LIST_IDENTIDAD_TITLE

Візьміть цю статтю з собою куди завгодно.

↓ Markdown ↓ Звичайний текст ↓ PDF

Файл буде завантажено на ваш пристрій. Звідти ви можете зберегти його, імпортувати в Solo2 або поділитися ним де завгодно. Cuadernos не вирішує місце призначення за вас.

Сургучна печатка · SHA-256 4c8d6042da57278b9bbc76cf77ce8344f6a5226c688e922dd51ad5140e648f30

Cuadernos Lacre · Видання Menzuri Gestión S.L. ·
автор R.Eugenio · під редакцією команди Solo2.