The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

یک اثبات رمزنگارانه می‌تواند راز را با دشوار کردن اثبات نبود شبیه‌ساز پنهان کند

اثبات‌های دانش صفر اجازه می‌دهند کسی یک گزاره را بدون آشکار کردن شاهد اثبات کند. نظریهٔ کلاسیک می گوید در معنای کامل نمی‌توان هم‌زمان یک پیام، بدون راه‌اندازی مورد اعتماد و درستی کامل داشت. مقالهٔ *Ilango Rahul* این عدم امکان را ناپدید نمی کند. هدف را عوض می کند: به‌جای اینکه شبیه‌ساز واقعاً وجود داشته باشد، کافی است سامانهٔ اثبات انتخاب‌شده نتواند به‌طور کارآمد ثابت کند که شبیه‌ساز وجود ندارد. تحت مفروضات مهم در پیچیدگی اثبات و رمزنگاری، این برای بازیابی پیامدهای ابطال‌پذیر و مبتنی بر بازیِ دانش صفر، ویژگی‌به‌ویژگی، کافی است. نکته یک *primitive* آماده برای اینترنت نیست؛ راهی نظریه‌اثباتی برای تبدیل اثبات‌ناپذیری ریاضی به پوشش رمزنگارانه است.

Written by Cinzia Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · AI-assisted, human-reviewed

Paper: *Gödel in Cryptography*: *Effectively Zero-Knowledge Proofs for NP with No Interaction, No Setup, and Perfect Soundness*, Rahul Ilango, FOCS 2025 / IACR ePrint 2025/1296 · DOI: 00058.2025.FOCS63196/1109.10

ترفند این نیست که ثابت کنیم راز پنهان مانده است

از ساده‌ترین نسخهٔ دانش صفر شروع کنیم.

آلیس می‌خواهد باب را قانع کند که یک جدول سودوکو راه‌حل دارد. اگر راه‌حل را بفرستد، باب قانع می‌شود، اما معما خراب می‌شود. چیزی که آلیس می‌خواهد عجیب‌تر است: اثبات اینکه راه‌حلی وجود دارد، بدون آشکار کردن خودِ راه‌حل.

این وعدهٔ یک اثبات دانش صفر (zero-knowledge proof) است. اثبات‌کننده (آلیس) وارسی‌کننده (باب) را قانع می‌کند که گزاره‌ای درست است، بی‌آنکه چیزی فراتر از درست بودن همان گزاره فاش کند.

مشکل این است که این وعده هزینه دارد. یک اثبات ریاضی معمولی دو ویژگی آسوده‌کننده دارد. **یک پیام** است: آن را می‌نویسید، تحویل می‌دهید و می‌روید. و **درستی کامل (perfect soundness)** دارد: یک گزارهٔ نادرست اصلاً هیچ اثبات معتبر ندارد. نتایج کلاسیکِ عدم امکان می‌گویند دانش صفر باید هر دو ویژگی را کنار بگذارد — و نه فقط ترکیب هر دو، هر کدام به‌تنهایی نیز ممنوع است.

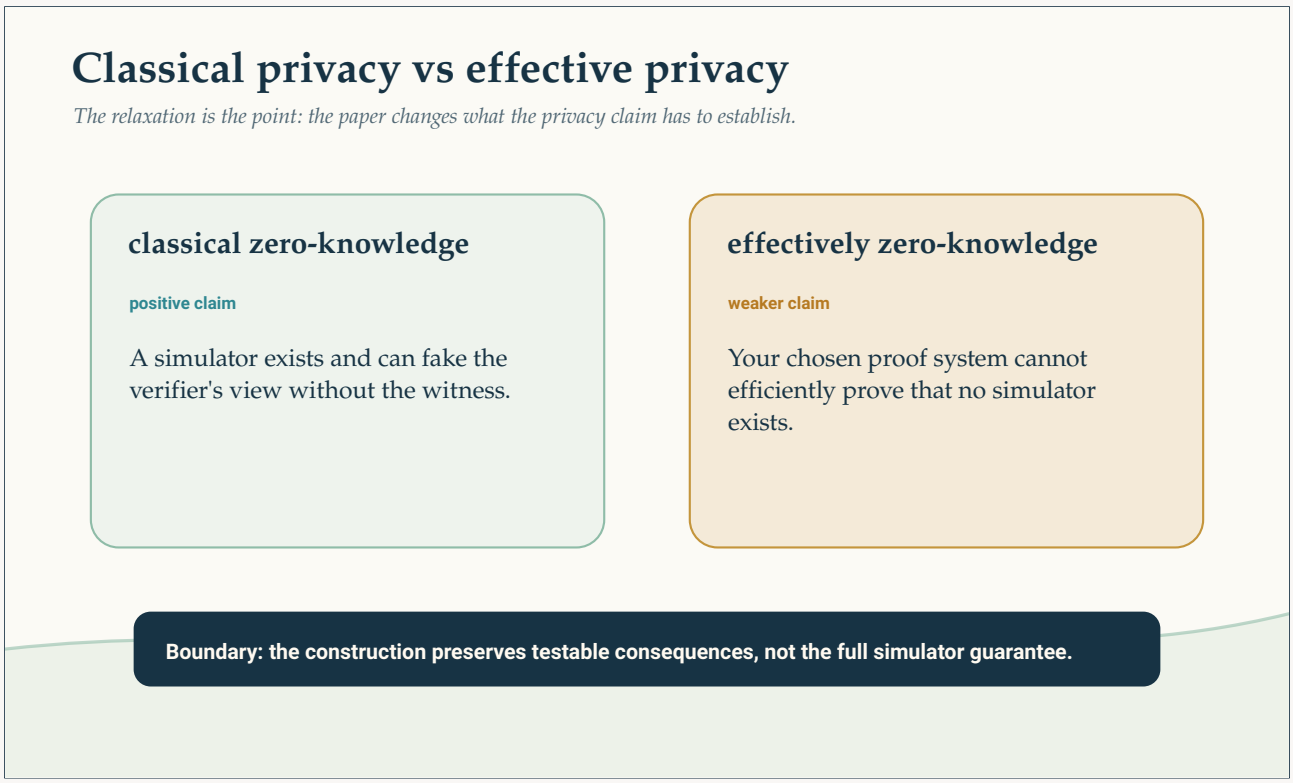
نخست، یک اثبات دانش صفر به گفت‌وگو نیاز دارد. اگر آلیس فقط یک پیام بفرستد و هیچ راه‌اندازی مورد اعتمادی از پیش ترتیب داده نشده باشد، تضمین دانش صفر فرو می‌ریزد — و این مستقل از آن است که حاضر باشید در عوض چه مقدار از درستی را قربانی کنید.

دوم، یک اثبات دانش صفر به اندکی تحمل خطا نیاز دارد. مطالبهٔ درستی کامل به شکل آرامی تعامل را نیز از بین می‌برد: وارسی‌کننده‌ای که هرگز نمی‌توان فریبش داد، فارغ از اینکه چه انتخاب‌های تصادفی‌ای می‌کند، می‌تواند همان انتخاب‌ها را از پیش ثابت کند — و وقتی وارسی‌کننده قابل پیش‌بینی شود، آلیس می‌تواند به همه‌چیز در یک پیام پاسخ دهد، دقیقاً همان حالتی که از قبل شکسته بود.

مقالهٔ Ilango Rahul راهی برای دور زدن این دیوار دوگانه بررسی می‌کند. نه با وانمود کردن اینکه دیوار وجود ندارد، و نه با ساختن دانش صفر کلاسیک در محیطی که ناممکن است. حرکت ظریف‌تر است: معنای «هیچ چیز آشکار نمی‌شود» را ضعیف کنید، اما به شکلی که ویژگی‌های امنیتی‌ای را حفظ کند که رمزنگاران واقعاً می‌توانند آزمایش کنند.

نتیجه **دانش صفر مؤثر (effectively zero-knowledge)** نام دارد.

دانش صفر پشت سه در بسته می‌ماند — تعامل، راه‌اندازی مورد اعتماد و درستی ناکامل. ساخت Ilango از در دیگری عبور می‌کند: کتاب قواعد نمی‌تواند به‌طور کارآمد وجود شبیه‌ساز را رد کند.



دانش صفر کلاسیک می‌پرسد آیا شبیه‌سازی وجود دارد؛ «دانش صفر مؤثر» فقط می‌پرسد آیا کتاب قواعد انتخاب‌شده می‌تواند به‌طور کارآمد ثابت کند که چنین شبیه‌سازی وجود ندارد. همین پرسش ضعیف‌تر است که اجازه می‌دهد ساخت، یک پیام، بدون راه‌اندازی و درستی کامل را نگه دارد.

آزمون قدیمی: یک شبیه‌ساز وجود دارد

روش کلاسیک برای صورت‌بندی دانش صفر از یک کمک‌کنندهٔ خیالی به نام **شبیه‌ساز (simulator)** استفاده می‌کند.

ایده این است: جین را تصور کنید که راز آلیس را **نمی‌داند.** اگر جین بتواند کاملاً به‌تنهایی اثبات‌هایی تولید کند که درست شبیه اثبات‌هایی به نظر برسند که باب از آلیس دریافت می‌کرد، آن‌گاه اثبات‌های آلیس چیز تازه‌ای به باب نیاموخته‌اند. جین از قبل می‌توانست بدون راز آلیس همان تجربه را جعل کند.

پس دانش صفر کلاسیک یک شبیه‌ساز واقعی می‌خواهد. باید الگوریتمی کارآمد وجود داشته باشد که بدون دانستن راز، اثبات‌های ظاهراً واقعی تولید کند — در اصطلاح فنی، بدون دانستن شاهد (*witness*)؛ در سودوکو، شاهد همان جدول حل‌شده است.

این تعریف قدرتمند است، اما دقیقاً همان جایی است که عدم امکان قدیمی ضربه می‌زند. شهود ماجرا چنین است. یک اثبات واقعاً غیرتعاملی فقط یک رشته است. وقتی باب آن رشته را داشته باشد، می‌تواند آن را به شخص دیگری نشان دهد: او توانایی اثبات گزاره برای دیگران را به دست آورده است، که از همان ابتدا چیزی بیش از «هیچ» به نظر می‌رسد. قضیه‌های کلاسیک این شهود را به عدم امکان‌های بالا دقیق می‌کنند.

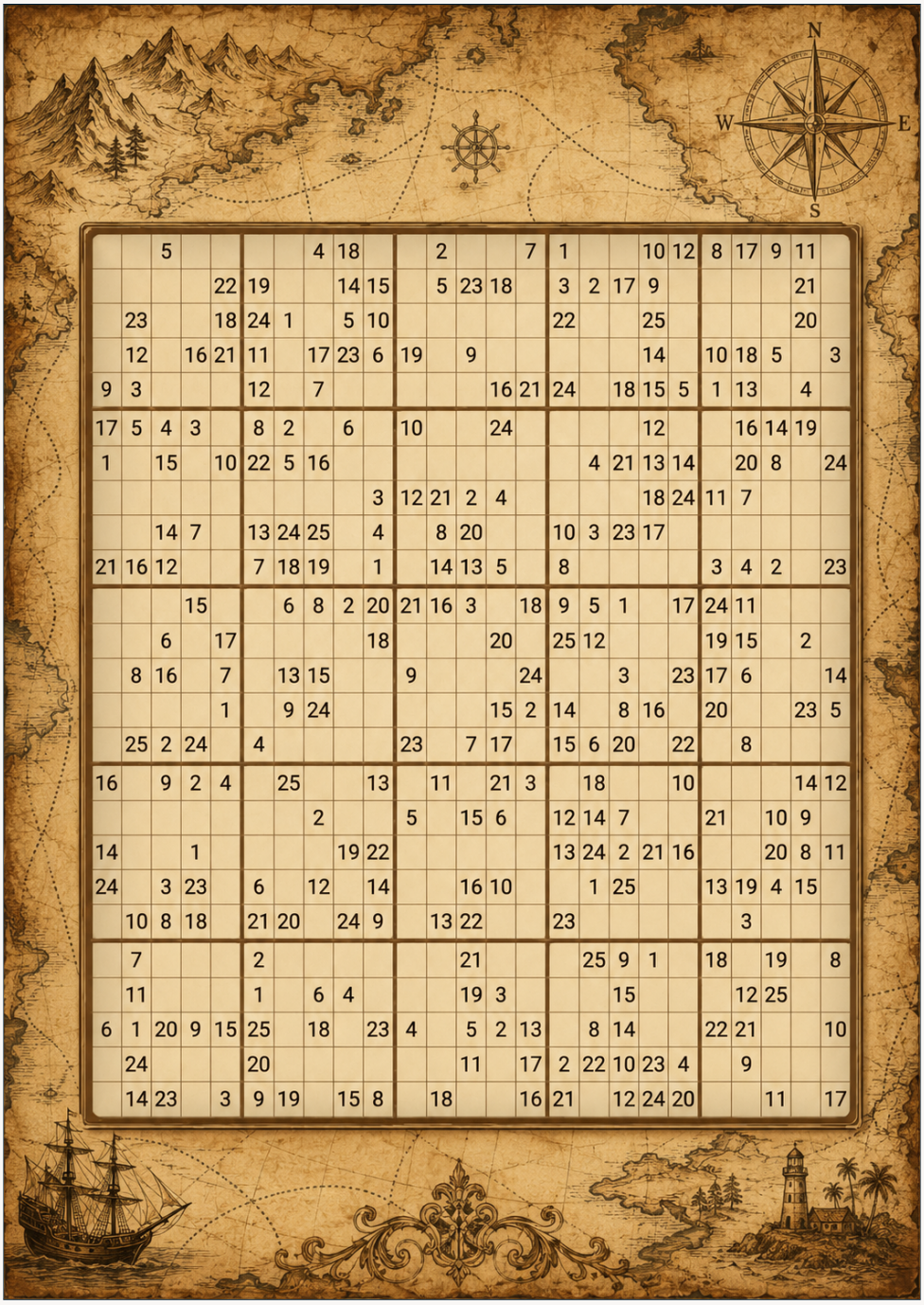
> **سه ویژگی‌ای که این مقاله بر آن‌ها اصرار دارد**
>
> می‌برد: نام را قید سه مقاله عنوان
>
> **بدون تعامل:** آلیس یک رشتهٔ اثبات می‌فرستد. هیچ پروتکل رفت‌وبرگشتی وجود ندارد.
> **بدون راه‌اندازی:** آلیس و باب به رشتهٔ مرجع مشترک مورد اعتماد یا تصادفی‌سازی عمومیِ از پیش ترتیب‌داده‌شده تکیه نمی‌کنند.
> بسیاری از سامانه‌هایی که «دانش صفر غیرتعاملی» نامیده می‌شوند همچنان به راه‌اندازی وابسته‌اند؛ این مقاله واقعاً راه‌اندازی صفر می‌خواهد.
> **درستی کامل:** یک گزارهٔ نادرست هیچ اثبات معتبری ندارد. نه اینکه «تقریباً هرگز پذیرفته نمی‌شود»؛ اصلاً اثبات معتبر وجود ندارد.
> این سه ویژگی دقیقاً همان چیزهایی‌اند که ریاضیات معمولِ نوشته‌شده دارد — و همان‌طور که بالا توضیح داده شد، دانش صفر کلاسیک نمی‌تواند هر سه را حفظ کند.

تفاوت، در قالب یک مگا-سودوکو

برای حس کردن تفاوت، این تصویر عمداً ساده‌شده را در نظر بگیرید.

برای بخش جدی قیاس از سودوکوی معمولی ۹ در ۹ استفاده نکنید. بیش از حد کوچک و محدود است: رایانه می‌تواند آن را حل کند یا ثابت کند راه‌حل ندارد. در عوض خانواده‌ای از معماهای **MegaSudoku(n)** را تصور کنید. قانون معمول را مقیاس دهید: اندازهٔ بلوک را n بگیرید، $N = n^2$ تعریف کنید و جدولی N در N بسازید که به بلوک‌های n در n تقسیم شده و N نماد دارد. سودوکوی معمولی فقط حالت کوچک $n = 3$ و $N = 9$ است: جدول ۹ در ۹، بلوک‌های ۳ در ۳ و نه نماد. داستان پیچیدگی اثبات تازه وقتی شروع می‌شود که اجازه دهیم n رشد کند و جدول بتواند ابزارک‌های اضافی (gadgets) حمل کند که آن را شبیه یک فرمول SAT در لباس سودوکو می‌کنند. فرمول SAT فقط فهرستی از قیود بله/خیر است: آیا می‌توانید به متغیرها مقادیر درست/نادرست بدهید به‌طوری

که همهٔ قیود ارضا شوند؟



یک سودوکوی ۲۵ در ۲۵: می‌توان قواعدش را بدون آشکار کردن جدول کامل‌شده وارسی کرد — جانشینی تصویری برای اثباتی که یک راه‌حل پنهان، یعنی شاهد، را تأیید می‌کند.

سودوکو و SAT: یک معما با دو لباس

جهت و می‌شود، انجام جهت دو هر در ترجمه نیست. استعاره کند» رفتار SAT فرمول یک «مانند می‌تواند سودوکو که ادعا این نوشت. کامل می‌توان را آسان

از سودوکو به SAT. فقط با درست/نادرست حرف می‌زند، پس برای هر سه‌تایی (سطر، ستون، مقدار) یک متغیر بولی تعریف کنید: $x(r,c,v)$ یعنی «سلول در سطر r و ستون c مقدار v دارد». یک سودوکوی ۴ در ۴ (بلوک‌های ۲ در ۲، مقادیر ۱ تا ۴) به $4 \cdot 4 \cdot 4 = 64$ متغیر نیاز دارد؛ سودوکوی کلاسیک ۹ در ۹ به 729. سپس هر قانون سودوکو به مجموعه‌ای از بندها (clauses) تبدیل می‌شود. (یک بند OR چند متغیر یا نقیض آن‌هاست؛ کل فرمول AND همهٔ بندهاست.)
هر سلول دست کم یک مقدار دارد — یک بند برای هر سلول:

$$x(1,1,1) \vee x(1,1,2) \vee x(1,1,3) \vee x(1,1,4)$$

هر سلول حداکثر یک مقدار دارد — برای هر جفت مقدار، یک بند «هر دو نه»:

$$\neg x(1,1,1) \vee \neg x(1,1,2) \quad \neg x(1,1,1) \vee \neg x(1,1,3) \quad \dots$$ و به همین ترتیب برای هر شش جفت.

هر سطر همهٔ مقادیر را دارد — برای سطر 1 و مقدار 3: دست کم یک بار،

$$x(1,1,3) \vee x(1,2,3) \vee x(1,3,3) \vee x(1,4,3)$$

و حداکثر یک بار: $\neg x(1,1,3) \vee \neg x(1,2,3)$، و به همین ترتیب برای هر جفت سلول در سطر.
ستون‌ها و بلوک‌ها — مجموعه‌هایی همانند، فقط گروه سلول‌ها تغییر می‌کند. برای بلوک بالا-چپ و مقدار 2:

$$x(1,1,2) \vee x(1,2,2) \vee x(2,1,2) \vee x(2,2,2)$$

به‌علاوهٔ بندهای جفتی «هر دو نه».
سرنخ‌های چاپ‌شده — ساده‌ترین بخش: هر سرنخ بندی با یک متغیر واحد است. عدد 3 چاپ‌شده در گوشهٔ بالا-چپ به این بند تبدیل می‌شود:

$$x(1,1,3)$$

همهٔ این‌ها دقیقاً زمانی ارضاپذیر است که سودوکو راه‌حل داشته باشد — و یک انتساب ارضاکننده خودِ راه‌حل است: ببینید کدام‌های $x(r,c,v)$ درست‌اند و جدول را پر کنید. برای سودوکوی ۹ در ۹، این یعنی 729 متغیر و چند هزار بند که یک حل‌کنندهٔ SAT امروزی در چند میلی‌ثانیه حل می‌کند. به بند سرنخ $x(1,1,3)$ توجه کنید: می‌گوید «این سلول دقیقاً برابر 3 است»، نه «این سلول‌ها همگی متفاوت‌اند» — همان نامتقارنی‌ای که در یادداشت پروتکل پایین‌تر، ترفند اضافی برای سلول‌های سرنخ را لازم می‌کند.
از SAT به سودوکو. مقاله به جهت مخالف و دشوارتر نیاز دارد: با داشتن یک فرمول SAT دلخواه، یک مگا-سودوکو بسازید که دقیقاً زمانی راه‌حل داشته باشد که فرمول ارضاپذیر باشد. قواعد بومی سودوکو فقط می‌توانند بگویند «این سلول‌ها همگی متفاوت‌اند»، پس قیود منطقی دلخواه باید ساخته شوند — و ابزارک‌ها دقیقاً همین‌اند. ابزارک خوشه‌ای کوچک و از پیش طراحی‌شده از سلول‌هاست، یکی برای هر بند فرمول، که در آن سلول‌های مشخص نقش متغیرها را بازی می‌کنند (نمادی که در خود نگه می‌دارند درست یا نادرست را کد می‌کند) و قیود داخلی خوشه طوری مهندسی شده‌اند که تنها پرشدن‌های قانونی آن با انتساب‌هایی متناظر باشند که آن بند را ارضا می‌کنند. این کار دستی استاندارد در اثبات‌های کامل-NP بودن است؛ برای سودوکوی تعمیم‌یافته Yato و Seta در سال ۲۰۰۳ آن را انجام دادند.
این دو جهت در کنار هم می‌گویند سودوکوی N در N و SAT یک مسئله‌اند که دو لباس متفاوت پوشیده‌اند. همین است که به این مقاله — و خود پژوهش — اجازه می‌دهد با جدول‌ها و نمادها دربارهٔ تمام NP داستان بگوید.

شاهد هنوز به‌راحتی قابل تصور است. آلیس یک پرکردن کامل و معتبرِ مگا-سودوکو را می‌داند. باب می‌خواهد قانع شود چنین پرکردنی وجود دارد، اما آلیس نمی‌خواهد آن را فاش کند. اگر کل جدول را بفرستد، باب قانع می‌شود، ولی راز از بین رفته است.

در نسخهٔ کلاسیک دانش صفر، آلیس و باب تعامل می‌کنند. یک مدل ذهنی قدیمی از مهره‌های پوشیده استفاده می‌کند. آلیس جدول حل‌شده را پنهان می‌کند، پیش از هر دور نام نمادها را محرمانه عوض می‌کند و به باب اجازه می‌دهد یکی از قیود محلی را به‌طور تصادفی بررسی کند: یک سطر، یک ستون، یک بلوک یا یک ابزارک. اگر سلول‌های بازشده نمادهایی همگی متفاوت نشان دهند، اطمینان باب بیشتر می‌شود. سپس همه‌چیز دوباره پوشیده می‌شود و نمادها با یک نام‌گذاری تازه عوض می‌شوند. (یک پیچیدگی وجود دارد: سرنخ‌های داده‌شدهٔ معما به ترفند دیگری نیاز دارند، چون تغییر نام نمادها آن‌ها را نیز پنهان می‌کند. یادداشت زیر توضیح می‌دهد پروتکل‌های کلاسیک چگونه این را حل می‌کنند؛ برای ادامه همین تصویر ساده کافی است.)

پروتکل‌های کلاسیک واقعاً با سلول‌های سرنخ چه می‌کنند

متفاوت همگی و متفاوت‌اند»، همگی سلول‌ها «این می‌گویند همگی بلوک و ستون سطر، قواعد دارد. کور نقطهٔ یک نام تغییر ترفند $\sigma(5)$ فقط باب نام تغییر از پس و است»، 5 دقیقاً سلول «این می‌گوید سرنخ یک اما می‌ماند. برقرار همچنان نامی تغییر هر از پس برطرف نقص این اگر کند. وارسی را چیزی نمی‌تواند نتیجه در بداند. را σ نام تغییر بی‌آنکه می‌بیند، را — پوشانده‌شده نمادی — است، گرفته نادیده کاملاً را چاپ‌شده سرنخ‌های که حالی در دارد، وجود معتبر جدول یک کند ثابت می‌تواند آلیس نشود، دارد. استاندارد راه‌حل دو کلاسیک ادبیات نمی‌کند. ثابت چیزی معما این دربارهٔ این

پالت. یک سطر اضافی با N سلول به جدول پنهان اضافه کنید — پالتی که آلیس آن را با نمادهای 1…N در ترتیب عمومی و ثابتی پر می‌کند و سپس همراه با بقیه تغییر نام می‌دهد، بنابراین شامل $\sigma(1)…\sigma(N)$ می‌شود. حالا چالش تصادفی باب یک گزینهٔ اضافه دارد. افزون بر انتخاب یک سطر، ستون، بلوک یا ابزارک برای باز کردن، می‌تواند **پالت به‌علاوهٔ یک سلول سرنخ** را انتخاب کند. آلیس هر دو را آشکار می‌کند؛ پالت تغییر نام همان دور را نشان می‌دهد و باب بررسی می‌کند که سلول سرنخ دقیقاً نسخهٔ تغییرنام‌یافتهٔ سرنخ چاپ‌شده را نشان دهد. این همچنان دانش صفر می‌ماند، چون باب فقط σ را می‌آموزد — که در هر دور تازه و تصادفی انتخاب می‌شود و به‌تنهایی ارزشی ندارد — و مقدار سلولی را که از خود معما از قبل می‌دانست. هیچ اطلاعاتی از سلول‌های محرمانه نشت نمی‌کند و شبیه‌ساز می‌تواند با انتخاب یک σ تصادفی همان نما را جعل کند. پروتکل درست است، چون آلیس متقلب در هر دور با احتمالی ثابت گیر می‌افتد و دورها آن‌قدر تکرار می‌شوند که تردید ناچیز شود.

کامپایل کردن سرنخ‌ها. نسخه‌ای ساختاری‌تر به‌جای افزودن چالش ویژه، آن را حذف می‌کند. به‌جای وارسی مقدار سرنخ، آن را با قیود نابرابری اجبار کنید: سلول سرنخ را به همهٔ سلول‌های پالت جز سلولی که مقدار خودش را حمل می‌کند متصل کنید — «متفاوت از $\sigma(1)$، متفاوت از $\sigma(2)$، …، متفاوت از همه‌چیز جز $\sigma(5)$». تنها نمادی که سلول می‌تواند قانوناً داشته باشد همان سرنخ است. اکنون همهٔ قیود دوباره از نوع «این دو متفاوت‌اند» شده‌اند — زیر تغییر نام ناوردا و دقیقاً مانند یک سطر قابل وارسی. این همان مانوری است که برای رأس‌های از پیش رنگ‌شده در پروتکل کلاسیک رنگ‌آمیزی گراف استفاده می‌شود و روح همان واژهٔ ابزارک در بالا را دارد: در تصویر مگا-سودوکو-به‌مثابه-SAT، سرنخ‌ها نیز مانند هر قید دیگری به ابزارک‌های نابرابری کامپایل می‌شوند.

پروتکل فیزیکی. پروتکل واقعی کارتی برای سودوکو (Gradwohl، Pinkas Naor و Rothblum، ۲۰۰۷) اصلاً از تغییر نام استفاده نمی‌کند و سرنخ‌ها را پیش از شروع پنهان‌سازی حل‌وفصل می‌کند. آلیس برای هر سلول سه کارت یکسان با مقدار همان سلول می‌گذارد — برای سلول‌های محرمانه پشت‌ورو، اما برای **سلول‌های سرنخ رو به بالا** تا باب پیش از برگرداندن کارت‌ها با چشم خود ببیند سرنخ‌ها رعایت شده‌اند. سپس یک کارت از هر سلول وارد بستهٔ سطرش، یکی وارد بستهٔ ستونش و یکی وارد بستهٔ بلوکش می‌شود؛ هر بسته بر زده و آشکار می‌شود و باب بررسی می‌کند همهٔ N نماد را داشته باشد. بر زدن اطلاعات مکان را از بین می‌برد (این همان دانش صفر است)، اما سرنخ‌ها از همان لحظهٔ چیدن کارت‌ها تثبیت شده بودند.

در هر دو حالت، درس همان چیزی است که این مقاله بارها به آن بازمی‌گردد: پروتکل دانش صفر حسابداری دقیقی است از اینکه کدام واقعیت‌ها از پنهان‌سازی جان سالم به در می‌برند. تغییر نام «همگی متفاوت» را حفظ می‌کند و «برابر 5 است» را پاک می‌کند — پس «برابر 5 است» باید از راه دیگری دوباره وارد شود.

این پروتکلِ مقاله نیست. فقط مدل ذهنی دانش صفر کلاسیک است:

- آلیس و باب رفت‌وبرگشت دارند.

- باب وارسی‌های تصادفی انتخاب می‌کند.
- آلیس فقط سازگاری محلی را آشکار می‌کند، نه کل راه‌حل را.
- اثبات حریم خصوصی با نشان دادن این کار می‌کند که نمای باب می‌توانست بدون راه‌حل محرمانهٔ آلیس تولید شود.

پس دانش صفر کلاسیک حول یک واقعیت مثبت ساخته شده است:

یک شبیه‌ساز واقعاً وجود دارد.

حالا بخش‌های راحت را حذف کنید. آلیس یک رشتهٔ اثبات می‌فرستد و می‌رود. هیچ راه‌اندازی مورد اعتمادی، هیچ رشتهٔ تصادفی مشترکی که از پیش آماده شده باشد، وجود ندارد و باب هرگز نباید یک معمای نادرست را بپذیرد. این همان محیطی است که دانش صفر کلاسیک در آن دوام نمی‌آورد.

پیش از ترفند به یک شخصیت دیگر نیاز داریم. یک **کتاب قواعد** ثابت کنید: یک سامانهٔ اثبات رسمی (proof system) به معنای منطقی — مجموعه‌ای ثابت از اصول موضوعه به‌علاوهٔ قواعد مکانیکی برای وارسی اثبات‌های ریاضی نوشته‌شده. ZFC، اصول موضوعهٔ استاندارد ریاضیات، مثال شاخص است. از اینجا به بعد همه‌چیز نسبت به کتاب قواعدی بیان می‌شود که از پیش انتخاب شده، و انتخاب انعطاف‌پذیر است: ساخت برای هر کتاب قواعدی که ثابت کنید، از جمله ZFC، کار می‌کند.

(یادداشتی دربارهٔ واژه‌ها، برگرفته از خود مقاله: «سامانهٔ اثبات» در اینجا همیشه به همین کتاب قواعد اشاره دارد — سامانهٔ رسمی‌ای که اثبات‌های ریاضی را وارسی می‌کند — و هرگز به پیام‌هایی که آلیس می‌فرستد اشاره نمی‌کند. ماشین آلیس و باب «اثبات‌کننده و وارسی‌کننده» نامیده می‌شود.)

نسخهٔ گودلی داستان مگا-سودوکو را حفظ می‌کند، اما اثبات را تغییر می‌دهد.

یک سامانهٔ قید دوم با همان اندازهٔ نمایش‌داده‌شده انتخاب کنید و نامش را **D** بگذارید. در داستان، S و D دو معمای MegaSudoku(n) با قالب یکسان‌اند. پشت صحنه، D ممکن است در اصل یک فرمول منطقی دشوار با اندازه‌ای متفاوت بوده باشد؛ اگر لازم باشد می‌توان با قیود ساختگی بی‌ضرر آن را پُر کرد تا در همان جدول جا شود. D از یک فرمول منطقی ساخته می‌شود که واقعاً **ارضاناپذیر (unsatisfiable)** است: هیچ انتسابی از مقادیر وجود ندارد که همهٔ قیودش را درست کند، درست مانند معمای شکسته‌ای که هیچ جدول کامل قانونی ندارد. نمونهٔ اسباب‌بازی فرمولی است که هم‌زمان می‌خواهد «X درست است» و «X نادرست است». پس D هیچ پرکردن معتبری ندارد.

اما D نباید معمای شکسته‌ای باشد که افشای شکستنش آسان است. مثال اسباب‌بازی بالا بد است: هر کتاب قواعدی «X و نقیض X» را در یک خط رد می‌کند. D باید به شکلی نادرست باشد که کتاب قواعد انتخاب‌شده نتواند با استدلالی کوتاه آن را گواهی کند. اگر کتاب قواعد می‌توانست D را با اثباتی کوتاه رد کند، داستان زیر فرو می‌ریخت: مسیر جایگزینی که شاید بدون راز آلیس اثبات تولید می‌کرد می‌توانست رسماً حذف شود و همراه آن تضمین حریم خصوصی نیز از بین می‌رفت. بنابراین D از خانواده‌ای انتخاب می‌شود که کتاب قواعد ثابت نمی‌تواند آن را به‌طور کارآمد رد کند: درون همان کتاب قواعد هیچ اثبات کوتاهی وجود ندارد که نشان دهد D راه‌حل ندارد.

اثبات تک‌پیامی آلیس حالا دربارهٔ یک گزارهٔ یا/یا است:

یا مگا-سودوکوی واقعی S راه‌حل دارد، یا طعمهٔ D راه‌حل دارد.

این پیوند منطقی است. D به شیوه‌ای جادویی ساخته **نمی‌شود** که S را درست کند. استدلال اثبات این نیست که «D راه‌حل ندارد، پس S راه‌حل دارد». چیزی که اثبات می‌شود فصل منطقی **S یا D** است. درستی کامل می‌گوید یک فصل منطقی نادرست نمی‌تواند اثبات معتبر داشته باشد. چون D در واقعیت نادرست است — راه‌حل ندارد — تنها راه درست بودن فصل این است که S درست باشد. پس اگر اثبات پذیرفته شود، S باید راه‌حل داشته باشد. طعمه نمی‌تواند S نادرست را درست کند.

اما برای بخش شبیه دانش صفر بپرسید اگر D راه‌حل داشت چه می‌شد. آن راه‌حل طعمه به‌عنوان شاهدی جایگزین عمل می‌کرد. به کسی

اجازه می‌داد بدون دانستن راه‌حل واقعی مگا-سودوکوی آلیس اثبات تولید کند — یعنی یک شبیه‌ساز. در واقعیت D راه‌حل ندارد، پس این مسیر شبیه‌سازی بسته است. نکته این است که کتاب قواعد نمی‌تواند به‌طور کارآمد ثابت کند که بسته است.

پس D دو وظیفه دارد. برای **درستی (soundness)**، D نادرست است، بنابراین اثبات معتبر S» یا «D ما را ناگزیر به S می‌رساند. برای **دانش صفر مؤثر**، رد کردن D سخت است، بنابراین کتاب قواعد نمی‌تواند به‌سرعت مسیر طعمه‌ای را که امکان شبیه‌سازی می‌داد کنار بزند.

پس آزمون امنیت دیگر این نیست:

آیا می‌توانیم ثابت کنیم که یک شبیه‌ساز واقعاً وجود دارد؟

بلکه می‌شود:

آیا کتاب قواعد شما می‌تواند به‌طور کارآمد ثابت کند که شبیه‌ساز ناممکن است؟

اگر پاسخ منفی باشد، نتیجه‌ای شگفت‌آور و قوی به دست می‌آید: هر تضمین امنیتی که (الف) با اجرای یک آزمون قابل مشاهده باشد و (ب) درون همان کتاب قواعد، به‌طور اثبات‌پذیر از وجود شبیه‌ساز نتیجه شود، واقعاً برقرار است. یک حملهٔ موفق علیه هر یک از آن‌ها خودش به همان ابطال کوتاهِ گمشده تبدیل می‌شد، و آن ابطال کوتاه وجود ندارد. این بخش «مؤثر» در دانش صفر مؤثر است.

پس تضاد کلاسی چنین است:

دانش صفر کلاسیک: اثبات‌ها امن‌اند چون یک شبیه‌ساز وجود دارد.

دانش صفر مؤثر به سبک گودل: برای آزمون‌های امنیتی قابل مشاهده، اثبات‌ها امن تلقی می‌شوند چون کتاب قواعد نمی‌تواند به‌طور کارآمد ثابت کند شبیه‌ساز ناممکن است.

ادعای دوم ضعیف‌تر است. همین ضعیف‌تر بودن نیز به مقاله اجازه می‌دهد سه ویژگی‌ای را حفظ کند که نسخهٔ کلاسیک را شکستند: یک پیام، بدون راه‌اندازی و درستی کامل.

آزمون جدید: نمی‌توانید ثابت کنید شبیه‌ساز غایب است

آرام‌سازی Ilango پرسش را تغییر می‌دهد.

دانش صفر کلاسیک می‌پرسد:

آیا یک شبیه‌ساز وجود دارد؟

دانش صفر مؤثر چیزی ضعیف‌تر می‌پرسد:

آیا کتاب قواعد انتخاب‌شدهٔ شما می‌تواند به‌طور کارآمد ثابت کند هیچ شبیه‌سازی وجود ندارد؟

این ممکن است یک فرار فنی به نظر برسد، اما ایدهٔ مرکزی همین است. ساخت در وضعیتی عجیب زندگی می‌کند: شبیه‌ساز در واقع واقعاً وجود **ندارد** — مقاله در این باره صریح است — اما کتاب قواعدی که ثابت کرده‌اید نمی‌تواند به‌طور کارآمد نبودنش را ثابت کند. اگر هر پیامد بدی که برایتان مهم است به چنین ابطالی نیاز داشته باشد، سامانه برای همان پیامدها همچنان مانند دانش صفر رفتار می‌کند.

اینجاست که گودل وارد می‌شود. نه به‌عنوان تزئین، و نه با ادعای «گودل رمزنگاری را امن می‌کند». پیوند از نظریهٔ اثبات می‌آید. یک کتاب قواعد بهینه نامیده می‌شود اگر به معنایی دقیق بهترین سامانهٔ ممکن باشد: هرگاه هر کتاب قواعدی بتواند فرمولی از نوع مربوطه را با اثباتی کوتاه رد کند، کتاب قواعد بهینه نیز بتواند آن را با اثباتی حداکثر چندجمله‌ای بلندتر رد کند. Krajíček و Pudlák در سال ۱۹۸۹ حدس زدند که **هیچ سامانهٔ اثبات بهینه‌ای وجود ندارد**: هر کتاب قواعدی را که ثابت کنید، کتاب قواعد دیگری وجود دارد که بعضی خانواده‌های گزاره‌های درست را بسیار موجزتر اثبات می‌کند. این یکی از حدس‌های باز مرکزی در پیچیدگی اثبات است و خویشاوند محدود و پیچیدگی‌محور قضیهٔ ناتمامیت گودل محسوب می‌شود: بعضی گزاره‌های درست در کتاب قواعد انتخاب‌شدهٔ شما اثبات کوتاه ندارند — نه چون اصولاً اثبات‌ناپذیرند، بلکه چون هر کتاب قواعد ثابت برخی حقیقت‌های کوتاه را بدون اثبات کوتاه باقی می‌گذارد.

مقاله این حدس را فرض می‌کند (در نسخه‌ای اندکی قوی‌تر از نوع «بی‌نهایت بار» یا infinitely ،often که هنگام استفادهٔ رمزنگارانه از حدس‌ها متداول است). حاصل، طبق قضیه‌ای از Krajíček و ،Pudlák ملموس است: برای هر کتاب قواعدی دنباله‌ای از فرمول‌ها وجود دارد که واقعاً ارضاناپذیرند، اما کتاب قواعد نمی‌تواند آن‌ها را با اثبات‌های کوتاه رد کند — و مهم‌تر اینکه یک الگوریتم کارآمد می‌تواند آن‌ها را تولید کند. ویژگی آخر، یعنی یکنواختی ،(unfiormity) کل ایده را از یک ادعای صرفاً وجودی به الگوریتم واقعی‌ای تبدیل می‌کند که آلیس می‌تواند اجرا کند: طعمه‌های D از خط تولید بیرون می‌آیند، نه از هیچ.

حرکت رمزنگارانه این است که همین کمبود توان اثبات را به کار بگیریم.

ساخت در حال انجام چه کاری است

شکل کلی ساخت مقاله چنین است.

یک کتاب قواعد ثابت کنید — مثلاً .ZFC تحت فرض پیچیدگی اثبات، دنباله‌ای با قابلیت تولید کارآمد از فرمول‌ها وجود دارد که واقعاً ارضاناپذیرند، اما کتاب قواعد هیچ اثبات کوتاهی برای ارضاناپذیری آن‌ها ندارد.

حالا یک اثبات تک‌پیامی با این شکل بسازید:

> یا گزارهٔ واقعی ارضاپذیر است، یا این فرمول سخت ویژه ارضاپذیر است.

فرمول سخت ویژه ارضاپذیر نیست. بنابراین اگر ماشین اثبات زیرین درستی کامل داشته باشد، پذیرفته شدن پیام همچنان یعنی گزارهٔ واقعی درست است. این درستی کامل را می‌دهد.

اما برای امنیت شبیه دانش صفر، تصور کنید فرمول سخت ویژه ارضاپذیر بود. آن‌وقت شاهد آن می‌توانست برای شبیه‌سازی اثبات‌ها بدون دانستن شاهد واقعی استفاده شود. فرمول در واقعیت ارضاپذیر نیست — اما کتاب قواعد نمی‌تواند این را به‌طور کارآمد ثابت کند. بنابراین نمی‌تواند به‌طور کارآمد ثابت کند شبیه‌ساز ناممکن است.

این همان لولاست. سامانه راز را با تولید یک شبیه‌ساز کلاسیک پنهان نمی‌کند. برای طبقهٔ بزرگی از آزمون‌های امنیتی قابل مشاهده، راز را پشت ناتوانی کتاب قواعد در گواهی کردن غیبت شبیه‌ساز پنهان می‌کند.

مقاله چه ادعایی دارد

قضیهٔ اصلی چند لایه دارد. نتیجهٔ هسته‌ای این است:

تحت یک فرض استاندارد رمزنگاری — وجود **اثبات‌های غیرتعاملی با تمایزناپذیری شاهد witness (non-interactive**

indistinguishable proofs)، اشیایی به‌خوبی مطالعه‌شده که از چند بستهٔ مفروضات شناخته‌شده نتیجه می‌شوند — و تحت حدس پیچیدگی اثبات که **هیچ سامانهٔ اثبات بهینه‌ای (در نسخهٔ infinitely often) وجود ندارد**، مقاله برای هر انتخاب کتاب قواعد یک اثبات‌کننده و وارسی‌کنندهٔ تک‌پیامی برای NP/SAT با **درستی کامل** و بدون راه‌اندازی می‌سازد که نسبت به همان کتاب قواعد دانش صفر مؤثر است. NP/SAT) همان «مخرج مشترک سخت» استاندارد برای مسائل شبیه معماست؛ مگا-سودوکو یکی از لباس‌های آن است.)

برای ادعای گسترده‌تر دربارهٔ حفظ ویژگی‌های امنیتی ابطال‌پذیر، مقاله یک فرض استاندارد دیگر می‌افزاید: باور حذف تصادفی‌سازی **P = BPP** (تقریباً یعنی تصادفی بودن هیچ قدرت ضروری اضافه‌ای به الگوریتم‌ها نمی‌دهد).

اگر زبان قضیه را کنار بگذاریم:

- اثبات یک پیام است.
- هیچ راه‌اندازی مورد اعتمادی وجود ندارد.
- گزاره‌های نادرست قابل اثبات نیستند.
- اثبات‌کننده دانش صفر کلاسیک نیست — شبیه‌ساز ندارد.
- اما هر پیامد امنیتیِ ابطال‌پذیر و مبتنی بر بازی از دانش صفر کلاسیک می‌تواند در این محیط به دست آید.

«ابطال‌پذیر» مهم است. یعنی شکست امنیتی را می‌توان با اجرای یک مهاجم در یک بازی آزمایش کرد. بسیاری از تعریف‌های امنیتی رمزنگاری چنین شکلی دارند: آیا مهاجم می‌تواند دو رمزمتن را از هم تشخیص دهد، تابعی را وارون کند، شاهد را بازیابی کند یا در آزمایش مشخصی برنده شود؟ قضیه برای هر ویژگی ابطال‌پذیر، یکی‌یکی، یک اثبات‌کننده می‌دهد. احتمالاً داشتن یک اثبات‌کنندهٔ واحد که همهٔ ویژگی‌های ابطال‌پذیر را هم‌زمان داشته باشد ناممکن است — حملهٔ قدیمیِ استفادهٔ مجدد («باب می‌تواند اثبات را به دیگران نشان دهد») خودش یک ویژگی ابطال‌پذیر است و در اینجا واقعاً شکست می‌خورد. پیشنهاد مقاله این است که احتمالاً یک اثبات‌کنندهٔ واحد می‌تواند همهٔ ویژگی‌های ابطال‌پذیرِ طبیعی — آن‌هایی که واقعاً در عمل رمزنگاری ظاهر می‌شوند — را پوشش دهد، اما آن بخش قضیه‌ای شرطی است که بر مفهوم غیررسمی «طبیعی» به‌علاوهٔ یک حدس صریح تکیه دارد. تضمین متوجه شکست‌های قابل مشاهده است، نه هر معنای فلسفی یا شبیه‌سازی‌محور از محرمانگی.

یک نتیجهٔ فرعی ملموس ارزش نام بردن دارد: ساخت نخستین اثبات‌های غیرتعاملیِ **پنهان‌سازی شاهد (witness hiding)** با اثبات‌کنندهٔ یکنواخت را به دست می‌دهد — «اثباتِ داشتن راه‌حل یک معما به شما برای یافتن آن راه‌حل کمک نمی‌کند»، بدون تعامل و بدون راه‌اندازی — شیئی که با وجود ظاهر متواضعانه‌اش دهه‌ها در برابر ساخت مقاومت کرده بود.

این نتیجه چه نمی‌گوید

این بخشی است که متن را صادق نگه می‌دارد.

نمی‌گوید قضیه‌های قدیمی عدم امکان اشتباه بودند. ساخت با تغییر تعریف از آن‌ها دور می‌زند.

دانش صفر معمولی و کلاسیک با بدون تعامل، بدون راه‌اندازی و درستی کامل ارائه نمی‌کند. مقاله صریحاً می‌گوید اثبات‌کنندهٔ ساخته‌شده شبیه‌ساز ندارد.

به این معنا نیست که اثبات را نمی‌توان دوباره استفاده کرد. یک اثبات تک‌پیامی همچنان می‌تواند به فرد دیگری نشان داده شود؛ مقاله ویژگی‌هایی از جنس امکان انکار (deniability) را حفظ نمی‌کند. (دانش صفر غیرتعاملی با راه‌اندازی مورد اعتماد نیز همین محدودیت را دارد.)

به این معنا نیست که با یک پروتکل عملی و آمادهٔ استقرار روبه‌رو هستیم. این نظریهٔ پیچیدگی و مبانی رمزنگاری است. نتیجه به مفروضات

بزرگی از پیچیدگی اثبات و رمزنگاری وابسته است و ساخت دربارهٔ چیزی است که در اصل ممکن است.

گودل را به یک primitive جادویی امنیتی تبدیل نمی‌کند. پیوند با گودل از سامانه‌های اثبات، سامانه‌های اثبات بهینه و همتاهای محدود ناتمامیت می‌آید. شهود قابل استفاده این نیست که «ناتمامیت از گذرواژه‌تان محافظت می‌کند». بلکه این است: اگر یک کتاب قواعد نتواند به‌طور کارآمد ثابت کند شبیه‌ساز ناممکن است، حمله‌هایی که به چنین اثباتی نیاز دارند را می‌توان در سطح تعریف امنیت مسدود کرد.

با این همه چرا جالب است

رمزنگاری اغلب سختی را به ایمنی تبدیل می‌کند. تجزیه به عوامل اول سخت است، پس مفروضات سبک RSA مفید می‌شوند. مسائل شبکه‌ای سخت‌اند، پس رمزنگاری شبکه‌ای مفید می‌شود. اینجا سختی عجیب‌تر است: نه «محاسبهٔ راز دشوار است»، بلکه «اثبات اینکه یک شیء اثباتی خاص نمی‌تواند وجود داشته باشد دشوار است».

همین است که مقاله را نامعمول می‌کند. با اصول موضوعه و کتاب‌های قواعد تقریباً مانند منابع رمزنگارانه رفتار می‌کند. عدم امکان معمول می‌گوید میان درستی و شبیه‌سازی تنش وجود دارد. حرکت Ilango این تنش را پشت پرده‌ای از نظریهٔ اثبات می‌گذارد: شبیه‌ساز غایب است، اما سامانهٔ رسمی نمی‌تواند به‌طور کارآمد این غیبت را آشکار کند.

برای خواننده، بخش شگفت‌آور این نیست که این کار جای سامانه‌های دانش صفر امروز را خواهد گرفت. احتمالاً دست‌کم مستقیماً چنین نخواهد کرد. شگفتی این است که یک محدودیت در منطق ریاضی می‌تواند به شکل سازنده استفاده شود: نه فقط به‌عنوان دیوار، بلکه به‌عنوان نوعی پوشش.

قدرت شواهد چقدر است؟

این یک مقالهٔ قضیه‌محور است، پس «شواهد» معنایی متفاوت از مقالهٔ زیست‌شناسی یا اخترشناسی دارد. پرسش این نیست که آیا آزمایشی تکرار شده است. پرسش این است که آیا تعریف‌ها، مفروضات و زنجیرهٔ اثبات از ادعا پشتیبانی می‌کنند.

اثبات رسمی است و مقاله دربارهٔ مفروضاتش شفاف است. این مفروضات سرسری نیستند. اثبات‌های غیرتعاملی با تمایزناپذیری شاهد اشیای استانداردی در رمزنگاری‌اند و از چند مجموعهٔ مفروضات تثبیت‌شده نتیجه می‌شوند. حدس نبود سامانهٔ اثبات بهینه یک حدس مرکزی در پیچیدگی اثبات است. BPP = P یک باور استاندارد دربارهٔ حذف تصادفی‌سازی است که فقط برای قضیهٔ گسترده‌ترِ ویژگی‌های ابطال‌پذیر استفاده می‌شود.

مقاله همچنین استدلال می‌کند این مفروضات هزینهٔ درست‌اند، نه داربستی دلخواه: یک جهت معکوس را اثبات می‌کند که نشان می‌دهد آن‌ها اساساً ضروری‌اند — اگر اصولاً ساخت‌هایی از این نوع وجود داشته باشند، باید اثبات‌های غیرتعاملی با تمایزناپذیری شاهد وجود داشته باشند و (با فرض وجود توابع یک‌طرفهٔ استاندارد) هیچ سامانهٔ اثبات بهینه‌ای نباید وجود داشته باشد. افزون بر این، مفروضات «برد-برد» هستند: رد هر یک از آن‌ها به‌خودی‌خود کشفی بزرگ در پیچیدگی اثبات، رمزنگاری یا نظریهٔ پیچیدگی خواهد بود.

اما چون نتیجه شرطی است، اعتماد به آن نیز شرطی است. اگر این مفروضات شکست بخورند، تفسیر قضیه تغییر می‌کند. و حتی اگر مفروضات برقرار باشند، تضمین دانش صفر کلاسیک کامل نیست؛ نسخهٔ آرام‌شده و نظریه‌اثباتی مقاله است.

پس سطح اعتماد مناسب چنین است: اعتماد بالا به اینکه مقاله یک نتیجهٔ امکان‌پذیری شرطیِ منسجم برقرار می‌کند؛ اعتماد متوسط به اینکه مفروضاتش جهان رمزنگاری واقعی ما را توصیف می‌کنند؛ و اعتماد پایین به هر پیامد عملی فوری.

چرا مهم است

مقاله مسیری را باز می‌کند که قرار بود بسته باشد.

نظریهٔ کلاسیک می‌گوید: دانش صفر کامل بدون راه‌اندازی نمی‌تواند در یک پیام جا شود و نمی‌تواند درستی کامل داشته باشد. مقالهٔ Ilango می‌گوید: اگر پیامدهای دانش صفر را که در بازی‌های امنیتی قابل آزمایش‌اند بخواهیم، و اگر اجازه دهیم تعریف امنیت به چیزی وابسته باشد که یک کتاب قواعد می‌تواند یا نمی‌تواند به‌طور کارآمد رد کند، بخش بزرگی از رفتار مفید را می‌توان بازیابی کرد — با یک پیام، بدون راه‌اندازی و با درستی کامل.

این یک دستکاری کوچک در تعریف نیست. راه متفاوتی برای فکر کردن به تضمین‌های رمزنگاری است. به‌جای اینکه فقط بپرسیم چه چیزی وجود دارد، بپرسیم کتاب قواعد شما چه چیزی را می‌تواند کنار بزند. به‌جای اینکه اثبات‌ناپذیری را مزاحمتی فلسفی بدانیم، از آن به‌عنوان ساختار استفاده کنیم.

دنیای عملی شاید فردا تغییر نکند. اما نقشهٔ مفهومی تغییر می‌کند. اکنون معنایی رسمی وجود دارد که در آن «هیچ‌کس نمی‌تواند به‌طور کارآمد ثابت کند راز نشت کرده است» می‌تواند آن‌قدر قوی باشد که بسیاری از حفاظت‌های مبتنی بر بازی را که از «راز نشت نکرده است» می‌خواستیم بازیابی کند.

به همین دلیل نام گودل در عنوان جای دارد.

خلاصهٔ تمیز

اثبات‌های دانش صفر به اثبات‌کننده اجازه می‌دهند وارسی‌کننده را قانع کند یک گزاره درست است، بدون آشکار کردن شاهد. نتایج کلاسیک عدم امکان می‌گویند دانش صفر را نمی‌توان بدون راه‌اندازی در یک پیام فشرده کرد و نمی‌توان برای آن درستی کامل داشت. مقالهٔ Ilango Rahul این عدم امکان‌ها را رد نمی‌کند. مفهومی ضعیف‌تر تعریف می‌کند، دانش صفر مؤثر: به‌جای اینکه بخواهد شبیه‌ساز واقعاً وجود داشته باشد، می‌خواهد سامانهٔ اثبات انتخاب‌شده — کتاب قواعد رسمی‌ای مانند ZFC — نتواند به‌طور کارآمد ثابت کند که هیچ شبیه‌سازی وجود ندارد. تحت مفروضات مهم رمزنگاری (اثبات‌های غیرتعاملی با تمایزناپذیری شاهد) و پیچیدگی اثبات (نبود سامانهٔ اثبات بهینه)، مقاله برای NP/SAT اثبات‌کننده‌های تک‌پیامی، بدون راه‌اندازی و با درستی کامل می‌سازد که پیامدهای ابطال‌پذیر و مبتنی بر بازی دانش صفر را ویژگی‌به‌ویژگی به دست می‌آورند. یک اثبات‌کنندهٔ واحد که همهٔ چنین ویژگی‌های «طبیعی» را پوشش دهد توسعه‌ای بیشتر و تا حدی حدسی است — و پوشش دادن به معنای واقعی کلمه همهٔ ویژگی‌های ابطال‌پذیر احتمالاً ناممکن است، چون اثبات‌ها همچنان قابل استفادهٔ مجددند. نتیجه نظری و شرطی است، نه ایprimitive مستقرشده، اما راه تازه‌ای برای استفاده از اثبات‌ناپذیری نظریه‌اثباتی به‌عنوان منبعی رمزنگارانه نشان می‌دهد.

بررسی بی‌پرده

مقاله چه چیزی نشان می‌دهد: تحت مفروضات بیان‌شده می‌توان برای NP/SAT اثبات‌کننده‌های تک‌پیامی، بدون راه‌اندازی و با درستی کامل ساخت که نسبت به هر سامانهٔ اثبات انتخاب‌شده دانش صفر مؤثرند و هر پیامد ابطال‌پذیرِ مبتنی بر بازی از دانش صفر کلاسیک را به دست می‌آورند.

چه چیزی محتمل است اما بدون قید اثبات نشده: اینکه مفروضات لازم در پیچیدگی اثبات و رمزنگاری برقرار باشند. این‌ها مفروضات جدی و به‌خوبی مطالعه‌شده‌اند — و مقاله نشان می‌دهد اساساً همان‌قدر که کافی‌اند، ضروری نیز هستند — اما همچنان مفروضات‌اند.

چه چیزی را نشان نمی‌دهد: دانش صفر کلاسیک با بدون تعامل، بدون راه‌اندازی و درستی کامل؛ سامانه‌ای عملی و آمادهٔ استقرار؛ انکارپذیری یا غیرقابل‌استفاده‌بودن مجدد اثبات‌ها؛ یا اینکه قضیهٔ ناتمامیت گودل به‌تنهایی رمزنگاری را امن می‌کند.

محدودیت‌های اصلی: تضمین یک آرام‌سازی دانش صفر است؛ گسترده‌ترین نسخه به چند فرض وابسته است؛ ادعاهای مربوط به یک اثبات‌کنندهٔ جهان‌شمول واحد تا حدی حدسی می‌مانند؛ و نتیجه در درجهٔ اول بنیادی و نظری است.

خوانندهٔ عمومی چقدر باید اعتماد داشته باشد؟ اعتماد بالا به اینکه، اگر تعریف‌ها پذیرفته شوند، این یک نتیجهٔ مهم نظری و شرطی است. اعتماد متوسط به اینکه مفروضات واقعیت را درست توصیف می‌کنند. اعتماد پایین برای کاربرد عملی فوری. برداشت امن این است: مقاله عدم امکان‌های دانش صفر را نمی‌شکند؛ راهی تازه و نظریه‌اثباتی پیدا می‌کند تا از بخش‌هایی از آن‌ها که برای بسیاری از بازی‌های امنیتی مهم‌اند عبور کند.

منابع

2025/1296 ePrint IACR Ilango,
https://eprint.iacr.org/2025/1296

00058.2025.FOCS63196/1109.10 DOI ,2025 FOCS
https://doi.org/10.1109/FOCS63196.2025.00058