انجمن رمز ایران

قطب علمی رمز

رمزگذاری تمام همریخت

وحید یوسفی پور

دانشکده مهندسی برق- دانشگاه صنعتی شریف

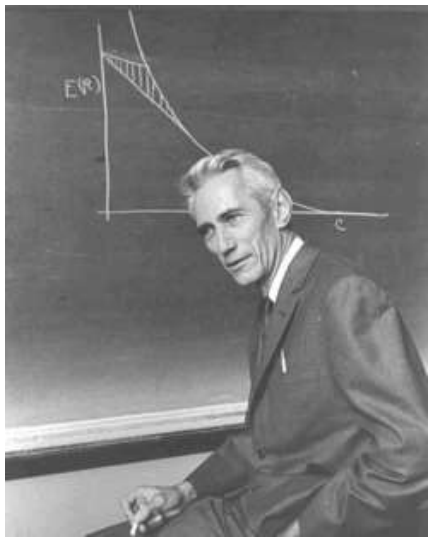
Yousefipoor_vahid@ee.sharif.edu

چشم‌انداز

- مقدمه ای بر رمزنگاری
- مفهوم همریختی و کاربرد آن در رایانش ابری
  - همریختی روی جمع و ضرب
  - مفهوم رمزگذاری تمام همریخت
  - کاربرد رمزگذاری تمام همریخت در رایانش ابری
- طرح رمزگذاری تمام همریخت
  - طرح جنتری
  - نسل دوم رمزگذاری تمام همریخت
  - پیاده سازی رمزگذاری تمام همریخت
  - پروژه PROCEED

مقدمه ای بر رمزنگاری

از سال ۴۰ پیش از میلاد تا ۱۹۴۹



کلود شانون

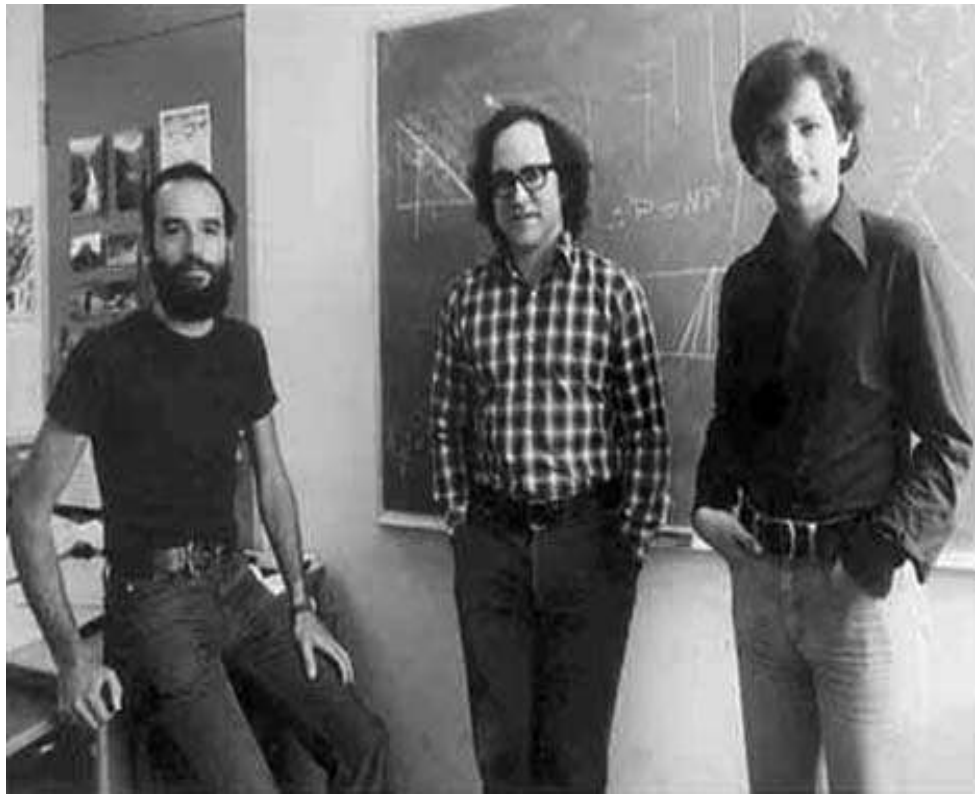
انیگما

ویجنر

ژولیوس سزار

رمزگذاری متقارن: بهره گیری از یک کلید برای رمز گذاری و رمزگشایی

دهه ۷۰ میلادی



آدلمن، ریوست، شامیر(۱۹۷۸)

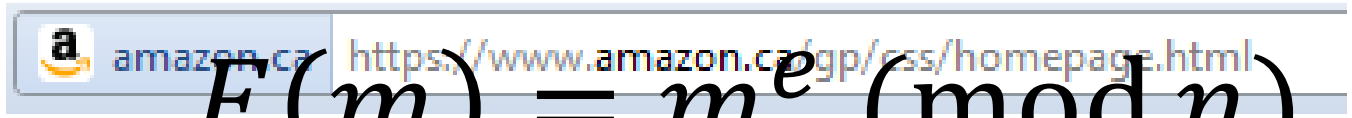
دیفی، هلمن و مرکل (۱۹۷۶)

رمزگذاری نامتقارن: بهره گیری از یک کلید برای رمزگذاری و یک کلید برای رمزگشایی

مقدمه ای بر رمزنگاری

رمزنگاری نامتقارن، ابزار اصلی برای امنیت داده در بستر اینترنت

الگوریتم RSA، پرکاربردترین و مشهورترین الگوریتم رمزنگاری نامتقارن



$$E(m) = m^e \pmod{n}$$

$$\mathrm{D}(c) = c^d \pmod{n}$$

Results courtesy of Ducas et al. (Link)

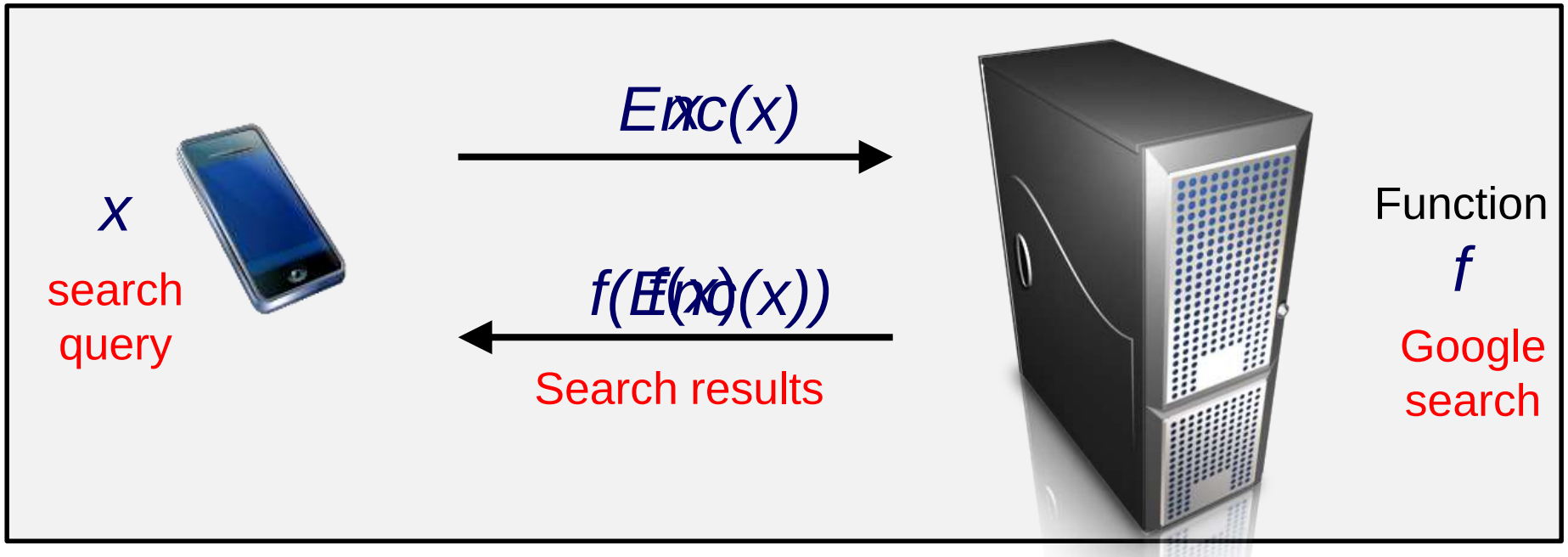
اما هنوز...

دنیا هنوز سیاه و سفید بود...

تنها عملیاتی که کاربر روی داده رمزشده اجرا می کرد

رمزگشایی آن بود...

محاسبه بر روی داده رمزشده



مفهوم همریختی و کاربرد آن در رایانش ابری

مفهوم همریختی

- هم ریختی نگاشتی حافظ عمل است.

$$\varphi: (G_1, *) \rightarrow (G_2, \diamond)$$
$$\forall a, b \in G_1 \rightarrow \varphi(a * b) = \varphi(a) \diamond \varphi(b)$$

$$E(m_1 * m_2) = E(m_1) * E(m_2)$$

همریختی روی ضرب

- هم ریختی RSA روی ضرب

$$E(m_1) = m_1^{\,e} \qquad E(m_2) = m_2^{\,e}$$

$$E(m_1) \times E(m_2)$$
$$= m_1^{\,e} \times m_2^{\,e}$$
$$= (m_1 \times m_2)^e$$
$$= E(m_1 \times m_2)$$

$$E(m_1 \times m_2) = E(m_1) \times E(m_2)$$

همریختی روی جمع

- به همین ترتیب می توان هم ریختی روی جمع را نیز تعریف کرد.

$$E(m_1 + m_2) = E(m_1) + E(m_2)$$

- اما چرا هم ریختی روی ضرب و جمع اهمیت دارد؟؟

ویژگی هم ریختی، راهکاری برای محاسبه روی داده رمزشده

- هر تابع دودویی دلخواه را می توان با ترکیب AND و XOR ساخت.

ضرب

AND

0 AND 0	0
1 AND 0	0
0 AND 1	0
1 AND 1	1

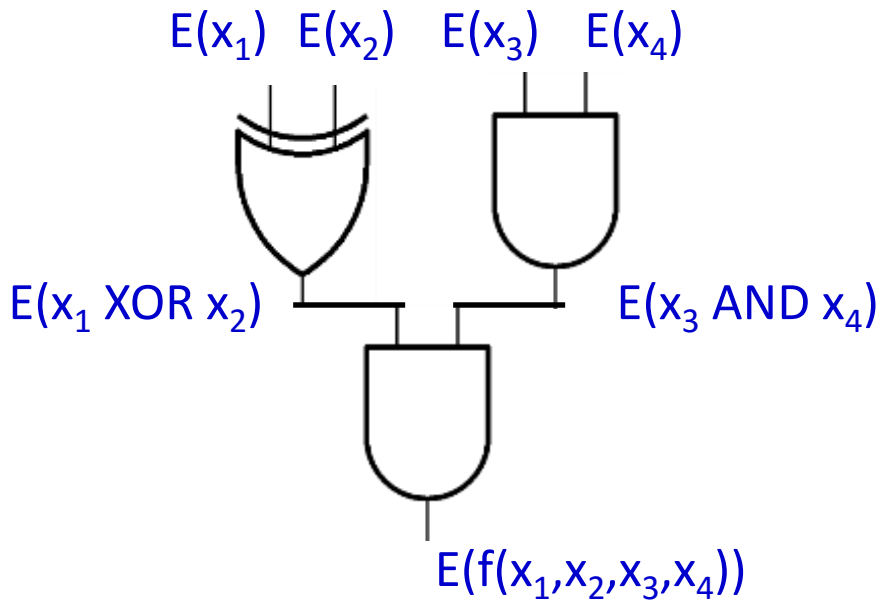
جمع

XOR

0 XOR 0	0
1 XOR 0	1
0 XOR 1	1
1 XOR 1	0

ویژگی هم ریختی، راهکاری برای محاسبه روی داده رمزشده

- اگر بتوان ضرب و جمع روی داده رمزشده انجام داد، آنگاه هر تابع دودویی دلخواه قابل اجرا روی داده رمزشده خواهد بود[RAD’78].



$$E\left(f(x_1, \dots, x_n)\right) = f\left(E(x_1), \dots, E(x_n)\right)$$

اهمیت رمزگذاری تمام همریخت

محاسبات دلخواه روی داده رمز شده بدون دسترسی به داده اصلی

کاربرد وسیع برای امنیت داده در رایانش ابری

طرح رمزگذاری تمام همریخت

تلاش برای ارائه طرح رمزگذاری تمام هم ریخت

سال ها تلاش برای ارائه طرحی با چنین ویژگی انجام شد...

اما همگی به شکست انجامید.

تا اینکه....

تلاش برای ارائه طرح رمزگذاری تمام هم ریخت

در اکتبر ۲۰۰۸، ***Craig Gentry*** با بهره گیری از ابزار مشبکه مسئله را حل کرد.

مشبکه چیست؟؟

- آرایه ای منظم از نقاط در فضای چند بعدی

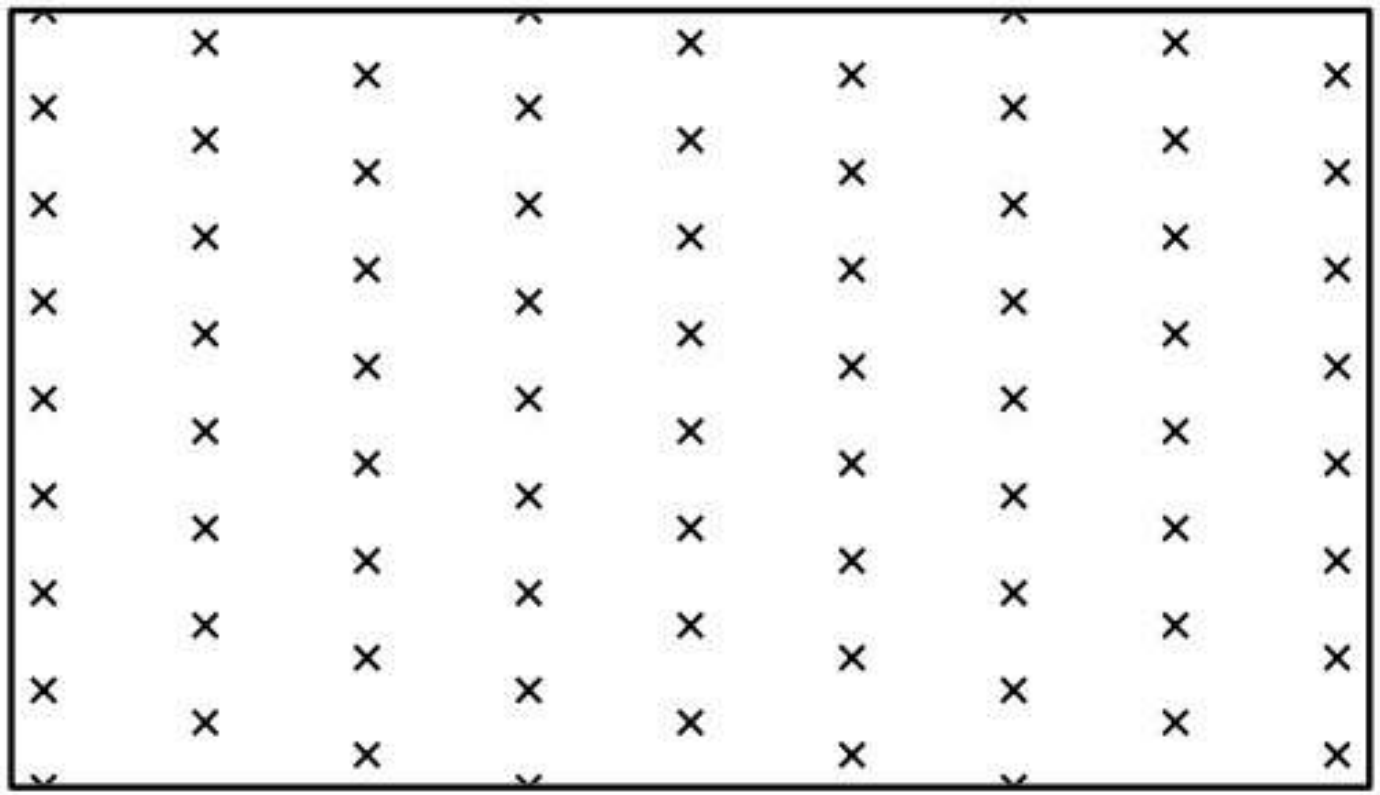


Image courtesy of Oded Regev (link)

تعریف مشبکه

❖ ترکیب خطی با ضرایب صحیح n بردار مستقل خطی $b_1, b_2, \dots b_n \in \mathbb{R}^m$

$$\mathcal{L}(b_1, b_2, \dots, b_n) = \left\{ \sum x_i b_i | x_i \in \mathbb{Z} \right\}$$

❖ بردارهای $b_1, b_2, \dots b_n \in \mathbb{R}^m$ پایه مشبکه نامیده می شوند.

پایه خوب و پایه بد!!

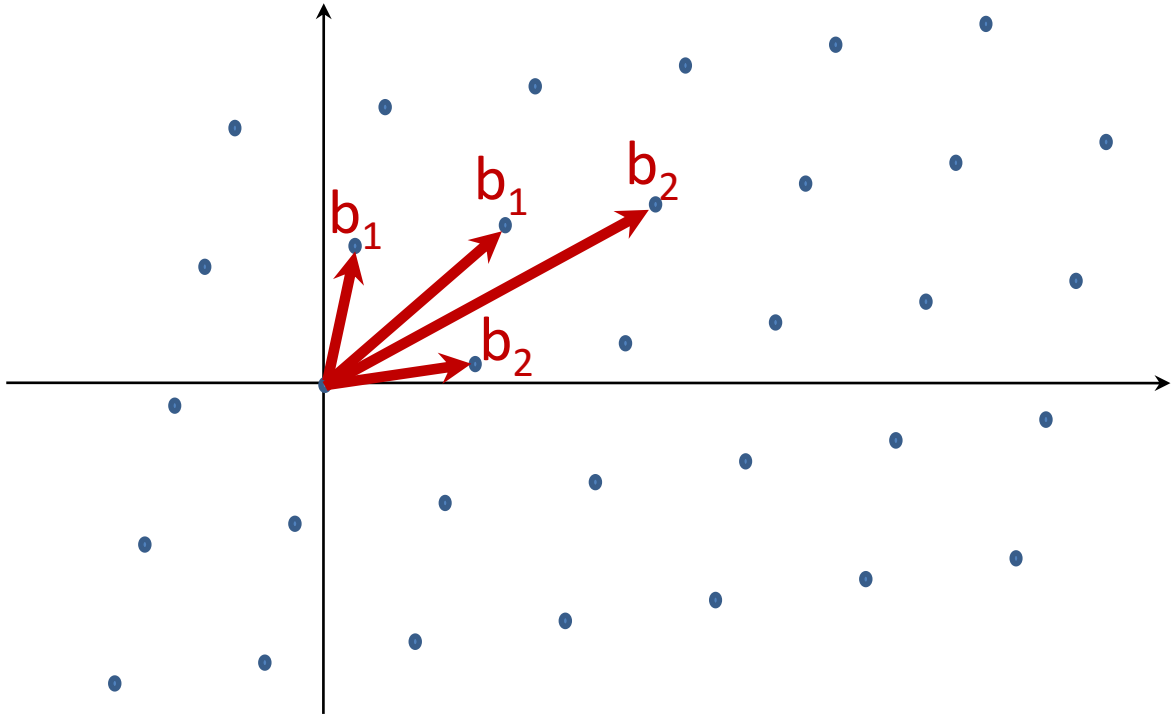
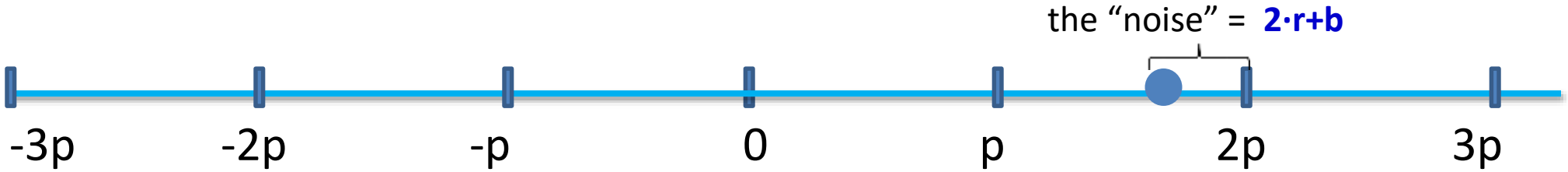


Image courtesy of Craig Gentry (link)

ایده اولیه طرح جنتری

- کلید محرمانه: عدد اول بزرگ p
- رمزگذاری بیت b
  - مضربی صحیح از p انتخاب می شود مثلا q.p
  - یک عدد کوچک r به تصادف انتخاب می شود.
  - رمز شده بیت b به صورت c=q.p+2.r+b تولید می شود.
- رمزگشایی c
  - b=(c mod p) mod 2

شرط رمزگشایی صحیح: نویز <p/2



بررسی تمام همریخت بودن طرح اولیه جنتری

$$c_1+c_2 = p\cdot(q_1 + q_2) + \underbrace{2\cdot(r_1+r_2) + (b_1+b_2)}$$

*noise= 2 * (initial noise)*

$$c_1c_2 = p\cdot(c_2\cdot q_1+c_1\cdot q_2-q_1\cdot q_2) + \underbrace{2\cdot(r_1r_2+r_1b_2+r_2b_1) + b_1b_2}$$

noise = (initial noise)²

نویز افزایش پیدا کرد

ایدهٔ bootstrapping برای رفع مشکل افزایش نویز

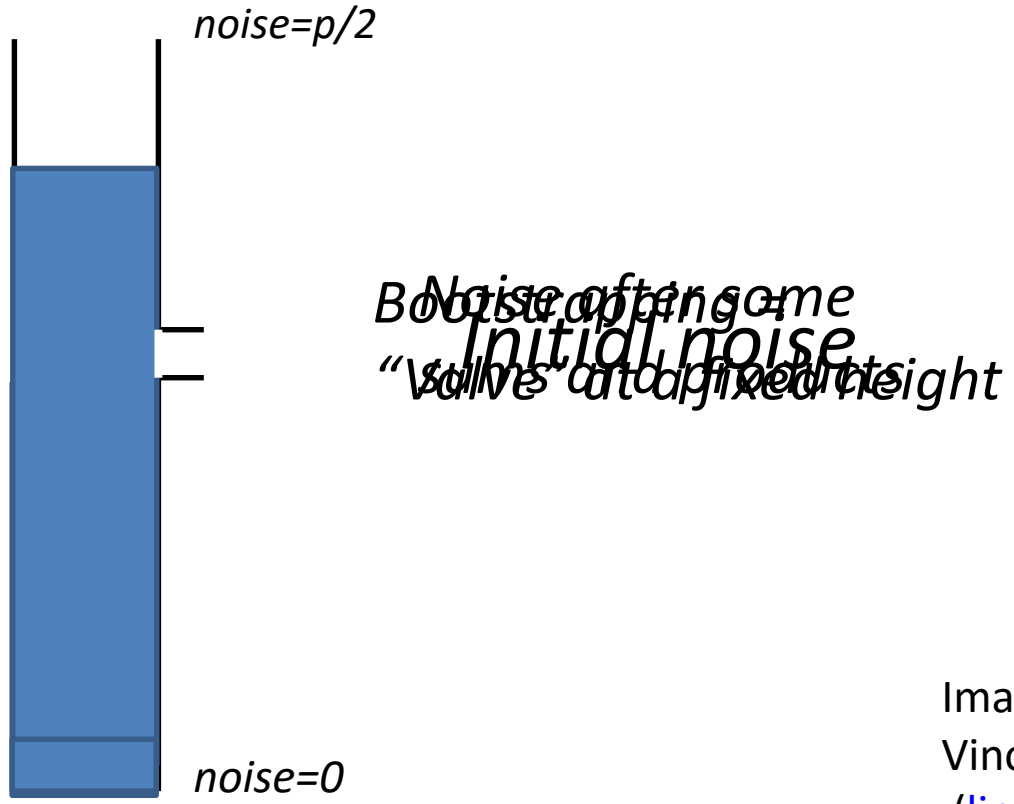
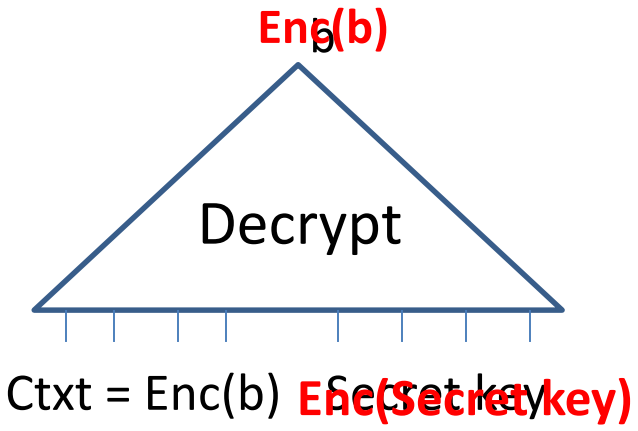


Image courtesy of
Vinod Vaikuntanathan
(link)

ایدهٔ bootstrapping برای رفع مشکل افزایش نویز

- بهره گیری از تابع رمزگشایی به عنوان راهکاری برای کاهش نویز



طرح جنتری

- طرح جنتری [Gentry'09]:
  - پیام: $\vec{m} \in \{0,1\}^n$
  - کلید عمومی: ماتریس پایه بد برای مشبکه ایده آل J: B_{pk}^J
  - کلید خصوصی: ماتریس پایه خوب برای مشبکه ایده آل J: B_{sk}^J
  - رمزگذاری: $\vec{c} = (2\vec{r} + \vec{m}) mod\ B_{pk}^J = 2\vec{r} + \vec{m} + \vec{J}$
  - رمزگشایی: $\vec{m} = (c\ mod\ B_{sk}^J) mod\ 2$

نسل دوم طرح های رمزگذاری تمام هم ریخت

- مبتنی بر مساله استاندارد Search-LWE
  - بردار محرمانه $\vec{s} \in \mathbb{Z}_q^n$، بردار تصادفی $\vec{a} \in \mathbb{Z}_q^n$ و مقدار تصادفی $e \leftarrow \chi$ داده شده است.

    قرار می دهیم $b = \vec{s}^t.\vec{a} + e$. هدف محاسبه $\vec{s}$ با استفاده از b است.
- نسل دوم رمزگذاری تمام همریخت [BV'11]:
  - کلید محرمانه: $\vec{S} = (-\vec{s},1)$
  - رمزگذاری بیت b: متن رمزشده برابر است با $\vec{C} = (\vec{c},c)$ به گونه ای که

    $$c = \vec{s}^t.\vec{c} + 2r + b = \vec{s}^t.\vec{c} + \mathrm{e}$$
  - رمزگشایی: $b = (\vec{S}^t.\vec{C})\ mod\ 2$

نسل دوم طرح های رمزگذاری تمام هم ریخت

❖ هم ریختی در جمع

$$\left(\vec{S}^t.\left(\overrightarrow{C_1}+\overrightarrow{C_2}\right)\right)=\left(\vec{S}^t.\overrightarrow{C_1}\right)+\left(\vec{S}^t.\overrightarrow{C_2}\right)$$

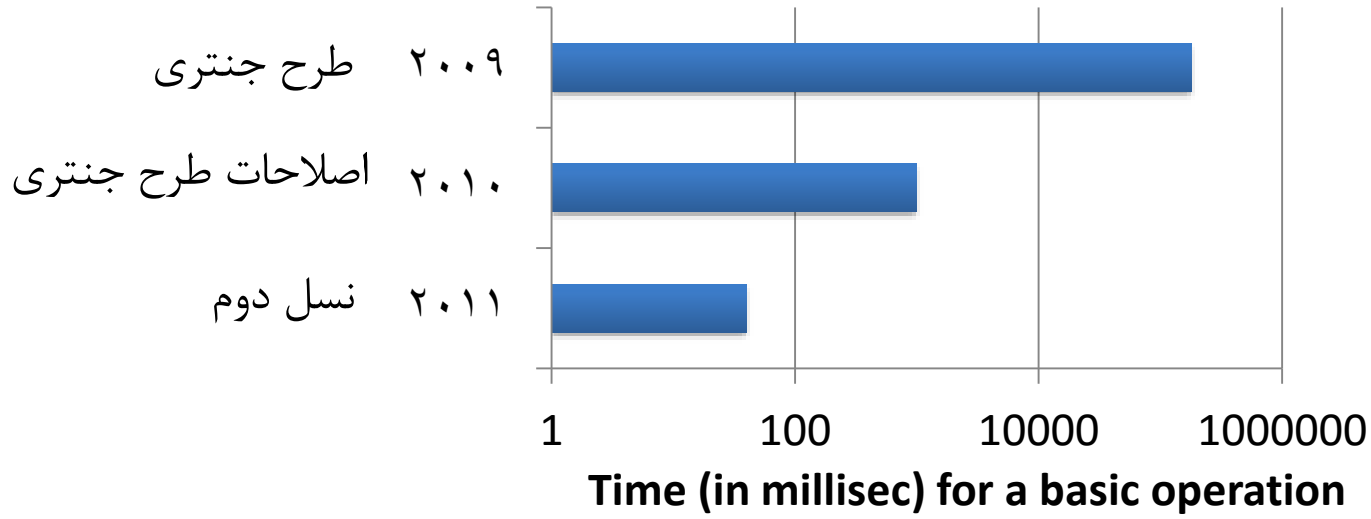
❖ هم ریختی در ضرب

$$\left(\overrightarrow{(S_1\otimes S_2)}^t.\overrightarrow{(C_1\otimes C_2)}\right)=\left(\vec{S}^t.\overrightarrow{C_1}\right)\times\left(\vec{S}^t.\overrightarrow{C_2}\right)$$

⊗ نماد ضرب تانسوری است.

نسل دوم طرح های رمزگذاری تمام هم ریخت

- علاوه بر امنیت، نسل دوم بهبود چشمگیری در افزایش سرعت طرح ها داشته است.



پیاده سازی طرح رمزگذاری تمام هم ریخت

- مشکلات زیادی برای پیاده سازی طرح رمزگذاری تمام همریخت وجود دارد از جمله:

۱. سرعت کم رمزگذاری: در حال حاضر سرعت رمزگذاری در بهترین حالت حدود ۱۹ ثانیه برای هر بیت است!!

۲. طول زیاد کلید: جدول زیر طول کلید لازم برای پیاده سازی طرح جنتری را نشان می دهد. طول کلید در نسل دوم نیز تقریبا با همین مقادیر برابر است.

طول کلید عمومی	ابعاد
۷۰ مگا بایت	۲۰۴۸
۲۸۵ مگا بایت	۸۱۹۲
۲,۳ گیگا بایت	۳۲۷۶۸

پروژه PROCEED (PROgramming Computation on EncyptEd Data)

- با هدف توانایی اجرای محاسبات پیچیده روی داده رمزشده، پروژه PROCEED از سال ۲۰۱۱ در سازمان پروژه های تحقیقاتی پیشرفته دفاعی ایالات متحده (DARPA) آغاز شده است.
- PROCEED مبتنی بر دو فناوری است:
  - رمزگذاری تمام هم ریخت
  - محاسبات امن چند عاملی

پروژه PROCEED (PROgramming Computation on EncyptEd Data)

- در طرح اصلی جنتری، اعمال تابع پیچیده دودویی f روی داده رمزشده نسبت به داده اصلی حدود 10^{14} برابر زمان می برد.
- در PROCEED هدف آن است که این زمان حداقل به 10^{7} کاهش پیدا کند.
- بودجه اولیه اختصاص یافته به پیاده سازی کارای رمزگذاری تمام هم ریخت در این پروژه ۲۰ میلیون دلار است.

زمینه های پژوهشی

- ارائه یک طرح رمزگذاری تمام همریخت کارا
  - سرعت رمزگذاری قابل مقایسه با طرح های رمزگذاری کنونی
  - طول کلید کوچکتر
  - حداقل نیاز به bootstrapping
- نسل سوم رمزگذاری تمام همریخت
  - مبتنی بر ابزار Gadget trapdoor

مراجع

- [RAD’78] Rivest, Ronald L., Len Adleman, and Michael L. Dertouzos. "On data banks and privacy homomorphisms." *Foundations of secure computation* 4.11 (1978): 169-180.
- [Gentry’09] Gentry, Craig. *A fully homomorphic encryption scheme*. Diss. Stanford University, 2009.
- [BV’11] Brakerski, Zvika, and Vinod Vaikuntanathan. "Fully homomorphic encryption from ring-LWE and security for key dependent messages." *Annual Cryptology Conference*. Springer Berlin Heidelberg, 2011.
- Libicki, Martin C., et al. *Ramifications of DARPA's Programming Computation on Encrypted Data Program*. RAND NATIONAL DEFENSE RESEARCH INST SANTA MONICA CA, 2014.