Residue arithmetic systems in cryptography: a survey on modern security applications

سیستم های عدد مانده ای در رمزنگاری: مروری بر کاربردهای امنیتی مدرن

تهیه کننده سینا سروش

استاد محترم خانم دکتر جاسبی

خرداد ۱۴۰۰

مقدمه:

- در این مطالعه مروری، به بعضی کاربردهای مدرن و غیرمتداول RNS در حوزه رمزنگاری پساکوانتومی ، زیرساخت های ابری و رمزنگاری مبتنی بر لاتیس پرداخته می شود.

مقدمه:

- پیشرفت روز افزون و مطرح شدن مباحث جدید مثل IoT، SDN، Cloud و ... نیاز به راه کارهایی برای افزایش امنیت سرویس های جدید میباشد. استفاده از RNS در بحث رمزنگاری خلا های امنیتی این سرویس های جدید را پوشش میدهد.

روش های پیشنهادی ارایه شده:

- رمزنگاری مبتنی بر لاتیس
- استفاده از RNS در امنیت Cloud

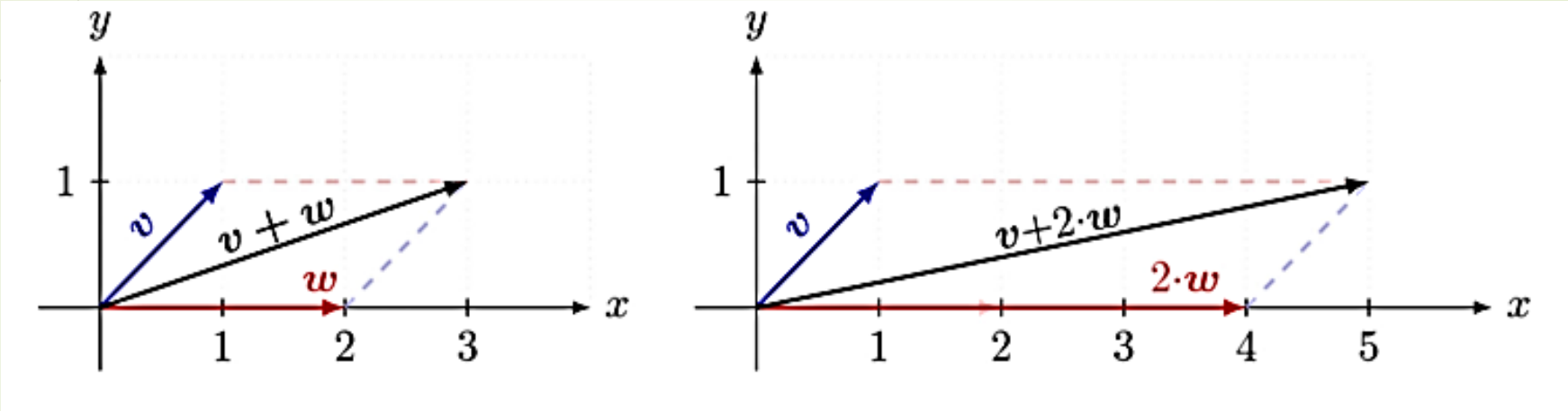
RNS در رمزنگاری پساکوانتومی:

- درحالی که آینده محاسبات کوانتومی هنوز مبهم است و هنوز مشخص نیست ما می توانیم مسئله برتری کوانتومی را حل کنیم یا خیر، تلاش برای ارائه طرح های رمزنگاری پساکوانتومی واقع گرایانه نیست. دلیلش کاملا واضح است چرا که یک کامپیوتر کوانتومی کامل دنیای رمزنگاری کلید عمومی را متحول می کند.

لاتیس (LBC):

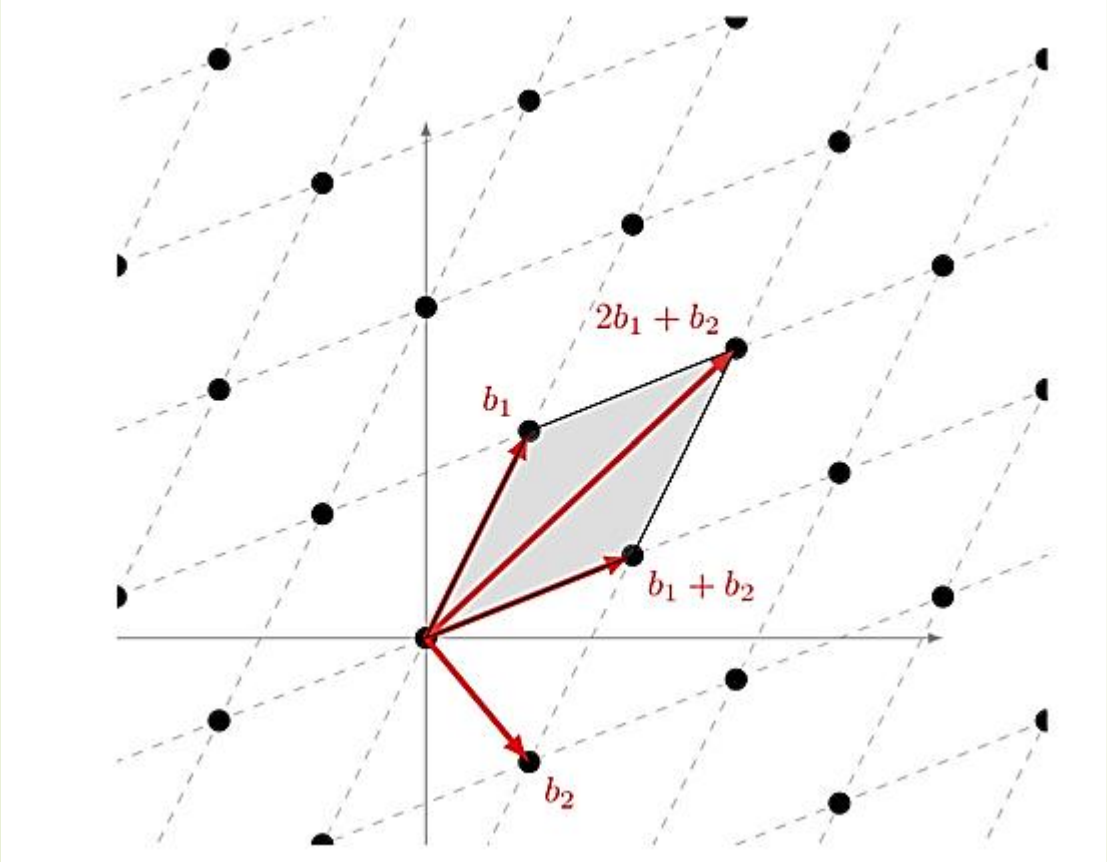
- سیستم های رمزنگاری مبتنی بر لاتیس (LBC) گزینه های بسیار امیدوارکننده ای برای دوره رمزنگاری پساکوانتومی هستند چرا که امنیت قوی آن ها در سختی بدترین حالت اثبات شده است و بعلاوه پیاده سازی های نسبتا کارامدی داشته و بسیار ساده هستند.

- قلب یک لاتیس یک بردار است؛ برای سادگی ما فضای دوبعدی را تصور می کنیم (جبر خطی در فضاهایی با ابعاد بالاتر نیز میسر است)



بطور کلی، اگر ضرب بردارهای v و w به ترتیب در اعداد c و d را ترکیب کنیم و جمع آن ها را در نظر بگیریم، ترکیب خطی cv + dw را بدست می آوریم. بعنوان مثال، عملیات روی سمت راست شکل بالا مربوط به ترکیب خطی v و w برای c=1 و d=2 است.

- حال اگر تمام ترکیب های ممکن برای اعداد c و d را در نظر بگیریم می توانیم دیاگرامی مانند شکل زیر رسم کنیم.



- با دو بردار بعنوان پایه (b_1,b_2) شروع می کنیم و تمام مقادیر ممکن برای c و d را در نظر می گیریم تا نمودار نقطه ای بدست آید که در آن هر نقطه مربوط به نقطه پایانی نتیجه $cb_1 + db_2$ است. این مجموعه نقاط را یک لاتیس می نامند.

$$\mathcal{L}(\mathbf{b_1}, \mathbf{b_2}, \ldots, \mathbf{b_n}) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b_i} : x_i \in \mathbb{Z} \right\}. \qquad (7)$$

رمزنگاری مبتنی بر لاتیس:

- روش استفاده از لاتیس ها در رمزنگاری اندکی مبهم است و تا انتشار مقاله آجتای [۱] اطلاعات زیادی درباره آن وجود نداشت. این مطالعه یک چارچوب مستحکم برای ساختن سیستم های رمزنگاری مبتنی بر لاتیس فراهم ساخت.
- وقتی بحث امنیت است، LBC را می توان به دو دسته تقسیم کرد: دسته اول شامل روش های کاربردی است که بسیار کارامد هستند اما اغلب فاقد اثبات قوی می باشند. نوع دوم امنیت قابل اثباتی بر اساس سختی بدترین حالت مسائل لاتیس دارند اما تنها تعداد انگشت شماری از آن ها در عمل قابل استفاده هستند

NTRU : یک سیستم رمزنگاری مبتنی بر لاتیس

- حال از مفاهیم اصلی بخش های قبلی برای توصیف سیستم رمزنگاری NTRU استفاده می کنیم. NTRU توسط هافمن، پیفر و سیلورمن پیشنهاد شده است[۳۱]. NTRU اساسا با در نظر گرفتن حسابان روی چندجمله ای ها در $Z_q[x]/(x^n - 1)$ برای n عدد اول و q عدد صحیح کوچک پیشنهاد شده است.

- یک ماتریس تصادفی $H \in Z^{n \times m}$ فرض می شود که می توان آن را به صورت یک ماتریس بلاک نوشت:

$$\mathbf{H} = [\mathbf{H}^{(1)}|\mathbf{H}^{(2)}|\dots|\mathbf{H}^{(m/n)}], \tag{8}$$

- که در آن هر بلاک $H^{(i)} \in Z^{n \times m}$ یک ماتریس دوری است:

$$\mathbf{H}^{(i)} = \begin{bmatrix} h_1^i & h_n^i & \cdots & h_3^i & h_2^i \\ h_2^i & h_1^i & \cdots & h_4^i & h_3^i \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{n-1}^i & h_{n-2}^i & \cdots & h_1^i & h_n^i \\ h_n^i & h_{n-1}^i & \cdots & h_2^i & h_1^i \end{bmatrix} \tag{9}$$

- یعنی تمام ستون ها از ستون اول $h^{(i)} = (h_1^i, h_2^i, \dots, h_3^i)$ با چرخش های دایره ای بدست آمده اند. در چرخش ماتریسی، می توان نوشت:

$$\mathbf{H} = \mathbf{P}^* \mathbf{H}^{(i)} = [\mathbf{h}^{(i)}, \mathbf{P}\mathbf{h}^{(i)}, \dots, \mathbf{P}^{n-1}\mathbf{h}^{(i)}], \qquad (10)$$

- که P یک ماتریس جایگشت است:

$$\mathbf{P} = \left[\begin{array}{c|c} \mathbf{0}^T & 1 \\ \hline \ddots & \\ \mathbf{I} & 0 \\ & \ddots \end{array}\right] \qquad (11)$$

- که مختصات $h^{(i)}$ را بطور دایره ای می چرخاند. از نامگذاری $P * H^{(i)}$ استفاده شده که نشان می دهد جایگشت P روی ستون های $H^{(i)}$ اعمال می شود.

سیستم NTRU بصورت زیر ایجاد می شود[۳۲]:

- -**پارامترها** : یک بعد اول n ، یک ماژول صحیح q ، یک عدد صحیح کوچک p و یک کران وزنی کوچک b_f .
- -**کلید خصوصی**: کلید خصوصی متشکل از دو چندجمله ای است که بصورت بردارهای دودویی (f,g) نشان داده می شوند بطوریکه $[P * F]$ ماژول معکوس پذیر q است،

$$[P * F] = I(\text{mod } p) \quad \text{و} \quad [P * g] = O(\text{mod } p).$$

- **کلید عمومی**: بدون وارد شدن به جزئیات ریاضی، کلید عمومی یک ماتریس ساخت یافته بصورت زیر است:

$$\mathbf{E} = \begin{bmatrix} \mathbf{I} & \mathbf{O} \\ \mathbf{P^*e} \; q \cdot \mathbf{I}, \end{bmatrix} \qquad (12)$$

که $e = [P * F]^{-1} g(\mathrm{mod}\ q)$. ساختار ماتریس ساده است چون کلید عمومی می تواند تنها با بردار e نمایش داده شود

رمزنگاری در NTRU:

- پیام ورودی بصورت یک بردار $m \in \{-1,0,1\}^n$ (یعنی سه مقدار برای ضرایب چندجمله ای در یک بردار n بیتی) با دقیقا $b_f + 1$ ورودی مثبت و b_f ورودی منفی رمزنگاری می شود. بردار m با یک بردار تصادفی $r \in \{-1,0,1\}^n$ به دقیقا $b_f + 1$ ورودی مثبت و b_f ورودی منفی الحاق می شود تا یک بردار کوتاه (r,m) بدست آید. سپس متن رمز بصورت زیر محاسبه می شود:

$$\mathbf{c} = \mathbf{m} + [\mathbf{P}^*\mathbf{e}]\mathbf{r} \pmod{q}. \qquad (13)$$

که در آن m پیام، e کلید عمومی و r یک مقدار تصادفی است.

رمزگشایی در NTRU:

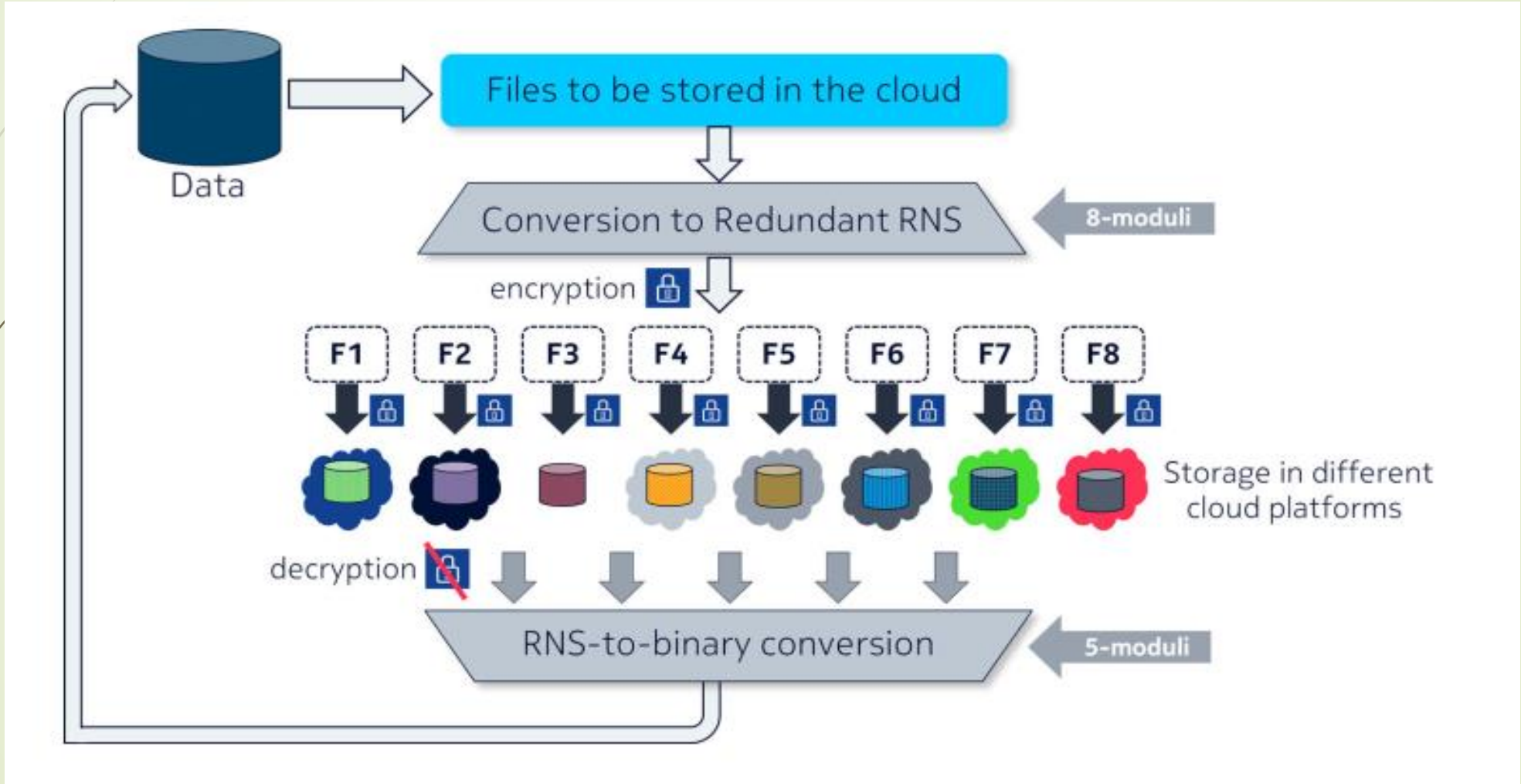
- رمزگشایی با ضرب متن رمز در ماتریس رمز $[P * F](\mathrm{mod}\ q)$ بصورت زیر انجام می شود:

$$
\begin{aligned}
[\mathbf{P}^*\mathbf{f}]\mathbf{c}\ (\mathrm{mod} q) &= \\
&= [\mathbf{P}^*\mathbf{f}]\mathbf{m} + [\mathbf{P}^*\mathbf{f}][\mathbf{P}^*\mathbf{e}]\mathbf{r}\ (\mathrm{mod} q) = \\
&= [\mathbf{P}^*\mathbf{f}]\mathbf{m} + [\mathbf{P}^*\mathbf{g}]\mathbf{r}\ (\mathrm{mod} q).
\end{aligned}
\tag{14}
$$

کاربرد RNS در ذخیره سازی ابر و IoT:

- RNS می تواند گزینه های منحصربفردی در یکی از مهم ترین مسائل در سیستم های ابر یعنی مسئله اعتماد فراهم کند. خواهیم دید که چگونه PRNS می تواند بطور موثر بعضی از نگرانی های مهم در معماری های ابری مانند یکپارچگی داده ها، دسترس پذیری و اعتمادپذیری را حل کند[۱۵].

ذخیره فایل توزیع شده در ابر با RNS:



- در معماری شکل بالا [۱۵] استفاده از PRNS با هشت ماژول فرض می شود که سه تای آن ها بعنوان ماژول تکراری بکار می روند. کاربر ابتدا داده هایی که قرار است در ابر ذخیره شوند را در قالب RNS ذخیره می کند.
- هر قسمت (Chunk) از طریق رمز نگاری متقارن رمزنگاری شده و در یک ابر متفاوت ذخیره می شود.
- کاربر یک فایل XML متا داده را همراه با اطلاعاتی درباره موقعیت فایل ها و روش بازیابی نگهداری می کند.

مزایای این روش (این معماری):

- هنگام خرابی سیستم کاربر می تواند هنوز داده ها را بازسازی. کند با این فرض که ۵ ماژول کافی است.
- مراکز سرویس دهی ابر از وجود فایل ها مطلع نیستند چرا که فقط کاربر می داند فایل ها کجا هستند و چگونه بازیابی می شوند.
- دانلود موازی فایل ها از سرویس دهندگان مختلف باعث افزایش بهره وری پهنای باند می شود.

جمع بندی:

- در این مطالعه، گزینه های بهینه سازی مختلف ارائه شده توسط محاسبات مانده ای با توجه به کاربردهای امنیتی بررسی شده اند. این کاربردها به پیشرفته ترین سیستم های رمزنگاری مبتنی بر لاتیس بسط داده شده اند که گزینه هایی نویدبخش برای محاسبات پساکوانتومی هستند. شباهت هایی میان این سیستم ها وجود دارد که منجر به مشاهده مهم می شود اینکه هر گونه بهبود در محاسبات RNS به معنای بهبود در مجموعه بزرگ تری از الگوریتم های کلید عمومی است.