به نام خدا

دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری ۲۹ بهمن ۱۳۹۱

جلسه‌ی ۲۰: رمز نگاری با کلید عمومی و مقدمه‌ای بر نظریه اعداد

مدرس: دکتر شهرام خزائی *نگارنده: آرین کمال*

هدف ما در این جلسه ارائه تعریفی برای مفهوم رمز نامتقارن[۱] و بررسی امنیت آن می‌باشد. در ادامه هم به بیان مفاهیم اولیه از نظریه اعداد می‌پردازیم.

۱ سیستم رمز نامتقارن

تا اینجا برای رمزگذاری و رمزگشایی از یک کلید مشترک استفاده کردیم. به همین جهت به آن سیستم رمزنگاری متقارن گویند که تنها به یک کانال امن برای انتقال کلید نیاز دارد. در ادامه سیستم رمزنگاری نامتقارن یا سیستم رمزنگاری با کلید عمومی[۲] را معرفی می‌کنیم. در این سیستم دو کلید متفاوت به نام کلید عمومی[۳] و کلید خصوصی[۴] (یا کلید مخفی[۵]) وجود دارد، که کلید عمومی در دسترس همگان است ولی کلید خصوصی را تنها گیرنده پیام دارد.

تعریف ۱ *سیستم رمز نامتقارن یک سه‌تایی مرتب به صورت* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *روی فضای متن* $\mathcal{M}$ *از الگوریتم‌های چند جمله‌ای تصادفی* (PPT)[۶] *است و الگوریتم‌های آن بدین صورت تعریف می‌شوند:*

- Gen *الگوریتم تولید کلید است که با ورودی* 1^n *زوج مرتب* (pk, sk) *را که به ترتیب کلید عمومی و کلید خصوصی هستند، تولید می‌کند.*
$$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$$
- Enc *الگوریتم رمزگذاری است که با داشتن کلید عمومی* pk *و متن* $m \in \mathcal{M}$ *به عنوان ورودی، متن رمز شده* c *را تولید می‌کند.*
$$c \leftarrow \mathsf{Enc}_{pk}(m)$$
- Dec *الگوریتم رمزگشایی است که قطعی است و با داشتن کلید خصوصی* sk *و متن رمز شده* c*، متن* $m \in \mathcal{M} \cup \{\perp\}$ *را تولید می‌کند.*
$$m \leftarrow \mathsf{Dec}_{sk}(c)$$

[۱] asymmetric cryptography
[۲] public-key cryptography
[۳] public-key
[۴] private-key
[۵] secret-key
[۶] probabilistic polynomial time

- *به ازای همه‌ی* $n \in \mathbb{N}$ *و* $m \in \mathcal{M}$

$$\Pr[(sk, pk) \leftarrow \mathsf{Gen}(1^n) : \mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = m] = 1$$

نکته ۱ *خروجی تابع* Dec *در فضای* $\mathcal{M} \cup \{\perp\}$ *است که* $\perp$ *تنها وقتی خروجی خواهد بود که متن ورودی* c *نامعتبر باشد.*

۱.۱ امنیت رمز نامتقارن

بیاد بیاورید که شانون امنیت کامل برای سیستم رمز متقارن را با استفاده از تمایزناپذیری توزیع‌های زیر برای هر زوج پیام دلخواه m_0 و m_1 در فضای پیام تعریف کرد:

$$\{k \leftarrow \mathsf{Gen}(1^n) : \mathsf{Enc}_k(m_0)\}$$
$$\{k \leftarrow \mathsf{Gen}(1^n) : \mathsf{Enc}_k(m_1)\}$$

ابتدا توجه کنید که کلید عمومی سیستم رمز نامتقارن در اختیار همه، از جمله مهاجم، قرار می‌گیرد. بنابراین اگر علاقه‌مند به تعریف امنیت کامل برای سیستم رمز نامتقارن باشیم، به طور طبیعی باید سیستمی را امن کامل در نظر بگیریم که برای هر زوج پیام دلخواه m_0 و m_1 در فضای پیام آن، توزیع‌های زیر تمایزناپذیر باشند:

$$\{(pk, sk) \leftarrow \mathsf{Gen}(1^n) : \langle pk, \mathsf{Enc}_{pk}(m_0)\rangle\} \quad (1)$$

$$\{(pk, sk) \leftarrow \mathsf{Gen}(1^n) : \langle pk, \mathsf{Enc}_{pk}(m_1)\rangle\} \quad (2)$$

قضیه ۱ (امکان‌ناپذیری امنیت کامل) *سیستم رمز نامتقارن با امنیت کامل وجود ندارد.*

برهان. یک زوج کلید عمومی و متن رمز شده را در نظر بگیرید. با توجه به شرط صحت رمزگشایی، امکان رمزگشایی متن رمز شده به دو متن متفاوت وجود ندارد. بنابراین وقتی یک متن دلخواه با استفاده از یک مقادیر تصادفی تحت کلید عمومی pk به یک متن رمز شده c تبدیل شده باشد، هیچ متن اصلی دیگری تحت هیچ مقدار تصادفی با استفاده از همان کلید pk به متن رمزشده c تبدیل نخواهد شد. لذا مهاجم با منابع نامحدود[۷] می‌تواند به راحتی پیام m_0 و m_1 را با همه مقادیر تصادفی که الگوریتم Enc استفاده می‌کند تحت کلید عمومی دریافتی رمز کرده و با مقایسه آنها با متن رمزی دریافتی، متن اصلی صحیح را تشخیص دهد. ■

بنابراین برای سیستم رمز نامتقارن باید به امنیت محاسباتی بسنده کرد. همانند سیستم رمز متقارن می‌توان امنیت تک‌پیامی محاسباتی را بر مبنای تمایزناپذیری محاسباتی توزیع‌های (۱) و (۲) تعریف نمود. بجای این کار، امنیت را با استفاده از آزمایش که به دنبال می‌آید تعریف می‌کنیم که امنیت تک‌پیامی محاسباتی را نتیجه می‌دهد.

[۷] unbounded adversary

آزمایش.$[\mathsf{PubK}^{\text{eav}}_{\mathcal{A},\Pi}]$ آزمایش امنیت تک‌پیامی برای سیستم رمز نامتقارن Π در برابر مهاجم $\mathcal{A}$ به صورت زیر است:

۱. چالشگر با اجرای الگوریتم تولید کلید، کلید عمومی pk و کلید خصوصی sk، را تولید می‌کند.

$$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$$

۲. مهاجم $\mathcal{A}$ با دریافت کلید pk دو پیام m_0 و m_1 از فضای $\mathcal{M}$، که $|m_0| = |m_1|$، را تولید می‌کند و به چالشگر بر می‌گرداند.

$$(m_0, m_1) \leftarrow \mathcal{A}(pk)$$

۳. چالشگر یک بیت تصادفی انتخاب می‌کند.

$$b \leftarrow \{0, 1\}$$

۴. چالشگر متن رمزی c، که رمز شده متن اصلی m_b تحت کلید pk است را محاسبه می‌کند و برای چالشگر می‌فرستد.

$$c \leftarrow \mathsf{Enc}_{pk}(m_b)$$

۵. مهاجم با گرفتن متن رمز شده c، بیت $\hat{b}$ را تولید می‌کند.

$$\hat{b} \leftarrow \mathcal{A}(c)$$

خروجی آزمایش که با $\mathsf{PubK}^{\text{eav}}_{\mathcal{A},\Pi}(n)$ نشان داده می‌شود برابر ۱ است اگر $b = \hat{b}$ و صفر است اگر $\hat{b} \neq b$.

تعریف ۲ (امنیت تک پیامی در سیستم رمز نامتقارن) *سیستم رمز نامتقارن* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *دارای امنیت تک‌پیامی است اگر برای هر مهاجم چندجمله‌ای احتمالاتی مانند* $\mathcal{A}$*، تابع ناچیز* $\epsilon(n)$ *وجود داشته باشد که*

$$\Pr\{\mathsf{PubK}^{\text{eav}}_{\mathcal{A},\Pi}(n) = 1\} \leq \frac{1}{2} + \epsilon(n)$$

تفاوت تعریف بالا، با تعریف‌های مشابه در سیستم رمزهای متقارن، در این است که کلید عمومی به مهاجم داده می‌شود که خودبه‌خود دسترسی اوراکلی به الگوریتم رمزنگاری را برای او فراهم می‌سازد. بنابراین، امنیت تک‌پیامی، امنیت متن اصلی انتخابی را نیز نتیجه می‌دهد. همانند سیستم رمزنگاری متقارن، می‌توان تعاریف را به امنیت چندپیامی گسترش داد. می‌توان نشان داد داریم در سیستم رمز نامتقارن امنیت چندپیامی با امنیت تک‌پیامی معادل است.

قضیه ۲ (معادل‌بودن امنیت‌ها) *سیستم رمز نامتقارن امن تک‌پیامی، دارای امنیت چندپیامی و متن اصلی انتخابی است.*

در سیستم‌های رمز نامتقارن نمی‌توان از الگوریتم‌های رمزنگاری قطعی استفاده کرد، زیرا اگر الگوریتم رمزنگاری قطعی باشد آنگاه چون مهاجم کلید عمومی را در اختیار دارد، می‌تواند رمز شده‌ی m_0 و m_1 را بدست آورد و با متن رمز شده مقایسه کند و به احتمال ۱ جواب صحیح را برگرداند.

قضیه ۳ (لزوم رمزنگاری‌تصادفی) *سیستم رمز نامتقارن با الگوریتم رمزنگاری تصادفی، امنیت تک‌پیامی ندارد.*

۲ مقدمه‌ای بر نظریه اعداد

تعریف ۳ (همنهشتی) *اگر N یک عدد صحیح مثبت باشد، اعداد صحیح a و b را همنشت به پیمانه N گوییم هرگاه*

$$N|a-b$$

همچنین همنهشتی را بدین گونه نشان می‌دهیم:

$$a \equiv b \pmod{N}$$

قضیه ۴ *رابطه همنهشتی به پیمانه عدد صحیح مثبت N، یک رابطه هم‌ارزی روی $\mathbb{Z}$ است.*

نکته ۲ *رابطه هم ارزی به پیمانه N، مجموعه اعداد صحیح $\mathbb{Z}$، را به کلاس‌های هم‌ارزی افراز می‌کند که مجموعه‌ی این کلاس‌ها با $\mathbb{Z}_N$ نشان داده می‌شود.*

نتیجه ۴ *اگر N یک عدد صحیح مثبت باشد، کلاس‌های هم‌ارزی $\mathbb{Z}$ به صورت زیر می‌باشد:*

$$\mathbb{Z}_N = \{[۰], [۱], ..., [N-۱]\}$$

که از این پس آن‌ها را به صورت زیر نشان می‌دهیم:

$$\mathbb{Z}_N = \{۰, ۱, ..., N-۱\}$$

مثال ۵ *روابط همنهشتی زیر به وضوح برقرارند*

۱. $۹ + ۷ = -۸ \pmod{۱۲}$

۲. $۵ \times ۷ = ۱۱ \pmod{۱۲}$

۳. $۵ - ۷ = ۱۰ \pmod{۱۲}$

نکته ۳ *اگر N یک عدد صحیح مثبت n- بیتی باشد و $x, y \in \mathbb{Z}_N$، آنگاه حاصل جمع و حاصل تفریق آن‌ها $x+y \pmod{N}$ و $x-y \pmod{N}$ در مرتبه زمانی $O(n)$ قابل محاسبه است. همچنین حاصل‌ضرب آن‌ها را $x \cdot y \pmod{N}$ در مرتبه زمانی $O(n^۲)$ می‌توان محاسبه کرد. البته با استفاده از الگوریتم کاراتسوبا[۸] می‌توان مرتبه زمانی محاسبه آن را به $O(n^{\log_۲ ۳})$ کاهش داد.*

۱.۲ بزرگترین مقسوم علیه مشترک

تعریف ۶ (بزرگترین مقسوم علیه مشترک) *اگر x, y دو عدد صحیح و مثبت باشند که حداقل یکی از آن‌ها ناصفر باشد، در این صورت عدد صحیح و مثبت d را بزرگترین مقسوم‌علیه مشترک (ب.م.م.) آنها می‌نامیم و با $\gcd(x, y)$ نمایش می‌دهیم، هرگاه شرایط زیر برقرار باشد:*

- $d|x$ و $d|y$

[۸] karatsuba algorithm

- *اگر $c|x$ و $c|y$ آنگاه $c|d$.*

مثال ۷ *به وضوح ۶ بزرگترین عددی است که بر ۱۲ و ۱۸ بخش‌پذیر است، پس:*

$$\gcd(۱۲, ۱۸) = ۶$$

برای محاسبه ب.م.م. می‌توان از الگوریتم ساده زیر استفاده کرد:

Algorithm 1 Algorithm: GREATEST COMMON DIVISOR

Input: two positive integers x, y
Output: $d = \gcd(x, y)$
function GCD(x, y)
  $(x, y) \leftarrow \big(\max(x, y), \min(x, y)\big)$
  if $y|x$ **then**
    return y
  else
    return GCD$(x - y, y)$

پیچیدگی این الگوریتم در بدترین حالت نمایی است، زیرا فرض کنید که $x = ۲^n - ۱$ باشد و $y = ۲$، آنگاه پیچیدگی الگوریتم از مرتبه $O(۲^n)$ است.
الگوریتم دیگری هم برای محاسبه ب.م.م وجود دارد که نسبت به الگوریتم قبلی کاراتر و از مرتبه زمانی $O(n^۲)$ است. این الگوریتم که به الگوریتم اقلیدس معروف است، تنها تغییر کوچکی در الگوریتم بالا می‌دهد.

Algorithm 2 Algorithm: EUCLID ALGORITHMS

Input: two positive integers $x \leq y$
Output: $d = \gcd(x, y)$
function GCD(x, y)
  if $x = 0$ **then**
    return y
  else
    return GCD$(x \mod y, y)$

صحت الگوریتم اقلیدس از لم زیر نتیجه می‌شود.

لم ۵ *اگر $x = yq + r$ باشد، آنگاه $\gcd(x, y) = \gcd(y, r)$*

قضیه ۶ *فرض کنید x, y دو عدد صحیح و مثبت باشند آنگاه اعداد صحیح $|a| \leq \frac{y}{۲}$ و $|b| \leq \frac{x}{۲}$ وجود دارند که $ax + by = \gcd(x, y)$. به علاوه $\gcd(x, y)$ کوچکترین عدد صحیح مثبتی است که به این صورت قابل بیان است.*

محاسبه ضرایب a, b با استفاده از تعمیمی از الگوریتم اقلیدس به صورت زیر و با همان پیچیدگی $O(n^۲)$ امکان‌پذیر است.

Algorithm 3 Algorithm: EXTENDED-EUCLID'S ALGORITHM

Input: two positive integers x, y with $x \geq y \geq 0$
Output: integers (a, b, d) such that $d = \gcd(x, y)$ and $ax + by = d$
function EXTENDED-EUCLID(x, y)
  if $y = 0$ **then**
    $(a, b, d) \leftarrow (1, 0, x)$
  else
    $(\hat{a}, \hat{b}, \hat{d}) =$ EXTENDED-EUCLID$(y, x \mod y)$
    $(a, b, d) \leftarrow (\hat{b}, \hat{a} - \lfloor \frac{a}{b} \rfloor \hat{b}, \hat{d})$
  return (a, b, d)

تعریف ۸ *می‌گوییم x, y نسبت به هم اول هستند، اگر و تنها اگر* $\gcd(x, y) = ۱$.

قضیه ۷ *دو عدد x, y نسبت به هم اول هستند، اگر و تنها اگر* $\exists a, b : ax + by = ۱$

۲.۲ گروه‌های همنهشتی

طبق تعریف گروه به راحتی می‌توان نشان داد که $(\mathbb{Z}_N, +)$ تشکیل یک گروه می‌دهد حال برای ساختن یک گروه ضربی روی $\mathbb{Z}_N$ باید تعریف دقیقی از معکوس داشته باشیم.

تعریف ۹ *فرض کنید $x \in \mathbb{Z}_N$ باشد، معکوس x (در صورت وجود) عضوی مانند $y \in \mathbb{Z}_N$ است به طوریکه:*

$$x \cdot y = ۱ \pmod N$$

y را به صورت $x^{-۱}$ نیز نشان می‌دهیم. معکوس x در صورت وجود یکتاست.

مثال ۱۰ *فرض کنید N یک عدد فرد باشد، در این صورت معکوس $۲ \in \mathbb{Z}_N$ برابر است با $\frac{N+۱}{۲}$. زیرا:*

$$۲ \cdot (\frac{N+۱}{۲}) = N + ۱ = ۱ \pmod N$$

لم ۸ *$x \in \mathbb{Z}_N$ معکوس دارد، اگر و تنها اگر* $\gcd(x, N) = ۱$.

برهان. ابتدا فرض کنید $\gcd(x, N) = ۱$ باشد، در این صورت بنا به قضیه ۴، وجود دارد a, b که $ax + bN = ۱$ بنابراین $ax = ۱ \pmod N$ پس $x^{-۱} = a \pmod N$. دقت کنید تمامی مراحل استدلال قسمت اول برگشت پذیر است، پس با توجه به آن حکم ثابت می‌شود. ■

نتیجه ۱۱ *محاسبه وارون ضربی با استفاده از الگوریتم اقلیدس در زمان $O(n^۲)$ امکان‌پذیر است.*

تعریف ۱۲ *Z_N^* را مجموعه تمام اعدادی از Z_N که نسبت به N اول باشد تعریف می‌کنیم، به عبارت دیگر:*

$$Z_N^* = \{x : \gcd(x, N) = ۱\}$$

همان طور که گفتیم $(\mathbb{Z}_N^*, +)$ تشکیل یک گروه را می‌دهد. حال سؤالی که مطرح می‌شود این است که آیا $(\mathbb{Z}_N^*, \times)$ تشکیل یک گروه می‌دهند؟ جواب مثبت است.

مثال ۱۳ *فرض کنید p عددی اول باشد، آنگاه داریم:*

$$\mathbb{Z}_p^* = \{x : \gcd(x, p) = ۱\} = \{۱, ۲, ..., p - ۱\}$$

مثال ۱۴

$$\mathbb{Z}_{۱۲}^* = \{۱, ۵, ۷, ۱۱\}$$

تعریف ۱۵ *تعداد اعضای مجموعه $\mathbb{Z}_N^*$ را با $\phi(N)$ نشان می‌دهیم:*

$$|\mathbb{Z}_N^*| = \phi(N)$$

مثال ۱۶ *فرض کنید p عددی اول است، آنگاه*

$$\phi(p) = p - ۱$$

مثال ۱۷ *فرض کنید $N = pq$ باشد که p, q اعدادی اول هستند، آنگاه*

$$\phi(N) = (p - ۱)(q - ۱)$$

۳ اعداد اول

برای کاربردهای رمزنگاری نیاز با تولید اعداد اول بزرگ هستیم. اولین الگوریتم قطعی چندجمله‌ای برای تشخیص اول بودن یک عدد در سال ۲۰۰۸ ارائه شد. اما الگوریتم‌های تصادفی از دیرباز مطرح بوده‌اند که هم اکنون نیز در عمل نسبت به الگوریتم قطعی بیشتر استفاده می‌شوند.

۱.۳ تعداد اعداد اول

قضیه ۹ (قضیه اعداد اول لاگرانژ) *تعداد اعداد اول کوچک‌تر از x را با $\pi(x)$ نشاد دهید، در این صورت:*

$$\lim_{x \to \infty} \frac{\pi(x)}{\frac{x}{\ln x}} = ۱$$

کران پایین نیز توسط چبیشف[۹] برای تعداد اعداد اول کوچک‌تر از x داده شده است:

$$\pi(x) \geq \frac{x}{۲ \log_۲^x}$$

پس احتمال اینکه عدد تصادفی n-بیتی اول باشد، حداقل $\frac{۱}{۲n}$ است،.لذا اگر الگوریتمی برای تشخیص اول بودن یک عدد در دسترس باشد، باید به طور متوسط آنرا بر روی حدود $۲n$ عدد تصادفی اجرا کرد تا یک عدد اول تولید کند. به طور دقیق‌تر اگر چنین الگوریتمی $۲cn$ بار اجرا شود، احتمال اینکه در هیچ یک از مراحل عدد اولی تولید نشود حداکثر برابر است با:

$$(۱ - \frac{۱}{۲n})^{۲cn} \leq e^{-c}$$

[۹] Chebyshev

۲.۳ آزمون اول بودن عدد

در این بخش الگوریتمی کارا برای پیدا کردن اعداد اول معرفی می‌شود.

قضیه ۱۰ (قضیه فرما) *اگر p عددی اول باشد، آنگاه به ازای هر $x \in \mathbb{Z}_p^*$ داریم:*

$$x^{p-۱} = ۱ \pmod p$$

مثال ۱۸ *فرض کنید $p = ۵$، آنگاه:*

$$۳^{۵-۱} = ۳^۴ = ۸۱ = ۱ \pmod ۵$$

نتیجه ۱۹ *به ازای هر $x \in \mathbb{Z}_p^*$، وارون ضربی x در $\mathbb{Z}_p^*$ برابر است با $x^{p-۲}$.*

برهان. به ازای هر x که عضو $\mathbb{Z}_p^*$ است، داریم $x^{p-۱} = ۱ \pmod p$. حال اگر این رابطه را بدین صورت بنویسیم که $x \cdot x^{p-۲} = ۱ \pmod p$ نشان دهنده این است که وارون ضربی برابر است با $x^{p-۲}$. ■

دقت کنید که لم فوق یک روش برای محاسبه معکوس ضربی در هنگ یک عدد اول با پیچیدگی $O(n^۳)$ پیشنهاد می‌کند. لذا در عمل بهتر است از تعمیم الگوریتم اقلیدس استفاده شود.

تعریف ۲۰ (آزمون فرما) *اگر m یک عدد صحیح باشد، گوییم m از آزمون فرما عبور می‌کند اگر به ازای هر $۱ \le a \le m-۱$ که $\gcd(a,m) = ۱$ داشته باشیم $a^{m-۱} = ۱ \pmod m$.*

نتیجه ۲۱ *اگر عدد صحیح m از آزمون فرما عبور نکند، قطعاً مرکب است. ولی اگر از آزمون فرما عبور کند، لزوماً اول نیست.*

تعریف ۲۲ (عدد کارمایل) *به عدد صحیح مرکب m که از آزمون فرما عبور می‌کند، عدد کارمایل[۱۰] می‌گوییم، یا به عبارت دیگر به عدد صحیح مرکب m کارمایل می‌گوییم اگر:*

$$\forall a, ۱ \le a \le m-۱, \gcd(a,m) = ۱ : a^{m-۱} = ۱ \pmod m$$

قضیه ۱۱ *اگر عدد صحیح مرکب m کارمایل نباشد، آنگاه به ازای حداقل نصف $۱ \le a \le m-۱$ ها*

$$a^{m-۱} \neq ۱$$

نکته ۴ *تنها اعداد مرکبی که از آزمون فرما عبور می‌کنند، اعداد کارمایکل هستند. با اینکه تعداد اعداد کارمایکل نامتناهی است اما کسر ناچیزی از اعداد کارمایکل هستند. کران‌های زیر در مورد $C(x)$، تعداد اعداد کارمایکل کوچکتر از x، وجود دارد:*

$$x^{۲/۷} < C(x) < x \exp\left(-\frac{\ln x \ln\ln\ln x}{\ln\ln x}\right)$$

علاوه بر آزمون فرما، آزمون دیگری به نام میلر-رابین وجود دارد که فقط اعداد اول از آن عبور می‌کنند و اعداد مرکب از آن عبور نمی‌کنند.

در عمل برای تشخیص اول بودن یک عدد m نمی‌توان با امتحان همه مقادیر $۱ \le a \le m-۱$ آزمون فرما را اعمال کرد. در عوض، برای تعداد معدودی a که به تصادف انتخاب می‌شوند برقراری رابطه‌های $\gcd(a,m) \neq ۱$ و $a^{m-۱} = ۱ \pmod m$ بررسی می‌شوند. اگر برای همه آنها این روابط برقرار باشد، تصمیم بر اول بودن m گرفته می‌شود. در غیر این‌صورت m مرکب است. اگه به تعداد ۸۰ تا مقدار تصادفی a امتحان شود، احتمال اینکه الگوریتم به اشتباه یک عدد تصادفی مرکب را اول اعلام نماید کمتر از $۲^{-۸۰}$ است.

[۱۰] Carmichael Number

۳.۳ الگوریتم تولید عدد اول

فرض کنید A الگوریتمی تصادفی باشد با ورودی عدد صحیح m که:

- اگر m اول باشد، همواره ۱ برمی‌گرداند.
- اگر m مرکب باشد، با احتمال ناچیزی (مثلا $۲^{-۸۰}$) خروجی ۱ برمی‌گرداند.

با استفاده از چنین الگوریتمی برای آزمون اولیه بودن اعداد می‌توان یک عدد اول تصادفی n بیتی به صورت زیر تولید کرد.

۱. یک عدد رشته $n-۱$ بیتی تصادفی را تولید و آنرا $\hat{m}$ بنامید. یعنی: $\hat{m} \leftarrow \{۰,۱\}^{n-۱}$.

۲. قرار دهید $m \leftarrow ۱||\hat{m}$.

۳. اگر خروجی $A(m) = ۱$ بود، عدد m را به عنوان خروجی الگوریتم برمی‌گردانیم.

۴.۳ تولید مولد برای گروه $\mathbb{Z}_p^*$

اگر p یک عدد اول باشد، گروه $\mathbb{Z}_p^*$ یک گروه دوری است و دارای $\phi(p-۱)$ مولد است. در حال حاضر الگوریتم کارایی برای تولید یک مولد برای گروه $\mathbb{Z}_p^*$ وجود ندارد مگر اینکه تجزیه به عوامل اول $p-۱$ را بدانیم. با این وجود می‌توان عدد اول p و مولد g را برای $\mathbb{Z}_p^*$ به طور همزمان تولید کرد. یک روش برای انجام این کار انتخاب عدد p به صورت $p = ۲q+۱$ به طوریکه q خود یک عدد اول باشد. یک عدد تصادفی n-بیتی با احتمال $۱/۴n^۲$ دارای این ویژگی است.

۴ تجزیه و لگاریتم گسسته

امنیت بسیاری از سیستم‌های رمز کلیدعمومی مبتنی بر مسائل مرتبط با دو مسأله تجزیه و لگاریتم گسسته است.

تعریف ۲۳ (مسأله تجزیه) *در مسأله تجزیه هدف پیدا کردن عوامل اول عدد N است که حاصل‌ضرب دو عدد اول بیتی تصادفی هم اندازه است.*

تعریف ۲۴ (مسأله لگاریتم گسسته روی گروه ضربی هنگی) *هدف پیدا کردن x از روی $g^x \mod p$ است که g مولد گروه ضربی $\mathbb{Z}_p^*$ است و $p-۱$ دارای یک عامل اول تقریباً هم‌اندازه با p است.*

به طور اعجاب‌آوری بهترین روش‌های شناخته شده حل این دو مسأله به‌ظاهر متفاوت مشابه هستند. الگوریتم NFS[۱۱] که در سال ۱۹۸۸ توسط پولارد[۱۲] برای مسأله تجزیه ارائه شد دارای پیچیدگی زمانی زیر است:

$$L(N) = O\left(e^{\sqrt[۳]{\frac{۶۴}{۹}\ln N(\ln\ln N)^۲}}\right)$$

نسخه لگاریتم گسسته این الگوریتم در زمان $L(p)$ مسأله لگاریتم گسسته روی $\mathbb{Z}_p^*$ را حل می‌کند. جدول زیر مقادیر تابع L را برای وقتی که N یا p یک عدد n-بیتی باشد، با صرف‌نظر از مقدار ثابت موجود در $O(\cdot)$ نشان می‌دهد:

[۱۱] Number Field Sieve (NFS)
[۱۲] Pollard

n	$\log_2 L(2^n)$
256	47
512	64
1024	87
2048	117

بزرگ‌ترین عدد مرکب تجزیه شده که حاصل‌ضرب دو عدد اول هم‌اندازه باشد یک عدد ۷۱۳ بیتی است که مقدار تابع L برای آن 2^{74} است، که تجزیه آن پس از ۲ سال محاسبه با پردازش موازی روی کامپیوترهای چندین مرکز دانشگاهی در سال ۲۰۱۰ انجام شد.

نکته ۵ *می‌توان مسأله لگاریتم گسسته را برای هر گروه متناهی $\mathbb{G}$ نیز درنظر گرفت. بر خلاف گروه $\mathbb{Z}_p^*$، برای بسیاری از گروه‌ها از جمله گروه‌هایی که مبتنی بر خم‌های بیضوی تعریف می‌شوند، بهترین الگوریتم شناخته شده نمایی با مرتبه $O(\sqrt{q})$ است که q بزرگترین عامل اول $|\mathbb{G}|$ است. این الگوریتم مبتنی بر تناقض روز تولد است و به صورت زیر کار می‌کند ...*