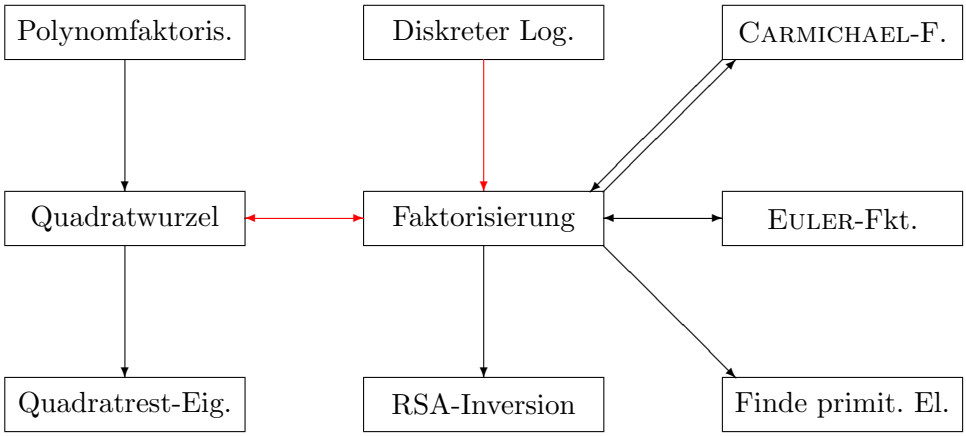
5 Harte zahlentheoretische Probleme

Die folgende Tabelle gibt einen Überblick über kryptologisch relevante zahlentheoretische Berechnungsprobleme. „Effizient“ bedeutet dabei „mit polynomialem Aufwand lösbar“.

Berechnungsproblem	**effizient?**	**behandelt in**
Primzahltest	ja	3.1–3.8
Für Primzahl p		
Finde primitives Element	ja (ERH oder prob.)	A.2, A.10
Finde quadratischen Nichtrest	ja (ERH oder prob.)	A.9
Quadratrest-Eigenschaft	ja	A.6
Quadratwurzel ziehen	ja (ERH oder prob.)	folgt in 5.3
Diskreter Logarithmus	? (vermutlich nein)	4.1, 4.6
Für zusammengesetzte Zahl n		
Faktorisierung	? (vermutlich nein)	2.2, 2.4
RSA-Inversion (e-te Wurzel)	? (vermutlich nein)	2.2
Berechn. der Euler-Funktion	? (vermutlich nein)	2.2
Berechn. der Carmichael-F.	? (vermutlich nein)	2.2
Finde primitives Element	? (vermutlich nein)	A.4
Quadratrest-Eigenschaft	? (vermutlich nein)	A.8
Quadratwurzel ziehen	? (vermutlich nein)	folgt in 5.5
Diskreter Logarithmus	? (vermutlich nein)	folgt in 5.1

„ERH“ bedeutet „unter Annahme der erweiterten Riemannschen Vermutung“, „prob.“ bedeutet „mit einem probabilistischen Algorithmus“.

Der Zusammenhang zwischen den Berechnungsproblemen für eine zusammengesetzte Zahl n wird in der folgenden Grafik dargestellt. Ein Pfeil von A nach B bedeutet dabei, dass das Problem B mit einem effizienten probabilistischen Algorithmus auf das Problem A reduzierbar ist. Die Umkehrungen bei den einfachen Pfeilen sind jeweils unbekannt. Die Reduktionen, die durch rote Pfeile bezeichnet sind, werden im folgenden bewiesen. (Z. T. nur für den Fall, dass n Produkt zweier Primzahlen ist.) [Die mit „Polynomfaktoris.“ bezeichnete Aufgabe, über dem Restklassenring $\mathbb{Z}/n\mathbb{Z}$ Polynome in einer Unbestimmten zu faktorisieren, wird hier nicht weiter behandelt.]


Polynomfaktoris.
Diskreter Log.
Carmichael-F.
Quadratwurzel
Faktorisierung
Euler-Fkt.
Quadratrest-Eig.
RSA-Inversion
Finde primit. El.

5.1 Diskreter Logarithmus und Faktorisierung

Für $a \in \mathbb{M}_n$ mit $\operatorname{Ord} a = s$ und zugehöriger Exponentialfunktion

$$\exp_a : \mathbb{Z} \longrightarrow \mathbb{M}_n$$

besteht das Problem des diskreten Logarithmus darin, einen Algorithmus zu finden, der für jedes $y \in \mathbb{M}_n$

- „nein“ ausgibt, wenn $y \notin \langle a \rangle$,
- sonst ein $r \in \mathbb{Z}$ ausgibt mit $0 \leq r < s$ und $y = a^r \bmod n$.

Satz 1 (E. Bach) *Sei $n = pq$ mit verschiedenen Primzahlen $p, q \geq 3$. Dann ist die Faktorisierung von n probabilistisch effizient auf das Problem des diskreten Logarithmus* mod n *reduzierbar.*

Beweis. Es ist $\varphi(n) = (p-1)(q-1)$. Für ein zufällig gewähltes $x \in \mathbb{M}_n$ ist stets $x^{\varphi(n)} \equiv 1 \pmod{n}$. Sei $y := x^n \bmod n$, also

$$y \equiv x^n \equiv x^{n-\varphi(n)} = x^{pq-(p-1)(q-1)} = x^{p+q-1} \pmod{n}.$$

Der diskrete Logarithmus liefert ein r mit $0 \leq r < \operatorname{Ord} x \leq \lambda(n)$ und $y = x^r \bmod n$. Also ist

$$x^{r-(p+q-1)} \equiv 1 \pmod{n}, \quad \operatorname{Ord} x \mid r - (p+q-1).$$

Da $|r - (p+q-1)| < \lambda(n)$, besteht eine große Wahrscheinlichkeit, dass $r = p+q-1$ – z. B. tritt das ein, wenn $\operatorname{Ord} x = \lambda(n)$. Andernfalls wird ein anderes x gewählt.

Aus den beiden Gleichungen

$$\begin{aligned} p+q &= r+1, \\ p \cdot q &= n \end{aligned}$$

sind die Faktoren p und q bestimmbar. $\diamond$

5.2 Quadratwurzeln und Faktorisierung

Satz 2 (M. RABIN) *Sei $n = pq$ mit verschiedenen Primzahlen $p, q \geq 3$. Dann ist die Faktorisierung von n probabilistisch effizient auf das Problem des Quadratwurzelziehens* mod n *reduzierbar.*

Beweis. Es gibt vier Einheitswurzeln in $\mathbb{Z}/n\mathbb{Z}$, also auch zu jedem Quadrat in $\mathbb{M}_n$ vier Quadratwurzeln.

Wählt man nun $x \in M_n$ zufällig, so liefert der Quadratwurzel-Algorithmus eine Wurzel $y \in \mathbb{M}_n$ von x^2, also

$$y^2 \equiv x^2 \pmod{n}.$$

Mit Wahrscheinlichkeit $\frac{1}{2}$ ist $y \not\equiv \pm x \pmod{n}$. Da

$$n \mid (x^2 - y^2) = (x+y)(x-y), \quad n \nmid (x \pm y),$$

ist $\mathrm{ggT}(n, x+y)$ echter Teiler von n. (Alternativ: $y/x \bmod n$ ist nichttriviale Einheitswurzel.) $\diamond$

D. h., wer Quadratwurzeln mod n ziehen kann, kann auch n faktorisieren. Die Umkehrung folgt in Abschnit 5.5.

5.3 Quadratwurzeln in endlichen Primkörpern

Oft ist das Ziehen von Quadratwurzeln trivial, wie die folgende einfache Überlegung zeigt:

Hilfssatz 1 *Sei G eine endliche Gruppe von ungerader Ordnung m. Dann gibt es zu jedem $a \in G$ genau ein $x \in G$ mit $x^2 = a$, nämlich $x = a^{\frac{m+1}{2}}$.*

Beweis. Da $a^m = 1$, ist $x^2 = a^{m+1} = a$. Insbesondere ist die Quadratabbildung $x \mapsto x^2$ surjektiv, also eine Bijektion $G \longrightarrow G$. $\diamond$

Hier soll gezeigt werden, wie man in einem endlichen Primkörper $\mathbb{F}_p$ effizient Quadratwurzeln zieht. Im Falle $p \equiv 3 \pmod 4$ ist das nach der Vorbemerkung besonders einfach: Ist $p = 4k + 3$, so hat die Gruppe $\mathbb{M}_p^2$ der Quadratreste die ungerade Ordnung $\frac{p-1}{2} = 2k + 1$. Ist also $z \in \mathbb{M}_p^2$ ein Quadratrest, so ist $x = z^{k+1} \bmod p$ die eindeutige Quadratwurzel in $\mathbb{M}_p^2$ [Lagrange 1769]. Der Aufwand für dieses Quadratwurzelziehen besteht aus höchstens $2 \cdot {}^2\log(p)$ Kongruenzmultiplikationen.

Beispiele

1. Für $p = 7 = 4 \cdot 1 + 3$ ist $k + 1 = 2$. Nach A.9 ist 2 Quadratrest; eine Wurzel ist $2^2 = 4$. Probe: $4^2 = 16 \equiv 2$.

2. Für $p = 23 = 4 \cdot 5 + 3$ ist $k + 1 = 6$. Nach A.9 ist 2 Quadratrest; eine Wurzel ist $2^6 = 64 \equiv 18$. Probe: $18^2 \equiv (-5)^2 = 25 \equiv 2$.

Ist $p \equiv 1 \pmod 4$, ist allerdings kein so einfaches Verfahren möglich. Es ist nämlich z. B. -1 ein Quadrat, aber keine Potenz von -1 kann gleichzeitig Quadratwurzel von -1 sein, da $[(-1)^m]^2 = (-1)^{2m} = 1 \neq -1$ immer gilt.

Es gibt aber u. a. ein allgemein funktionierendes Verfahren, das nach Adleman, Manders und Miller auch AMM benannt wird, im wesentlichen aber schon 1903 von Cipolla angegeben wurde. Dazu wird $p - 1$ zerlegt in $p - 1 = 2^e \cdot u$ mit ungeradem u. Ferner wählt man (ein- für allemal) einen beliebigen Nichtquadratrest $b \in \mathbb{F}_p^\times - \mathbb{M}_p^2$; dies ist der einzige nichtdeterministische Schritt – dazu siehe Abschnitt A.9. Insbesondere ist das Verfahren unter der allgemeinen Riemannschen Vermutung deterministisch, und natürlich ebenso in den vielen Fällen, wo man einen Nichtquadratrest sowieso kennt.

Nun soll aus dem Quadratrest $z \in \mathbb{M}_p^2$ die Wurzel gezogen werden. Da $z \in \mathbb{M}_p^2$, ist $\mathrm{Ord}(z) \mid \frac{p-1}{2}$, die Zweierordnung $r = \nu_2(\mathrm{Ord}(z))$ von $\mathrm{Ord}(z)$ also $\leq e - 1$, und r ist minimal mit $z^{u2^r} \equiv 1$.

Jetzt wird rekursiv eine Folge $z_1, z_2, \ldots$ gebildet:

$$z_1 = z \quad \text{mit } r_1 = \nu_2(\mathrm{Ord}(z_1)).$$

Ist bereits $z_i \in \mathbb{M}_p^2$ gewählt und r_i die Zweierordnung von $\mathrm{Ord}(z_i)$, so bricht die Folge ab, wenn $r_i = 0$; sonst wird

$$z_{i+1} = z_i \cdot b^{2^{e-r_i}}$$

gesetzt. Dann ist $z_{i+1} \in \mathbb{M}_p^2$. Ferner ist

$$z_{i+1}^{u \cdot 2^{r_i - 1}} \equiv z_i^{u \cdot 2^{r_i - 1}} \cdot b^{u \cdot 2^{e-1}} \equiv 1,$$

denn der erste Faktor ist $\equiv -1$, weil r_i minimal war, und der zweite $\equiv (\frac{b}{p}) = -1$, weil $u \cdot 2^{e-1} = \frac{p-1}{2}$. Also ist $r_{i+1} < r_i$. Der Abbruchpunkt $r_n = 0$ wird spätestens nach e Folgegliedern erreicht, $n \leq e \leq \log_2(p)$.

Dann wird rückwärts berechnet:

$$x_n = z_n^{\frac{u+1}{2}} \bmod p$$

mit $x_n^2 \equiv z_n^{u+1} \equiv z_n$ (denn $\mathrm{Ord}(z_n) \mid u$, da es ungerade ist). Und weiter:

$$x_i = x_{i+1} / b^{2^{e-r_i-1}} \bmod p,$$

das per Induktion

$$x_i^2 \equiv x_{i+1}^2 / b^{2^{e-r_i}} \equiv z_{i+1} / b^{2^{e-r_i}} \equiv z_i$$

erfüllt. Also ist $x = x_1$ die gesuchte Wurzel von z.

Abgesehen vom Aufwand, um b zu finden, sind folgende Schritte nötig:

- Berechnung der Potenzen $b^2, \ldots, b^{2^{e-1}}$, und das bedeutet $(e-1)$-mal modular quadrieren.
- Berechnung der Potenzen $b^u, b^{2u}, \ldots, b^{2^{e-1}u}$, und das bedeutet höchstens $2 \cdot \log_2(u) + e - 1$ Kongruenzmultiplikationen.
- Berechnung von z^u mit höchstens $2 \cdot \log_2(u)$ Kongruenzmultiplikationen.
- Dann wird für jedes $i = 1, \ldots, n \leq e$ berechnet:
  - z_i mit einer Kongruenzmultiplikation,
  - z_i^u aus z_{i-1}^u mit einer Kongruenzmultiplikation,
  - $z_i^{u2^r}$ aus $z_{i-1}^{u2^r}$ mit einer Kongruenzmultiplikation,
  - und daraus r_i.

  Das sind höchstens $3 \cdot (e-1)$ Kongruenzmultiplikationen.
- x_n als Potenz mit höchstens $2 \cdot \log_2(u)$ Kongruenzmultiplikationen.

- x_i aus x_{i+1} jeweils durch eine Kongruenzdivision mit Aufwand $O(\log(p)^2)$.

Das macht zusammen einen Aufwand der Größenordnung $O(\log(p)^3)$ mit einer eher kleinen Proportionalitätskonstanten.

Beispiel. Sei $p = 29$ und $z = 5$. Dann ist $p-1 = 4 \cdot 7$, also $e = 2$ und $u = 7$. Nach den obigen Bemerkungen ist $b = 2$ geeigneter Nichtquadratrest. Zu berechnen sind die Potenzen

$$b^2 = 4, b^u \equiv 128 \equiv 12, b^{2u} \equiv 144 \equiv -1,$$

$$z^2 \equiv 25 \equiv -4, z^4 \equiv 16, z^6 \equiv -64 \equiv -6, z^7 \equiv -30 \equiv -1.$$

Nun ist

$$z_1 = 5, z_1^u \equiv -1, z_1^{2u} \equiv 1, r_1 = 1,$$

$$z_2 \equiv z_1 b^2 \equiv 5 \cdot 4 = 20, z_2^u \equiv z_1^u b^{2u} \equiv (-1)(-1) = 1, r_2 = 0.$$

Jetzt geht's rückwärts:

$$x_2 \equiv z_2^{\frac{u+1}{2}} = z_2^4 = (z_2^2)^2 \equiv 400^2 \equiv (-6)^2 = 36 \equiv 7,$$

$$x_1 = x_2/b \bmod p = 7/2 \bmod 29 = 18.$$

Also ist $x = 18$ die gesuchte Wurzel. Probe: $18^2 = 324 \equiv 34 \equiv 5$.

Übungsaufgaben. Finde je einen deterministischen Algorithmus (eine einfache Formel) zum Ziehen von Quadratwurzeln in den Körpern

- $\mathbb{F}_p$ mit $p \equiv 5 \pmod 8$,
- $\mathbb{F}_{2^m}$ mit $m \geq 2$. [Ansätze: 1. Betrachte die Ordnung des jeweiligen Körperelements in der multiplikativen Gruppe. 2. Invertiere die lineare Abbildung $x \mapsto x^2$.]
- Was wird aus dem Algorithmus für einen Körper mit Primpotenzordnung $q = p^m$?

Alternative Algorithmen. Fast alle bekannten effizienten Algorithmen, die den Fall $p \equiv 1 \pmod 4$ vollständig abdecken, sind probabilistisch; ihre deterministische Version ist unter der erweiterten Riemannschen Vermutung noch von polynomialem Aufwand. In dem Buch von Forster (*Algorithmische Zahlentheorie*) wird eine Variante des Cipolla/ AMM-Algorithmus beschrieben, die die quadratische Körpererweiterung $\mathbb{F}_{p^2} \supseteq \mathbb{F}_p$ benützt und konzeptionell besonders einfach ist. Im *Handbook of Applied Cryptography* (Menezes/ van Oorschot/ Vanstone) wird ein Algorithmus angegeben, der von Tonelli 1891 stammt und recht kurz zu formulieren ist, aber den Aufwand $O(\log(p)^4)$ benötigt. Eine weitere Methode ist ein Spezialfall

des Verfahrens von CANTOR/ ZASSENHAUS zur Faktorisierung von Polynomen über endlichen Körpern, siehe VON ZUR GATHEN/ GERHARD: *Modern Computer Algebra.* Ein weiteres Verfahren beruht auf der LUCAS-Folge (a_n) mit $a_1 = b$, $a_2 = b^2 - 2z$, wobei $b^2 - 4z$ Nicht-Quadratrest ist; dieses Verfahren stammt von LEHMER [keine Referenz]. Der einzige bekannte deterministische Algorithmus mit polynomialem Aufwand stammt von SCHOOF, verwendet die Theorie der elliptischen Kurven und ist praktisch unterlegen – Aufwand $O(\log(p)^9)$.

Für einen Überblick siehe:

- E. BACH/ J. SHALLIT: *Algorithmic Number Theory.* MIT Press, Cambridge Mass. 1996.
- D. J. BERNSTEIN: Faster square roots in annoying finite fields. Preprint (siehe die Homepage des Autors `http://cr.yp.to/`).

5.4 Quadratwurzeln bei Primzahlpotenz-Moduln

Mit einem einfachen Verfahren (hinter dem das HENSELsche Lemma steckt) kann man von Primzahlmoduln zu Primpotenzmoduln übergehen. Sei p eine Primzahl $\neq 2$ und $e \geq 2$. Sei z ein Quadratrest $\bmod p^e$. Gesucht ist eine passende Quadratwurzel.

Natürlich ist z auch Quadratrest $\bmod p^{e-1}$. Angenommen, dafür haben wir schon eine Wurzel gefunden, also ein y mit $y^2 \equiv z \pmod{p^{e-1}}$. Sei

$$a = 1/2y \bmod p$$

und $y^2 - z = p^{e-1} \cdot u$. Dann wird

$$x = y - a \cdot (y^2 - z) \bmod p^e$$

gesetzt. Damit gilt

$$\begin{aligned} x^2 &\equiv y^2 - 2ay(y^2 - z) + a^2(y^2 - z)^2 \equiv y^2 - 2ayp^{e-1}u \\ &\equiv y^2 - p^{e-1}u = z \pmod{p^e}. \end{aligned}$$

Also ist x die gesuchte Wurzel.

Dieser Algorithmus soll hier nicht explizit aufgeschrieben, aber an zwei Beispielen verdeutlicht werden:

Beispiele

1. $n = 25$, $z = 19$. Es ist $p = 5$, $19 \bmod 5 = 4$. Also kann man $y = 2$ und $a = 1/4 \bmod 5 = 4$ nehmen. Dann ist $y^2 - z = -15$ und

   $$x = 2 + 15 \cdot 4 \bmod 25 = 62 \bmod 25 = 12.$$

   Probe: $12^2 = 144 = 125 + 19$.

2. $n = 27$, $z = 19$. Es ist $p = 3$, $19 \bmod 3 = 1$. Also kann man im ersten Schritt $y = 1$ und $a = 1/2 \bmod 3 = 2$ nehmen. Dann ist $y^2 - z = -18$ und

   $$x = 1 + 2 \cdot 18 \bmod 9 = 37 \bmod 9 = 1.$$

   Beim zweiten Schritt (von 9 nach 27) ist also wieder $y = 1$, $y^2 - z = -18$ und damit

   $$x = 37 \bmod 27 = 10.$$

   Probe: $10^2 = 100 = 81 + 19$.

Der Aufwand besteht aus zwei Teilen

1. $\bmod p$ wird eine Wurzel gezogen und einmal dividiert. (Der Quotient a muss insgesamt nur einmal bestimmt werden, da $x \equiv y \pmod p$.)

2. Bei jeder Erhöhung des Exponenten sind zwei Kongruenzmultiplikationen und zwei Subtraktionen fällig.

Der Gesamtaufwand bleibt also $O(\log(n)^3)$, wenn n der Modul ist.

Es bleibt noch der Fall zu untersuchen, dass $n = 2^e$ eine Zweierpotenz ist. Ist $e \leq 3$, so ist 1 der einzige Quadratrest, und seine Wurzel ist 1. Für größere Exponenten e wird wieder auf $e - 1$ rekurriert: Sei z eine ungerade Zahl (alle invertierbaren Elemente sind ungerade). Sei y bereits gefunden mit $y^2 \equiv z \pmod{2^{e-1}}$. Dann ist $y^2 - z = 2^{e-1} \cdot t$. Ist t gerade, so auch $y^2 \equiv z \pmod{2^e}$. Andernfalls setzt man $x = y + 2^{e-2}$. Dann ist

$$x^2 \equiv y^2 + 2^{e-1}y + 2^{2e-4} \equiv z + 2^{e-1} \cdot (t + y) \equiv z \pmod{2^e},$$

da $t + y$ gerade ist. Also ist $x = y$ oder $y + 2^{e-2}$ die gesuchte Wurzel. Der Gesamtaufwand ist hier sogar kleiner als $O(\log(n)^3)$.

Nebenbei haben wir gezeigt, dass z genau dann Quadratrest mod 2^e ist (für $e \geq 3$), wenn $z \equiv 1 \pmod{8}$.

5.5 Quadratwurzeln bei zusammengesetzten Moduln

Ist die Primzerlegung eines Moduls n bekannt, so lassen sich in $\mathbb{M}_n$ effizient Quadratwurzeln ziehen; die Probleme „Faktorisierung“ und „Ziehen von Quadratwurzeln“ sind also in ihrer Komplexität äquivalent.

Zur Durchführung wird n sukzessive in teilerfremde Faktoren zerlegt (bis hinunter zu den Primpotenzen).

Sei also $n = rs$ mit teilerfremden Faktoren r und s. Zuerst werden mit dem erweiterten Euklidischen Algorithmus Koeffizienten a und b mit $ar + bs = 1$ bestimmt. Aus z soll die Quadratwurzel gezogen werden. Sei u die Quadratwurzel $\bmod r$ und v die Quadratwurzel $\bmod s$. Dann erfüllt $x = arv + bsu \bmod n$:

$$\begin{aligned} x &\equiv bsu \equiv u \pmod{r}, & x &\equiv arv \equiv v \pmod{s}, \\ x^2 &\equiv u^2 \equiv z \pmod{r}, & x^2 &\equiv v^2 \equiv z \pmod{s}, \end{aligned}$$

insbesondere ist $x^2 \equiv z \pmod{n}$.

Der Aufwand für dieses Verfahren besteht aus zwei Quadratwurzeln modulo den Faktoren, einem Euklidischen Algorithmus und 4 Kongruenzmultiplikationen (+ 1 Kongruenzaddition). Er bleibt also in der Größenordnung $\mathrm{O}(\log(n)^3)$.

Für Blum-Zahlen gibt es sogar einen noch einfacheren Algorithmus, nämlich eine explizite Formel:

Korollar 1 *Sei $n = pq$ mit Primzahlen $p, q \equiv 3 \pmod 4$. Dann gilt*

(i) *$d = \frac{(p-1)(q-1)+4}{8}$ ist ganzzahlig.*

(ii) *Für jedes Quadrat $x \in \mathbb{M}_n^2$ ist x^d Quadratwurzel aus x.*

Beweis. (i) Ist $p = 4k + 3$, $q = 4l + 3$, so $(p-1)(q-1) = 16kl + 8k + 8l + 4$, also $d = 2kl + k + l + 1$.

(ii) Der Exponent der multiplikativen Gruppe $\mathbb{M}_n$,

$$\lambda(n) = \mathrm{kgV}(p-1, q-1) = 2 \cdot \mathrm{kgV}(2k+1, 2l+1)$$

ist Teiler von $2 \cdot (2k+1) \cdot (2l+1)$, der Exponent der Quadrat-Untergruppe $\mathbb{M}_n^2$ ist $\frac{\lambda(n)}{2}$, also Teiler von $(2k+1) \cdot (2l+1) = 4kl + 2k + 2l + 1 = 2d - 1$. Also gilt $x^{2d} \equiv x \pmod{n}$ für alle $x \in \mathbb{M}_n^2$, d. h., das Quadrat von x^d ergibt x. $\diamond$

Diese einfache Formel bewirkt, dass das Verschlüsselungsverfahren von Rabin für Blum-Moduln besonders leicht zu handhaben ist.