# Zwakke sleutels bij het RSA-cryptosysteem

## DEEL 1

[ Benne de Weger ]

### 1. Inleiding

In het nieuwe vak Wiskunde D is er, gelukkig, aandacht mogelijk voor cryptografie als een leuke en veelgebruikte toepassing van wiskunde. Daarbij kan goed gekozen worden voor het behandelen van het RSA-cryptosysteem (zie paragraaf 3 voor de herkomst van de afkorting RSA). Een tekst over cryptografie voor gebruik bij Wiskunde D die dit doet, is [1; Lambeck]. Aantrekkelijke aspecten van RSA zijn dat de wiskundige basis ervan goed te begrijpen is voor leerlingen in de bovenbouw van het vwo, en dat de wiskunde die er bij komt kijken, niet de standaardwiskunde uit de hoek van de analyse of de statistiek is. Zo krijgen de leerlingen een goede illustratie van de breedheid en de toepasbaarheid van de wiskunde. Daarnaast laat het leerlingen kennismaken met een actief onderzoeksgebied in de wiskunde, dat ook andere disciplines raakt, zoals informatica.
Bij het behandelen van RSA kan goed uitgelegd worden hoe elementaire getaltheorie een cruciale rol speelt in moderne technieken voor beveiliging van informatie, zoals vertrouwelijkheid (versleuteling) en authenticatie (digitale handtekeningen). Onderwijsteksten waarin de basisideeën van RSA worden uitgelegd komen echter nogal eens niet verder dan te vertellen hoe een sleutelpaar in elkaar zit, welke rol het ontbinden in factoren speelt bij het kraken van de sleutel, en hoe de RSA-operaties in hun werk gaan. Dat is wel te begrijpen, want die aspecten van het onderwerp hebben al een redelijke omvang, geven een aardig beeld van moderne cryptografie, en er valt al genoeg plezier aan te beleven.
In dit artikel, verdeeld in twee delen, willen we een minder bekend aspect van RSA wat verder uitdiepen: het bestaan van zogeheten *zwakke sleutels*. Dat zijn sleutels die vermeden moeten worden, omdat het gebruik ervan leidt tot het uitlekken van de geheime sleutel. Dit is een actief onderzoeksgebied in de wiskundige cryptologie. Enkele basisresultaten hieruit vereisen alleen elementaire getaltheoretische voorkennis, en die zullen we bespreken. Het doel daarvan is de docent die RSA wil behandelen, wat meer achtergrond te geven. Allicht zijn er docenten die ook iets hiervan willen doorgeven aan hun leerlingen.
Dit artikel kent de volgende opbouw: na het kort vermelden van twee belangrijke resultaten uit de getaltheorie in paragraaf 2, beschrijven we kort, in de paragrafen 3, 4 en 5, de basis van RSA, hoe sleutelparen bij RSA er uitzien en hoe versleutelen en ontsleutelen werkt, en wat kraken van een sleutelpaar betekent. Daarna behandelen we twee methoden om een RSA-sleutelpaar te kraken: in paragraaf 6, nog in dit deel, de methode van Fermat die direct werkt op het ontbinden van bepaalde, verkeerd gekozen, grote getallen, en in het te verschijnen tweede deel van dit artikel, in paragraaf 7, de methode van Wiener die op een andere manier de privésleutel aanvalt als die zwak gekozen is. Beide methoden leiden tot een volledige kraak van het slecht gekozen RSA-sleutelpaar, en berusten op elementaire getaltheorie. We sluiten dan concluderend af in paragraaf 8.

### 2. Getaltheorie

RSA is gebaseerd op elementaire getaltheorie, en wel op modulo-rekenen met een vaste *modulus*. Zo'n modulus is een vast geheel getal $n$, en modulo-rekenen betekent dat bij het rekenen veelvouden van $n$ 'verwaarloosd' worden, zoals we bij klokrekenen een veelvoud van 12 (of 24) niet meenemen. De notatie hiervoor is: $a \equiv b \pmod{n}$ als $a - b$ een veelvoud van $n$ is. In zo'n geval 'identificeren' we $a$ en $b$, net zoals we zeggen dat het 2 uur na 11 uur niet 13 uur is, maar 1 uur, want $11 + 2 = 13 \equiv 1 \pmod{12}$.
De rekenregels voor optellen, aftrekken en vermenigvuldigen gelden ook voor modulo-rekenen. Daarbij maakt het voor het antwoord niet uit op welk moment (vóór de berekening, of achteraf) er een veelvoud van de modulus bij een getal wordt opgeteld of afgetrokken. Meestal is het handig de getallen zo veel mogelijk terug te brengen naar de verzameling $\{0, 1, 2, \ldots, n-1\}$ door er het juiste aantal keer de modulus $n$ van af te trekken of bij op te tellen.
Zo is bijvoorbeeld $25 \times (2 - 16) \pmod{17}$ te berekenen door eerst het gedeelte '(mod 17)' te negeren:
$25 \times (2 - 16) = 25 \times (-14) = -350$
en dan het juiste aantal malen 17 er bij op te tellen:
$-350 \equiv -350 + 21 \times 17 = 7 \pmod{17}$
Maar u mag net zo goed de 25 eerst vervangen door $25 - 17 = 8$ en de -16 door +1. Dan gaat het zo:
$25 \times (2 - 16) \equiv 8 \times (2 + 1) = 8 \times 3 = 24 \equiv 7 \pmod{17}$
Dat laatste heeft als voordeel dat de getallen altijd klein blijven.
Bij delen ligt het iets subtieler: dan geldt als extra voorwaarde voor het bestaan van de deling $a/b \pmod{n}$ dat de noemer $b$ en de modulus $n$ geen gemeenschappelijke deler hebben (anders dan 1). Daarop gaan we aan het eind van deze paragraaf nog kort in. Zie [2; de Weger] voor meer details.
Bij machtsverheffen is er iets aparts aan de hand. Omdat machtsverheffen niets anders is dan herhaald vermenigvuldigen, geldt dat als $a \equiv b \pmod{n}$ dan ook $a^2 \equiv b^2 \pmod{n}$, $a^3 \equiv b^3 \pmod{n}$, …, $a^k \equiv b^k \pmod{n}$, voor alle exponenten $k$.
Bijvoorbeeld, voor de machten van 7 modulo 15 vinden we:
$7^0 = 1$, $7^1 = 7$,
$7^2 = 49 \equiv 4 \pmod{15}$,
$7^3 = 7 \times 7^2 = 7 \times 4 = 28 \equiv 13 \pmod{15}$,
$7^4 = 7 \times 7^3 \equiv 7 \times 13 = 91 \equiv 1 \pmod{15}$.
En dan begint het van voren af aan:
$7^5 \equiv 7 \pmod{15}$, $7^6 \equiv 4 \pmod{15}$,
$7^7 \equiv 13 \pmod{15}$, enzovoorts.
Blijkbaar is $7^k \equiv 7^j \pmod{15}$ niet per se waar als $k \equiv j \pmod{15}$, maar wel als $k \equiv j \pmod{4}$.
De belangrijkste stelling in dit verband is de *Stelling van Euler*, die we hier alleen weergeven voor het speciale geval dat we bij RSA tegenkomen:

| | | |
|---|---|---|
| Invoer: twee positieve gehele getallen $a$, $b$ | | |
| Uitvoer: $d = \text{ggd}(a, b)$, en $u$, $v$ zodat $d = ua + vb$ | | |
| $d_{\text{nieuw}} := a$; $d := b$; $u_{\text{nieuw}} := 1$; $u := 0$; $v_{\text{nieuw}} := 0$; $v := 1$ | | |
| Herhaal | | |
| | Bereken het gehele deel $q$ van $d_{\text{nieuw}} / d$, en de rest $r$ | |
| | Als de rest $r \neq 0$: | |
| | Dan | vervang $d_{\text{nieuw}}$ door $(d_{\text{nieuw}} - qd)$; verwissel dan $d$ en $d_{\text{nieuw}}$<br>vervang $u_{\text{nieuw}}$ door $(u_{\text{nieuw}} - qu)$; verwissel dan $u$ en $u_{\text{nieuw}}$<br>vervang $v_{\text{nieuw}}$ door $(v_{\text{nieuw}} - qv)$; verwissel dan $v$ en $v_{\text{nieuw}}$<br>en ga door (met herhalen) |
| | Anders | Stop (de herhaling) |
| Druk af: $d$, $u$, $v$ | | |

figuur 1 Uitgebreide algoritme van Euclides

***Voorbeeld*** - Het berekenen van $d = \text{ggd}(62, 23)$

| $q$ | $r$ | $d_{\text{nieuw}}$ | $d\ (= r)$ | $u_{\text{nieuw}}$ | $u$ | $v_{\text{nieuw}}$ | $v$ |
|---|---|---|---|---|---|---|---|
| | | 62 | 23 | 1 | 0 | 0 | 1 |
| 2 | 16 | 23 | 16 | 0 | 1 | 1 | -2 |
| 1 | 7 | 16 | 7 | 1 | -1 | -2 | 3 |
| 2 | 2 | 7 | 2 | -1 | 3 | 3 | -8 |
| 3 | 1 | 2 | 1 | 3 | -10 | -8 | 27 |
| 2 | 0 | | | | | | |

De conclusie is dat $\text{ggd}(62, 23) = 1 = -10 \times 62 + 27 \times 23$. Let erop dat in iedere regel van de tabel geldt dat $d = u \times 62 + v \times 23$.

***Stelling van Euler (speciaal geval)***: Laten $p$ en $q$ twee verschillende priemgetallen zijn. Laat $n = pq$ (dit is de modulus) en $\phi(n) = (p - 1)(q - 1)$ (de $\phi$-functie van Euler). Laat $a \in \{1, 2, \ldots, n - 1\}$ niet door $p$ of $q$ deelbaar zijn.
Dan geldt: $a^{\phi(n)} \equiv 1 \pmod{n}$.

Bij het rekenen modulo $n$ geldt voor machtsverheffen, het berekenen van $a^k \pmod{n}$, dus dat de modulus $n$ alleen voor het grondtal geldt, maar voor de exponent een andere modulus gehanteerd moet worden, namelijk $\phi(n)$.
Een algoritme dat we een paar keer nodig zullen hebben, is het *Uitgebreide Algoritme van Euclides*. Het algoritme van Euclides berekent de grootste gemene deler van twee getallen. De ggd van twee getallen is altijd een lineaire combinatie van die getallen, met andere woorden: als $d = \text{ggd}(a, b)$, dan zijn er gehele coëfficiënten $u$ en $v$ zodat $d = ua + vb$. Het uitgebreide algoritme van Euclides berekent met de ggd ook die coëfficiënten; ***zie figuur 1*** en het voorbeeld eronder.

De werking van het algoritme wordt verklaard door de volgende eigenschappen te zien:

- de verzameling van gemeenschappelijke delers van $d$ en $d_{\text{nieuw}}$ verandert bij het vervangen van $d_{\text{nieuw}}$ niet; de grootste gemeenschappelijke deler dus ook niet; aan het begin van het uitvoeren van het algoritme is deze $\text{ggd}(a,b)$, en aan het einde is $d$ een deler van $d_{\text{nieuw}}$ en is $d$ dus ook de grootste gemeenschappelijke deler van $d$ en $d_{\text{nieuw}}$ en dus van $a$ en $b$;
- bij het begin van het algoritme geldt $d = u \times a + v \times b$ (want dan is $u = 0$, $v = 1$ en $d = b$), de drie vervangregels in het algoritme behouden deze eigenschap, en ze geldt aan het eind dus nog steeds;
- zolang de rest niet 0 is, wordt $d_{\text{nieuw}}$ vervangen door een getal dat echt kleiner is; het algoritme stopt dus na een eindig aantal stappen.

We geven nog een voorbeeld, met een toepassing op het delen bij modulo-rekenen (zie onder).

### 3. Sleutelparen bij RSA

RSA is een bekend cryptografisch systeem, genoemd naar Ron Rivest, Adi Shamir en Len Adleman, die het in de jaren '70 van de vorige eeuw bedacht hebben. RSA is onder andere geïmplementeerd in vrijwel alle webbrowsers, voor het beveiligen van internetcommunicatie.
RSA is een *asymmetrisch* of *publieke-sleutel-cryptosysteem*. Dat wil zeggen dat het werkt met *sleutelparen*. Een sleutelpaar is een tweetal bij elkaar behorende sleutels: een *publieke sleutel* en een *privésleutel*. Als RSA wordt gebruikt voor geheimschrifttoepassingen, dan moet met de publieke sleutel worden versleuteld, en met de privésleutel ontsleuteld.

***Voorbeeld.*** Zoek $k$ waarvoor geldt dat $46k \equiv 1 \pmod{77}$, mits $k$ bestaat.
Dit kan door het uitgebreide algoritme van Euclides toe te passen op $a = 77$ en $b = 46$, want dat geeft $d = \text{ggd}(77, 46)$ en $u$, $v$ met $d = 77u + 46v$.
Als het zo blijkt te zijn dat $d = 1$, dan zien we dat $46v = 1 - 77u \equiv 1 \pmod{77}$, en dan kunnen we dus $k = v$ nemen.
En als $d \neq 1$ blijkt te zijn, dan bestaat zo'n $k$ niet, want dan zou $d$ ook een deler van 1 moeten zijn.

| $d$ | $r$ | $d_{\text{nieuw}}$ | $d\ (= r)$ | $u_{\text{nieuw}}$ | $u$ | $v_{\text{nieuw}}$ | $v$ |
|---|---|---|---|---|---|---|---|
| | | 77 | 46 | 1 | 0 | 0 | 1 |
| 1 | 31 | 46 | 31 | 0 | 1 | 1 | -1 |
| 1 | 15 | 31 | 15 | 1 | -1 | -1 | 2 |
| 2 | 1 | 15 | 1 | -2 | 3 | 3 | -5 |
| 15 | 0 | | | | | | |

De conclusie is dat zo'n $k$ bestaat, en wel $k = -5$. Ter controle:
$46 \times (-5) = -230 \equiv -230 + 3 \times 77 = 1 \pmod{77}$
Merk op dat we hier in feite een deling hebben uitgerekend: $1/46 \pmod{77}$.

Een RSA-sleutelpaar wordt als volgt gemaakt. In paragraaf 4 zal dan duidelijk worden gemaakt dat met een op deze manier gekozen sleutelpaar handig kan worden versleuteld en ontsleuteld.

1. Kies een *veiligheidsparameter s*, het aantal bits dat nodig is om de modulus weer te geven in het tweetallig stelsel. Een op dit moment nog redelijk veilige keuze is $s = 1024$. Dat betekent dat we gaan werken met getallen in de orde van grootte van $2^{1024}$; dat is ruim 300 decimale cijfers.
2. Kies twee willekeurige priemgetallen $p$, $q$ van elk ongeveer $\frac{1}{2}s$ bits (met $s = 1024$ dus ruim 150 cijfers). Laten we $p > q$ nemen. We leggen hier niet uit waarom zulke grote priemgetallen bestaan, zelfs in grote overvloed, en hoe ze gevonden kunnen worden. Er zijn efficiënte methoden voor; zie [2; de Weger].
3. Bereken de *modulus* $n = pq$ van $s$ bits (het zouden er $s - 1$ kunnen zijn, dat is niet erg; als u dat wel erg vindt, kies dan nieuwe $p$ en $q$).
4. Bereken $\phi(n) = (p - 1)(q - 1)$.
5. Kies een *publieke exponent e* en een *privé-exponent d*, beide groter dan 2 en kleiner dan $\phi(n) - 2$, die de relatie $ed \equiv 1 \pmod{\phi(n)}$ bezitten. Dat kan als volgt. Kies één van de twee, willekeurig of volgens een bepaald stramien, zolang die exponent maar geen deler gemeen heeft met $\phi(n)$. Bereken dan de andere door het uitgebreide algoritme van Euclides toe te passen op de eerste gekozen exponent en $\phi(n)$. Zie het laatste voorbeeld onderaan pag. 257.
6. Gooi $p$, $q$ en $\phi(n)$ weg, want die zijn niet meer nodig, en, ze mogen beslist niet in verkeerde handen vallen.
7. De publieke sleutel is nu het paar $(n, e)$, en de privésleutel is het paar $(n, d)$.

***Voorbeeld*** met veiligheidsparameter $s = 16$.
$p = 211$, $q = 197$, $n = 211 \times 197 = 41567$
$\phi(n) = 210 \times 196 = 41160$
$e = 24377$, $d = 17393$
Nu is inderdaad aan de relatie $ed \equiv 1 \pmod{\phi(n)}$ voldaan, want:
$24377 \times 17393 = 1 + 10301 \times 41160$
De publieke sleutel is:
$(n, e) = (41567, 24377)$.
De privésleutel is: $(n, d) = (41567, 17393)$.

*Opmerking. Bij het boek* [2; de Weger] *hoort een webpagina met een Java-applet waarmee u dergelijke berekeningen zelf eenvoudig kunt uitvoeren, zelfs met grotere getallen. Het is aan te raden alle voorbeelden bij dit artikel na te rekenen, en ook eens met andere getallen te proberen. Een rekenhulpje als de genoemde applet is dan onmisbaar.*

De privésleutel $(n, d)$ moet goed bewaakt worden door de eigenaar. Als de privé-exponent $d$ in verkeerde handen valt, is de veiligheid van het sleutelpaar helemaal weg. De publieke sleutel $(n, e)$ is daarentegen *echt* publiek: die mag aan iedereen bekend gemaakt worden. In het bijzonder moeten we ervan uitgaan dat ook een eventuele kraker de modulus $n$ en de publieke exponent $e$ weet.

## 4. Versleutelen en ontsleutelen met RSA

Veronderstel dat Benne een geheime boodschap aan Alda wil sturen, maar hij heeft alleen de beschikking over een onveilig communicatiekanaal (bijvoorbeeld het internet), dat namelijk wordt afgeluisterd door Karel. Alda heeft een RSA-sleutelpaar gemaakt, en heeft de publieke sleutel $(n, e)$ aan Benne gegeven. De privésleutel $(n, d)$ houdt zij angstvallig geheim. Benne codeert de boodschap tot een *bericht* in de vorm van een getal $b$, dat tussen 1 en $n$ ligt, en geen deler met $n$ gemeen heeft. Met behulp van Alda's publieke sleutel $(n, e)$ kan hij nu het *geheimschrift* $g$ berekenen als:
$g \equiv b^e \pmod{n}$
Dit geheimschrift stuurt hij naar Alda. Dat Karel het geheimschrift $g$ kan afluisteren is helemaal niet erg; hij heeft immers niet de sleutel waarmee het ontcijferd kan worden. Alléén Alda heeft die privésleutel $(n, d)$. Zij kan daarmee het bericht $b$ weer terugvinden uit het geheimschrift $g$ door het berekenen van:
$b \equiv g^d \pmod{n}$
Dit gaat goed, want de relatie tussen $e$ en $d$ geeft het bestaan van een $k$ zodat:
$ed = 1 + k \cdot \phi(n)$
En nu volgt met behulp van de Stelling van Euler dat:
$g^d \equiv (b^e)^d = b^{ed} = b^{1+k\cdot\phi(n)} =$
$b \cdot (b^{\phi(n)})^k \equiv b \cdot 1^k = b \pmod{n}$

***Voorbeeld***
Stel dat Alda's sleutelpaar bestaat uit haar publieke sleutel: $(n, e) = (41567, 24377)$ en haar privésleutel: $(n, d) = (41567, 17393)$.
Benne wil het bericht 'lief' versturen aan Alda. Dat kan hij bijvoorbeeld coderen (volgens a=01, b=02, …, z=26) als de twee getallen: 1209, 506.
Benne versleutelt deze getallen met Alda's publieke sleutel als volgt:
$120924377 \equiv 1671 \pmod{41567}$
$50624377 \equiv 12949 \pmod{41567}$
Benne verstuurt als geheimschrift de getallen 1671, 12949 aan Alda. Een afluisteraar kan hier niets mee. Alleen Alda kan het geheimschrift ontcijferen met haar privésleutel, en wel als:
$167117393 \equiv 1209 \pmod{41567}$,
$1294917393 \equiv 506 \pmod{41567}$
Uit de getallen 1209, 506 kan Alda meteen de boodschap 'lief' decoderen.

*Terzijde. RSA wordt in de praktijk niet op de hierboven beschreven manier gebruikt. Daarvoor zijn de machtsverheffingen van grote getallen veel te veel rekenwerk. De boodschap wordt doorgaans met een veel efficiënter symmetrisch cryptosysteem, zoals AES (Rijndael), versleuteld. De sleutel daarvoor is klein (128 bits), wordt door de verzender willekeurig gekozen (voor ieder bericht een andere), en moet nu op een veilige manier naar de ontvanger gestuurd worden. Alleen de AES-sleutel wordt dan met RSA versleuteld; dat is wel efficiënt want het is maar een korte rij bits. De versleutelde sleutel wordt dan met het geheimschrift meegestuurd. De ontvanger moet nu eerst RSA gebruiken om de AES-sleutel terug te kunnen vinden, en kan dan daarmee het geheimschrift met AES ontsleutelen.*

## 5. Kraken van de RSA-sleutel

Een kraker heeft alleen de beschikking over publieke informatie, en wil daaruit graag geheime informatie achterhalen. Bij RSA bestaat de publieke informatie uit de modulus $n$ en de publieke exponent $e$. Het doel van de kraker is de privé-exponent $d$ te achterhalen, want dan kan hij geheime boodschappen gaan ontcijferen. De kraker kan wel raden dat de modulus twee priemfactoren heeft, maar niet welke dat zijn. Ook $\phi(n)$ kent hij niet.

Twee manieren om een RSA-sleutelpaar te kraken zijn:

- Proberen de modulus $n$ te ontbinden in zijn factoren: $n = p \cdot q$. Als u dat kunt, dan kunt u makkelijk $\phi(n) = (p-1)(q-1)$ berekenen, en dan is het vinden van de privé-exponent $d$ makkelijk, net zoals bij het maken van een sleutelpaar.
- Proberen direct de privé-exponent $d$ te berekenen. Als u dat kunt, dan kunt u ook zonder $p$ en $q$ te weten ontsleutelen, want dat ging namelijk met $b \equiv g^d \pmod{n}$.

Van beide methoden geven we een voorbeeld, waarbij kraken lukt omdat het sleutelpaar op een zwakke manier gekozen was.

## 6. De methode van Fermat voor het ontbinden van de modulus

De eerste methode voor het kraken van RSA die we bekijken, grijpt alleen aan op de modulus, en probeert die te factoriseren. Het idee, afkomstig van de bekende Pierre de Fermat, is gebaseerd op het *merkwaardige product*

$X^2 - Y^2 = (X+Y)(X-Y)$

Hier zien we dat ieder verschil van twee kwadraten makkelijk te vinden factoren heeft. Voor oneven $n$ is er altijd een triviale manier om $n$ te schrijven als $X^2 - Y^2$, namelijk met

$X = \frac{1}{2}(n+1)$, $Y = \frac{1}{2}(n-1)$

Dit zegt niets over de priemfactoren van $n$, want nu is $X+Y=n$, $X-Y=1$. U moet op zoek gaan naar andere $X$, $Y$ met $n = X^2 - Y^2$ om een echte ontbinding te vinden. Omdat $p$ en $q$ priemgetallen zijn met $p > q$, zijn er geen andere oplossingen $X$, $Y$ van $n = pq = X^2 - Y^2$ dan de bovengenoemde triviale, en de oplossing gegeven door $X+Y=p$, $X-Y=q$. En die oplossing is:

$X = \frac{1}{2}(p+q)$, $Y = \frac{1}{2}(p-q)$

Uit $p > q$ en $n = pq$ volgt dat $p > \sqrt{n}$. Een eenvoudige zoekmethode begint nu met $X$-en achtereenvolgens uit te proberen, te beginnen bij het kleinste gehele getal dat groter is dan $\sqrt{n}$ (het kleinste gehele getal groter dan of gelijk aan $a$ geven we in hetgeen volgt aan met $\lceil a \rceil$). Bij iedere uit te proberen $X$ gaan we na of $X^2 - n$ een kwadraat is. Zodra dat zo is stoppen we, en hebben we de oplossing gevonden. ***In figuur 2*** is een en ander als een algoritme weergegeven.

Het 'stopgetal' $m$ is opgenomen om het algoritme niet langer te laten doorgaan dan gewenst. We laten in een voorbeeld zien hoe de methode werkt.

| Invoer: te ontbinden $n$, en een 'stopgetal' $m$<br>Uitvoer: factoren van $n$, of een melding dat die niet gevonden zijn | | |
|---|---|---|
| $X := \lceil \sqrt{n} \rceil$; $i := 0$ | | |
| Zolang $i < m$ Doe | | |
| | $Z := X^2 - n$ | |
| | Als $Z$ een kwadraat is: | |
| | Dan | $Y := \sqrt{Z}$; $p := X + Y$; $q := X - Y$<br>Druk af: $p$, $q$<br>Stop |
| | Anders | hoog $X$ en $i$ elk met 1 op, en ga door |
| Druk af: "Geen oplossing gevonden." | | |

figuur 2 Ontbinden in factoren volgens Fermat

***Voorbeeld***

Neem $n = 41567$; dan is $\sqrt{n} = 203{,}87\ldots$; dus beginnen we met $X = 204$.
Dan is $Z = 204^2 - 41567 = 49$, en dat is meteen al een kwadraat, namelijk van $Y = 7$.
We vinden als ontbinding $41567 = (204+7)(204-7) = 211 \times 197$.
Neem nu $n = 41561$, dan is $\sqrt{n} = 203{,}86\ldots$, dus beginnen we weer met $X = 204$.
Dan is $Z = 204^2 - 41561 = 57$, en dat is geen kwadraat. Dus proberen we achtereenvolgens $X = 205, 206, \ldots$ totdat $Z = X^2 - 41561$ wel een kwadraat is.
Dat blijkt pas bij $X = 219$ te gebeuren:
$Z = 219^2 - 41561 = 6400$, en dat is het kwadraat van $Y = 80$.
We vinden als ontbinding
$41561 = (219+80)(219-80) = 299 \times 139$.
(Merk op dat 299 geen priemgetal is; het is $13 \times 23$.)

De vraag is nu hoe goed dit algoritme van Fermat is. We willen achterhalen voor welke soorten sleutelparen de methode efficiënt werkt. We tellen daarom het aantal stappen, $i+1$, dat het algoritme heeft doorlopen als het een oplossing $p$, $q$ heeft gevonden. Bij die oplossing is $X = \frac{1}{2}(p+q)$, $Y = \frac{1}{2}(p-q)$, en $i = X - \lceil \sqrt{n} \rceil$, en natuurlijk ook $X^2 - Y^2 = n$. Nu is:

$$i = X - \lceil \sqrt{n} \rceil < X - \sqrt{n} = \frac{X^2 - n}{X + \sqrt{n}} = \frac{Y^2}{X + \sqrt{n}} < \frac{Y^2}{2\sqrt{n}} = \frac{(p-q)^2}{8\sqrt{n}}$$

Het aantal stappen is dus erg klein als de priemgetallen $p$ en $q$ dicht bij elkaar liggen. Om precies te zijn: bij een stopgetal $m$ lukt het om $n$ te ontbinden als:

$p - q < \sqrt{(8m)} \cdot n^{1/4}$

De priemgetallen $p$ en $q$ zijn ongeveer zo groot als $\sqrt{n}$, en hebben dus elk ongeveer half zoveel cijfers als $n$. Als nu hun verschil $p-q$ niet veel groter is dan $n^{1/4}$, dan hebben $p$ en $q$ dus bijna de bovenste helft van hun cijfers gemeenschappelijk. In zo'n geval kan het algoritme in een redelijk aantal stappen de ontbinding vinden. We geven een wat groter voorbeeld, van 64 bits.

***Voorbeeld***

$n = 16\,585\,512\,232\,168\,543\,399$; dan is $\lceil \sqrt{n} \rceil = 4\,072\,531\,429$;
$i = 0$: $X = 4\,072\,531\,429$;
$Z = X^2 - n = 8\,024\,238\,642$ is geen kwadraat;
$i = 1$: $X = 4\,072\,531\,430$;
$Z = X^2 - n = 16\,169\,301\,501$ is geen kwadraat;
$i = 2$: $X = 4\,072\,531\,431$;
$Z = X^2 - n = 24\,314\,364\,362$ is geen kwadraat;
$i = 3$: $X = 4\,072\,531\,432$;
$Z = X^2 - n = 32\,459\,427\,225 = 180\,165^2$ is een kwadraat. Dus:
$Y = 180\,165$, en
$p = X + Y = 4\,072\,711\,597$,
$q = X - Y = 4\,072\,351\,267$
Inderdaad komen de eerste 4 van de 10 cijfers (bijna de helft dus) van $p$ en $q$ overeen, en $(p-q)^2/(8\sqrt{n}) = 3{,}98\ldots$. Dat komt goed overeen met het feit dat we de ontbinding in 4 stappen hebben gevonden.

De conclusie van deze paragraaf is dat u bij het maken van een RSA-sleutelpaar niet de fout moet maken de twee priemgetallen te dicht bij elkaar te kiezen, maar op een afstand die flink groter is dan $n^{1/4}$. Anders levert dat een erg zwakke sleutel op.

**Verwijzingen**


[1] Ernst Lambeck e.a. (2008): *Geheim? Cryptografie en Getaltheorie.* Eindhoven: Regionaal Steunpunt Wiskunde D (zie *www.win.tue.nl/wiskunded*).

[2] Benne de Weger (2009): *Elementaire getaltheorie en asymmetrische cryptografie.* Utrecht: Epsilon Uitgaven (verschijnt voorjaar 2009).
Dit boek beoogt een toegankelijke tekst te zijn over (onder andere) RSA en de onderliggende getaltheorie. Bij het boek hoort een webpagina met een Java-applet, waarmee alle rekenbewerkingen waarvan in dit artikel sprake is, makkelijk uit-gevoerd kunnen worden, ook met grote getallen (zie *www.win.tue.nl/~bdeweger/MCR/*).



**Over de auteur**

Benne de Weger werkt als universitair docent cryptologie aan de Technische Universiteit Eindhoven.
E-mailadres: *b.m.m.d.weger@tue.nl*