Zwakke sleutels bij het RSA-cryptosysteem

DEEL 2

[Benne de Weger]

Wat vooraf ging

Dit is het tweede en laatste deel van een artikel over zwakke RSA-sleutelparen[1]. Zwakke sleutelparen zijn sleutelparen waarvan het geheime deel makkelijk te kraken is omdat bepaalde parameters op een onhandige manier gekozen zijn. In deel 1 is uitgelegd hoe en waarom het RSA-cryptosysteem werkt en is een klasse van zwakke sleutels aangewezen, gebaseerd op de factorisatiemethode van Fermat. De conclusie was dat een RSA-sleutelpaar zwak is als de priemfactoren van de modulus te dicht bij elkaar liggen.
In dit tweede deel bespreken we een andere klasse van zwakke sleutels, namelijk die waarbij de privé-exponent *te klein* gekozen is. In dit deel speelt het uitgebreide algoritme van Euclides weer een belangrijke rol. Voor het gemak van de lezer herhalen we hier dit algoritme, en vermelden we nog eens kort hoe een RSA-sleutelpaar er uitziet. Voor meer achtergrond en voorbeelden kunt u terecht in de paragrafen 2 en 3 van deel 1 van dit artikel[1].

Het uitgebreide algoritme van Euclides berekent de grootste gemene deler ggd(a, b) van de (gehele) getallen a en b, samen met de gehele coëfficiënten u en v zodat $d = ua + vb$; ***zie figuur 1*** op pagina 308.
Een RSA-sleutelpaar bestaat uit een publieke sleutel (n, e) en een privé-sleutel (n, d). Hierbij is de modulus n een groot getal dat is opgebouwd uit twee priemfactoren p en q, dus $n = pq$. De samenhang tussen de publieke en de privé-sleutel wordt gegeven door de relatie $ed \equiv 1 \pmod{\phi(n)}$ voor de publieke exponent e en de privé-exponent d. Hierbij is $\phi(n) = (p-1)(q-1)$.

7. De methode van Wiener voor het achterhalen van de privé-exponent

In paragraaf 6 van deel 1 zagen we een methode om zwakke sleutels voor RSA te identificeren, die uitsluitend gebruik maakt van de modulus. Nu bekijken we een andere methode, die zich niet in eerste instantie op de modulus richt, maar op de privé-exponent d. Als die niet goed gekozen is, is het sleutelpaar ook niet veilig. We laten zien dat dan de privé-exponent d te achterhalen is, en dat ook in dit geval de ontbinding van de modulus makkelijk berekend kan worden.
Het is aantrekkelijk voor bouwers van RSA-software om d klein te willen hebben. De snelheid van het ontsleutelen hangt namelijk sterk van de grootte van d af. Maar een te kleine waarde voor d is niet goed. Een heel kleine d is te raden door eenvoudig alle mogelijkheden af te gaan. In de praktijk betekent dit dat men $d > 2^{80}$ wil hebben, want dan is hij zelfs met heel veel rekenkracht niet in redelijke tijd te achterhalen door alle mogelijkheden uit te proberen. Dus d zal minstens 80 bits moeten hebben, maar men wil soms wel aanzienlijk minder dan s (vaak 1024) bits bereiken. We laten zien dat te kleine d (maar *wel* groter dan 80 bits) niet veilig is.
We roepen de relatie tussen d en e in herinnering:
$ed \equiv 1 \pmod{\phi(n)}$
Dus $ed = 1 + k \cdot \phi(n)$ voor een positieve gehele k. Ook k is (als het goed is) onbekend voor de kraker.
Michael Wiener heeft in 1990 laten zien hoe je uit deze relatie de breuk k/d kunt vinden als d niet al te groot is. De relatie laat zien dat k/d heel dicht bij $e/\phi(n)$ ligt, immers:

$$\frac{e}{\phi(n)} - \frac{k}{d} = \frac{ed - k\cdot\phi(n)}{d\cdot\phi(n)} = \frac{1}{d\cdot\phi(n)}$$

Het probleem (voor de kraker) is dat $\phi(n)$ niet bekend is. Wiener's idee maakt nu gebruik van het feit dat de onbekende $\phi(n)$ en de bekende n vrij dicht bij elkaar liggen. Immers:
$\phi(n) = (p-1)(q-1) = pq - p - q + 1 = n - (p + q - 1)$
waarbij p en q allebei van de grootteorde van $\sqrt{n}$ zijn. Dus $\phi(n)$ is n min iets dat veel kleiner is. Ongeveer de bovenste helft van de cijfers van $\phi(n)$ is gelijk aan die van n.
Nu volgt:
$ed - kn = (ed - k\cdot\phi(n)) + k\cdot(\phi(n) - n) = 1 - k(p + q - 1)$
en dus:

$$\frac{e}{n} - \frac{k}{d} = \frac{1 - k(p+q-1)}{nd}$$

Dit ligt echt dicht bij 0, zoals we zo dadelijk in detail zullen aantonen. Het interessante is dat e/n een volledig bekende breuk is, en k/d een nog onbekende breuk er vlakbij, maar met veel kleinere teller en noemer dan e/n. Een techniek om zulke breuken k/d te vinden bestaat; daarover zullen we zo ook iets zeggen.
Maar eerst de afschatting voor $e/n - k/d$.
Uit $2 < e < \phi(n)$ en $ed = 1 + k\cdot\phi(n)$ volgt:

$$k = \frac{ed-1}{\phi(n)} < \frac{ed}{\phi(n)} < d$$

Uit $ed - kn = 1 - k(p + q - 1)$ volgt dan:

$$\left|\frac{e}{n} - \frac{k}{d}\right| < \frac{k(p+q-1)}{nd} < \frac{p+q}{n}$$

Omdat p en q evenveel bits hebben, geldt:
$q < p < 2q$
Met $pq = n$ volgt nu dat:
$\sqrt{n} < p < \sqrt{(2n)}$
Uit $pq = n$ volgt $q = n/p$. Dus:
$p + q = p + n/p$.
De functie $f(p) = p + n/p$ heeft op het interval $[\sqrt{n}\,;\,\sqrt{(2n)}]$ haar maximum bij $p = \sqrt{(2n)}$; dus is:

$$f(p) \le \sqrt{(2n)} + \frac{n}{\sqrt{(2n)}} = (\sqrt{2} + \frac{1}{\sqrt{2}})\cdot\sqrt{n} < 2{,}13\sqrt{n}$$

Zodat $p + q < 2{,}13\sqrt{n}$.
Voor de afschatting van de twee breuken levert dit op:

$$\left|\frac{e}{n} - \frac{k}{d}\right| < \frac{p+q}{n} < \frac{2{,}13}{\sqrt{n}}$$

En dat is inderdaad heel klein.

Nu resteert de vraag hoe je bij een gegeven breuk A/B (alle) andere breuken a/b kunt vinden met veel kleinere teller en noemer, zodat $|A/B - a/b|$ heel klein is. De theorie van de *kettingbreuken* vertelt hoe je de breuken a/b met:

$$\left|\frac{A}{B} - \frac{a}{b}\right| < \frac{1}{2b^2}$$

kunt bepalen (zie het onderstaande voorbeeld). Om dit in ons geval te kunnen gebruiken wil je zeker weten dat:

$$\frac{2{,}13}{\sqrt{n}} < \frac{1}{2d^2}$$

Dat is zo als $d < 0{,}8\, n^{1/4}$.
Bij een veiligheidsparameter van $s = 1024$

betekent dit dat de privé-exponent d dus niet minder dan ongeveer $\frac{1}{4}s = 256$ bits mag hebben. En daarmee hebben we een nieuwe categorie zwakke sleutels van RSA opgespoord, namelijk die waarbij de privé-exponent d niet groter is dan ongeveer $n^{1/4}$.

Hoe werkt dat kettingbreukalgoritme? Dat is eigenlijk niets anders dan het uitgebreide algoritme van Euclides voor het bepalen van de grootste gemene deler van de teller A en de noemer B. We laten dit zien aan de hand van een eenvoudig voorbeeld.

Voorbeeld

Neem $A = 62$, $B = 23$. Het gehele deel van 62/23 is 2, en de rest is 16; dus:
$62/23 = 2 + 16/23$
Merk op dat we vinden dat 16 een lineaire combinatie van 62 en 23 is, namelijk
$16 = 62 - 2 \times 23$.
De laatste breuk 16/23 draaien we nu om, en we doen er hetzelfde mee: $23/16 = 1 + 7/16$, en $7 = 23 - 16$.
Dit laatste kunnen we terugrekenen tot een lineaire combinatie van de oorspronkelijke 62 en 23:
$7 = 23 - (62 - 2 \times 23) = -62 + 3 \times 23$
En opnieuw: de restbreuk 7/16 draaien we om: $16/7 = 2 + 2/7$, en:
$2 = 16 - 2 \times 7 = (62 - 2 \times 23) - 2 \times (-62 + 3 \times 23) = 3 \times 62 - 8 \times 23$
Enzovoorts.
We zetten de gevonden lineaire combinaties van 62 en 23 nog eens op een rijtje:

$$\begin{array}{rrcrcr} 16 = & 1 & \times 62 + & -2 & \times 23 \\ 7 = & -1 & \times 62 + & 3 & \times 23 \\ 2 = & 3 & \times 62 + & -8 & \times 23 \\ 1 = & -10 & \times 62 + & 27 & \times 23 \end{array}$$

De oorspronkelijke breuk 62/23 kunnen we aan de hand hiervan als volgt schrijven:

$$\frac{62}{23} = 2 + \frac{16}{23} = 2 + \frac{1}{\frac{23}{16}} = 2 + \frac{1}{1 + \frac{7}{16}} = 2 + \frac{1}{1 + \frac{1}{\frac{16}{7}}} = 2 + \frac{1}{1 + \frac{1}{2 + \frac{2}{7}}}$$

Zo kunnen we doorgaan. Met een handiger notatie – iedere keer '$+\frac{1}{\ldots}$' vervangen door een komma – wordt het:

$$\frac{62}{23} = [2, \tfrac{23}{16}] = [2, 1, \tfrac{16}{7}] = [2, 1, 2, \tfrac{7}{2}]$$
$$= [2, 1, 2, 3, \tfrac{1}{2}] = [2, 1, 2, 3, 2]$$

Hier houdt het op. Vervolgens laten we de 'staartstukken' die kleiner dan 1 zijn, telkens weg om benaderingsbreuken van 62/23 te krijgen:

$$\frac{62}{23} \approx [2] = \frac{2}{1}, \quad \frac{62}{23} \approx [2, 1] = \frac{3}{1},$$
$$\frac{62}{23} \approx [2, 1, 2] = \frac{8}{3}, \quad \frac{62}{23} \approx [2, 1, 2, 3] = \frac{27}{10}$$

De coëfficiënten van de lineaire combinaties (2,1), (3,1), (8,3), (27,10) komen precies weer terug in de benaderingsbreuken.

De stelling van Legendre uit de theorie van de kettingbreuken zegt nu dat iedere vereenvoudigde breuk a/b met $a < A$ en $b < B$ die voldoet aan:

$$\left|\frac{A}{B} - \frac{a}{b}\right| < \frac{1}{2b^2}$$

altijd te vinden zal zijn als benaderingsbreuk met het kettingbreukalgoritme. Het bewijs van deze stelling is wat lastiger. Voor meer over kettingbreuken kunt u het boek van Beukers[2] raadplegen.
We sluiten af met een voorbeeld met wat grotere getallen.

Voorbeeld

We nemen:
$n = 3\,100\,970\,273$ en $e = 1\,234\,322\,263$
De kettingbreuk e/n begint met:
$[0, 2, 1, 1, 19, 1, \ldots]$
Dat levert de volgende benaderingsbreuken op:
$\frac{1}{2}$, $\frac{1}{3}$, $\frac{2}{5}$, $\frac{39}{98}$, $\frac{41}{103}$, ...
Die breuken kunnen we zien als kandidaten voor k/d. Zo'n kandidaat testen we door te kijken of $ed - 1$ deelbaar is door k, en of de resulterende $R = (ed - 1)/k$ wel dicht genoeg bij n ligt om $\phi(n)$ te kunnen zijn:

k	d	R	conclusie
1	2	2 468 644 525	ligt te ver van n
1	3	3 702 966 788	ligt te ver van n
2	5	3 085 805 657	ligt te ver van n
39	98	120963581773/39	is niet eens geheel
41	103	3 100 858 368	dit lijkt een mooie kandidaat

De laatste kandidaat gaan we verder testen.
Als $\phi(n) = 3\,100\,858\,368$, dan is:
$p + q = n + 1 - \phi(n) = 111\,906$
En we weten natuurlijk ook dat:
$pq = n = 3\,100\,970\,273$
Twee getallen vinden waarvan som en product bekend zijn, doen we met de *abc*-formule. Dat levert:
$p = 61\,409$ en $q = 50\,497$
En inderdaad zijn deze geheel.

Hiermee hebben we dus niet alleen d gevonden maar meteen ook een ontbinding in factoren van n.

Terzijde. *We hebben zojuist in feite gezien dat het ontbinden van n makkelijk is als we, naast e, zowel d als k weten. Als we alleen d hebben, en niet k, dan is het ontbinden van n wat lastiger, maar het kan wel. Voor het ontsleutelen van geheimschriften is kennis van d (en natuurlijk n) voldoende.*

De conclusie van deze paragraaf is dat je bij het maken van een RSA-sleutelpaar ook niet de fout moet maken de privé-exponent d te klein te kiezen, zeker flink groter dan $n^{1/4}$, want anders levert ook dat een erg zwakke sleutel op.

8. Slotopmerkingen

In dit artikel hebben we laten zien dat er enkele categorieën zwakke sleutels zijn bij het RSA-cryptosysteem. We hebben ons daarbij beperkt tot die categorieën waarvan het met elementaire getaltheorie in te zien is waarom ze zwak zijn.
Er zijn wel meer zwakke sleutels bekend. Voor iedere N zijn er ongeveer $N/(\log N)^2$ RSA-moduli $n = pq$ die kleiner dan N zijn, en het is bekend dat er daarvan ten minste $N^{3/4}$ zwak zijn. Dat lijkt in absolute zin veel, maar relatief gezien is het slechts 1 op de $N^{1/4}/(\log N)^2$, dus nog altijd bijna niets.
Om nog een voorbeeld te noemen. Het hierboven behandelde resultaat van Wiener dat de sleutels met $d < n^{1/4}$ zwak zijn, is verbeterd (of, zo u wilt, verslechterd) tot $d < n^{0.292}$, en vermoed wordt dat alle $d < \sqrt{n}$ wel eens zwak zouden kunnen blijken. Recente bijdragen tot, en een overzicht van deze en uitgebreidere theorieën, waarbij ook gekeken is naar hoeveel informatie je over een sleutel prijs moet geven om hem helemaal te kunnen kraken, zijn te vinden in het proefschrift van Ellen Jochemsz[3].
Er zijn wel andere manieren om RSA aan te vallen. De beste nu bekende methoden om grote getallen in factoren te ontbinden zijn de getallenlichamenzeef (Number Field Sieve), en een methode gebaseerd op elliptische krommen. In feite zijn dat slimmere manieren om n te schrijven als verschil van twee kwadraten. Nederlandse wiskundigen als de broers Hendrik en Arjen Lenstra (hoogleraar in Leiden resp. Lausanne) hebben bij het ontwikkelen van deze methoden een belangrijke rol gespeeld.
RSA kan ook zonder ontbinden in factoren aangevallen worden. Er zijn allerlei aanvallen mogelijk die niet zozeer ingrijpen op de getaltheoretische achtergrond, maar op de manier waarop RSA gebruikt wordt of geïmplementeerd is in software of hardware.
RSA als zodanig is veilig en niet gekraakt. Maar om RSA echt veilig in te zetten moet er wel heel wat meer gebeuren dan het kiezen van grote priemgetallen. Dat hebben we in dit artikel willen illustreren aan de hand van twee voorbeelden.

Verwijzingen

[1] Het eerste deel van dit artikel staat in *Euclides* 84(7), pp. 256-260.
[2] Frits Beukers (1999): *Getaltheorie voor beginners*. Utrecht: Epsilon Uitgaven.

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_{nieuw} / d, en de rest r	
	Als de rest $r \neq 0$:	
	Dan	vervang d_{nieuw} door $(d_{\text{nieuw}} - qd)$; verwissel dan d en d_{nieuw} vervang u_{nieuw} door $(u_{\text{nieuw}} - qu)$; verwissel dan u en u_{nieuw} vervang v_{nieuw} door $(v_{\text{nieuw}} - qv)$; verwissel dan v en v_{nieuw} en ga door (met herhalen)
	Anders	Stop (de herhaling)
Druk af: d, u, v		

figuur 1 Uitgebreide algoritme van Euclides

Zie *www.epsilon-uitgaven.nl/E42.php* .

[3] Ellen Jochemsz (2007): *Cryptanalysis of RSA variants using small roots of polynomials*. Proefschrift, TU Eindhoven.
Zie *http://alexandria.tue.nl/extra2/200711750.pdf* .

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*