Primtal, faktorisering och RSA

Johan Håstad

KTH Numerical Analysis
and Computer Science

17 november, 2007

Ett Exempel

$$N = 93248941901237910481523319394135$$
$$41141253923482543847923483201340 94$$
$$30191341511661395185103412561530 23$$
$$23245252392306242109601232341201 56$$
$$80910410950130349861401286512 3$$

Kan vi avgöra om N är sammansatt?

Ett Exempel

$$N = \begin{aligned} &93248941901237910481523319394135 \\ &41141253923482543847923483201340 94 \\ &30191341511661395185103412561530 23 \\ &23245252392306242109601232341201 56 \\ &80910410950130349861401286 5123 \end{aligned}$$

Kan vi avgöra om N är sammansatt?

Om så kan vi faktorisera N?

Ett Exempel

$$N = 9324894190123791048152331939413541141412539234825438479234832013409430191341511661395185103412561530232324525239230624210960123234120156809104109501303498614012865123$$

Kan vi avgöra om N är sammansatt?

Om så kan vi faktorisera N?

Spelar det någon roll?

Ett Exempel

$$N = 9324894190123791048152331939413541141412539234825438479234832013409430191341511661395185103412561530232324525239230624210960123234120156809104109501303498614012865123$$

Kan vi avgöra om N är sammansatt? Ja, lätt.

Om så kan vi faktorisera N?

Spelar det någon roll?

Ett Exempel

$$N = \begin{aligned} &9324894190123791048152331939413 5 \\ &4114125392348254384792348320134094 \\ &3019134151166139518510341256153023 \\ &2324525239230624210960123234120156 \\ &8091041095013034986140128651 23 \end{aligned}$$

Kan vi avgöra om N är sammansatt? Ja, lätt.

Om så kan vi faktorisera N? Betydligt svårare.

Spelar det någon roll?

Ett Exempel

$$N = \begin{aligned} &93248941901237910481523319394135 \\ &41141253923482543847923483201340 94 \\ &30191341511661395185103412561530 23 \\ &23245252392306242109601232341201 56 \\ &80910410950130349861401286512 3 \end{aligned}$$

Kan vi avgöra om N är sammansatt? Ja, lätt.

Om så kan vi faktorisera N? Betydligt svårare.

Spelar det någon roll? Kunskap värt miljarder.

Plan

1. En hel del om primtal.
2. Lite om faktorisering.
3. Kryptosystemet RSA som finns på grund av våra förmåga att testa primtal och vår oförmåga att faktorisera.

Plan

1. En hel del om primtal.
2. Lite om faktorisering.
3. Kryptosystemet RSA som finns på grund av våra förmåga att testa primtal och vår oförmåga att faktorisera.

Det är små heltal i mina exempel men tänk på tal större än det på förra sidan.

Fermatprimtal

$F_i = 2^{2^i} + 1.$

$$
\begin{aligned}
F_0 &= 3 \\
F_1 &= 5 \\
F_2 &= 17 \\
F_3 &= 257 \\
F_4 &= 65537 \\
F_5 &= 4294967297
\end{aligned}
$$

Pierre de Fermat (amatörmatematiker från 1600-talet) noterade att det första 5 är primtal och påstods ha trott att alla F_i är primtal.

Vi vet idag

Fermat hade (helt) fel!

$F_5 = 4294967297 = 641 \cdot 6700417$ och inga andra kända tal i serien är primtal.

Borde Fermat insett att F_5 inte är primtal?

Fermats lilla sats

Sats: Om $1 \leq a \leq p-1$ så är resten av a^{p-1} vid division med p lika med 1.

$\equiv_p$ betyder rest vid divison med p

$$
\begin{aligned}
1^6 &= 1 \equiv_7 1 \\
2^6 &= 64 \equiv_7 1 \\
3^6 &= 729 \equiv_7 1 \\
4^6 &= 4096 \equiv_7 1 \\
5^6 &= 15625 \equiv_7 1 \\
6^6 &= 46656 \equiv_7 1
\end{aligned}
$$

Slutsatser av Fermats lilla sats

$2^8 = 256 \equiv_9 4$

$2^{10} = 1024 \equiv_{11} 1$

$4^{14} = 268435456 \equiv_{15} 1$

Slutsatser av Fermats lilla sats

$2^8 = 256 \equiv_9 4$ 9 definitivt inte primtal

$2^{10} = 1024 \equiv_{11} 1$

$4^{14} = 268435456 \equiv_{15} 1$

Slutsatser av Fermats lilla sats

$2^8 = 256 \equiv_9 4$ 9 definitivt inte primtal

$2^{10} = 1024 \equiv_{11} 1$ 11 är kanske primtal

$4^{14} = 268435456 \equiv_{15} 1$

Slutsatser av Fermats lilla sats

$2^8 = 256 \equiv_9 4$ 9 definitivt inte primtal

$2^{10} = 1024 \equiv_{11} 1$ 11 är kanske primtal

$4^{14} = 268435456 \equiv_{15} 1$ 15 är kanske primtal

Lite större tal

$N = 257$

$$3^{256} = 1390084523771447327649397867896613031142188508085291379916048244303607262976643594100176915410960952181166554054889943552 1$$

och ger rest 1 vid division med 256 vilket är rimligt då 257 är primtal.

Lite större tal

$N = 257$

$$3^{256} = 13900845237714473276493978678966130311421885080852913799160482443036072629766435941001769154109609521811665540548899435521$$

och ger rest 1 vid division med 256 vilket är rimligt då 257 är primtal.

Börjar bli tungt att räkna...

Åter till F_5

$3^{4294967296}$ ger rest 3029026160 vid division med $F_5 = 4294967297$ och vi kan med säkerhet säga att F_5 inte är primtal.

Hur inser man detta?

$3^{4294967296}$ är ett tal med ett par miljarder siffror...

Bättre räkningar

Vad är sista siffran i 1541^{2311}?

Bättre räkningar

Vad är sista siffran i 1541^{2311}? 1

Bättre räkningar

Vad är sista siffran i 1541^{2311}? 1

Vad är sista siffran i 1543^{2311}?

Bättre räkningar

Vad är sista siffran i 1541^{2311}? 1

Vad är sista siffran i 1543^{2311}? Samma som 3^{2311}

Bättre räkningar

Vad är sista siffran i 1541^{2311}? 1

Vad är sista siffran i 1543^{2311}? Samma som 3^{2311}

Sista siffran i 3-potenser är 3,9,7,1,3,9,7,1...

och svaret blir 7.

Generalisering

Sista siffran är resten vid division med 10.

För att beräkna den behöver vi bara sista siffran i operanderna och kan slänga alla andra siffor.

Generalisering

Sista siffran är resten vid division med 10.

För att beräkna den behöver vi bara sista siffran i operanderna och kan slänga alla andra siffor.

Allmänt: Vill vi veta resten av svaret av en beräkning vid division med N, kan vi kasta alla multipler av N på vägen.

Att räkna ut 3^{256} modulo 257

$$3^{256} = 9^{128}$$

Att räkna ut 3^{256} modulo 257

$$\begin{aligned} 3^{256} &= 9^{128} \\ &= 81^{64} \end{aligned}$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4} \\
&\equiv_{257} 241^{4}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4} \\
&\equiv_{257} 241^{4} \\
&= 58081^{2}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4} \\
&\equiv_{257} 241^{4} \\
&= 58081^{2} \\
&\equiv_{257} 256^{2}
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4} \\
&\equiv_{257} 241^{4} \\
&= 58081^{2} \\
&\equiv_{257} 256^{2} \\
&= 55536 \equiv_{257} 1
\end{aligned}
$$

Att räkna ut 3^{256} modulo 257

$$
\begin{aligned}
3^{256} &= 9^{128} \\
&= 81^{64} \\
&= 6561^{32} \\
&\equiv_{257} 136^{32} \\
&= 18496^{16} \\
&\equiv_{257} 249^{16} \\
&= 62001^{8} \\
&\equiv_{257} 64^{8} \\
&= 4096^{4} \\
&\equiv_{257} 241^{4} \\
&= 58081^{2} \\
&\equiv_{257} 256^{2} \\
&= 55536 \equiv_{257} 1
\end{aligned}
$$

Enbart 8 multiplikationer av tresiffriga tal.

Bättre beräkning F_5

Fermat kunde ha räknat ut resten av $3^{4294967296}$ vid division med 4294967297 med 32 multiplikationer av 10 siffriga tal och 32 resttagningar.

Tar högst någon timme för en van handräknare.

Bättre beräkning F_5

Fermat kunde ha räknat ut resten av $3^{4294967296}$ vid division med 4294967297 med 32 multiplikationer av 10 siffriga tal och 32 resttagningar.

Tar högst någon timme för en van handräknare.

Skulle dock inte ha gett faktoriseringen som kräver betydligt mer jobb att få fram för hand.

Datorkörningar

En dator klarar att göra ett "Fermattest" på 1000 siffriga tal på högst någon minut, en bra implementation på någon sekund.

Slutsats antingen Definitivt inte primtal eller Kanske primtal.

Bättre upp

Gary Miller och Michael Rabin visade 1976 att om vi utvidgar testet en aning och provar 50 slumpvis a så fås

Definitivt inte primtal eller Troligen primtal.

Sannolikhet att ha fel är 2^{-100}.

Behövs slump för primtalstest?

Felsannolikhet 2^{-100} gör detta till en akademisk fråga av litet praktiskt intresse.

Stort filosofiskt och teoretiskt intresse. Finns det ett effektivt deterministiskt primtalstest?

Agarwal, Saxena, Kayal, 2002

Det finns ett deterministiskt primtalstest som på tal med n siffor gör $\approx n^6$ operationer.

Mycket bättre än provdivision som går i tid $\sqrt{N} \approx 10^{n/2}$, men mycket sämre än Miller-Rabin som går i tid $\approx n^3$.

Hur svårt är faktorisering?

Bästa datorprogrammen klarar ca 200 decimala siffror om man kör på hundratals datorer under ett par månader.

"Troligen" mycket svårare än primtalstest.

Inte i närheten av ett matematiskt bevis att faktorisering verkligen är svårare.

Kryptering baserat på dessa idéer

Betrakta följande:

Fermats lilla sats ger att a^{21} och a ger samma rest vid divison med 11.

Kryptering baserat på dessa idéer

Betrakta följande:

Fermats lilla sats ger att a^{21} och a ger samma rest vid divison med 11.

Sätt b till resten av a^7 vid divison med 11.

Kryptering baserat på dessa idéer

Betrakta följande:

Fermats lilla sats ger att a^{21} och a ger samma rest vid divison med 11.

Sätt b till resten av a^7 vid divison med 11.

Då $21 = 7 \cdot 3$ ger resten av b^3 vid division med 11 tillbaka a.

$a = 2$ ger $a^7 = 128 \equiv_{11} 7 = b$ och $b^3 = 343 \equiv_{11} 2$.

Kryptering baserat på dessa idéer

Betrakta följande:

Fermats lilla sats ger att a^{21} och a ger samma rest vid divison med 11.

Sätt b till resten av a^7 vid divison med 11.

Då $21 = 7 \cdot 3$ ger resten av b^3 vid division med 11 tillbaka a.

$a = 2$ ger $a^7 = 128 \equiv_{11} 7 = b$ och $b^3 = 343 \equiv_{11} 2$.

Vi kan återskapa förvanskad information, kryptering och dekryptering.

RSA Kryptering

Välj stort tal N och e och d så att a^{ed} och a alltid ger samma rest vid division med N.

Ta ett medelande och koda som ett stort heltal M.

Kryptotexten blir resten av M^e vid division med N, kalla den C.

Klartexten återskapas som resten av C^d med division med N.

Exempel

Meddelande HEJ JOHAN. A=1, B=2 etc ger

$$M = 805101015080114$$

Vi kan ha $N = 23942194232123139$ och lämpliga e och d.

Längre meddelanden delas upp i delar, ett blockkrypto.

RSA efter upptäcktarna av Ron Rivest, Adi Shamir och Len Adleman.

Egenskaper hos RSA

Kryptering och dekryptering är tämligen effektiva, ungefär som ett Miller-Rabin test.

Välj N som produkten av två stora (kanske 150 decimala siffror) primtal.

Beräkna lämpliga e och d. Kräver eftertanke men är blixtsnabbt.

Publicera N och e.

Egenskaper hos RSA

Kryptering och dekryptering är tämligen effektiva, ungefär som ett Miller-Rabin test.

Välj N som produkten av två stora (kanske 150 decimala siffror) primtal.

Beräkna lämpliga e och d. Kräver eftertanke men är blixtsnabbt.

Publicera N och e.

Vi publicerar KRYPTERINGSNYCKELN, alla kan kryptera, även de vi inte träffat.

Public key cryptography

Öppen nyckel ger fantastiska möjligheter.

Alla publicerar sina nycklar på sina hemsidor.

Vi kan kommunicera med alla utan att nyfikna hackers eller myndigheter kan snoka.

RSA egenskaper

Lätt att skapa nycklar via primtalstest.

Enda kända sättet att forcera är att faktorisera N.

Lätt att se att N inte är primtal, svårt att faktorisera.

Kan eventuellt använda $e = 3$ men $e = 65537$ är bättre.

Slutord

Inte svårt att skriva egen implementation av RSA för realistiskt stora tal. Kräver bara aritmetik på stora heltal.

Rimligt projekt att förstå talteorin bakom att RSA fungerar.

Matematiken ligger till grund för stora investeringar och samhällets säkerhet.