RSA, integer factorization, record computations

Inria Nancy, France

Summer school CIMPA, Douala, Cameroon, July 2024

These slides at https://people.rennes.inria.fr/Aurore.Guillevic/talks/2024-07-Douala/24-07-Douala-RSA.pdf

Outline

Outline

Introduction: public-key cryptography

Introduced in 1976 (Diffie–Hellman, DH) and 1977 (Rivest–Shamir–Adleman, RSA)
Asymmetric means distinct public and private keys

- encryption with a public key
- decryption with a private key
- deducing the private key from the public key is a very hard problem

Two hard problems:

- Integer factorization (for RSA)
- Discrete logarithm computation in a finite group (for Diffie–Hellman)

Public-key encryption

Alice **Bob**

Public-key encryption

Alice

Bob

public key PK_A

secret key sk_A

Public-key encryption

Alice **Bob**

public key PK_A

secret key sk_A

PK_A →

Public-key encryption

Alice

public key PK_A
secret key sk_A

PK_A →

Bob

1. gets Alice's public key PK_A
2. encrypts $\mathcal{M}$ with PK_A
3. sends $C = \mathsf{Enc}_{\mathsf{PK}_A}(\mathcal{M})$ to Alice

Public-key encryption

Alice

public key PK_A

secret key sk_A

Bob

PK_A →

1. gets Alice's public key PK_A
2. encrypts $\mathcal{M}$ with PK_A
3. sends $C = \mathsf{Enc}_{\mathsf{PK}_A}(\mathcal{M})$ to Alice

← C

Public-key encryption

Alice

public key PK_A
secret key sk_A

PK_A →

Bob

1. gets Alice's public key PK_A
2. encrypts $\mathcal{M}$ with PK_A
3. sends $C = \mathsf{Enc}_{\mathsf{PK}_A}(\mathcal{M})$ to Alice

← C

4. gets C from Bob
5. computes $\mathsf{Dec}_{\mathsf{sk}_A}(C) = \mathcal{M}$

RSA Public-key encryption

Alice **Bob**

secret primes p, q, $\varphi(N) = (p-1)(q-1)$
public modulus $N = pq$, encryption exponent $e = 3$ or $2^{16} + 1$
secret decryption exponent $d = 1/e \bmod (p-1)(q-1)$
so that $e \cdot d = 1 \bmod (p-1)(q-1)$
and $x^{ed} = x \bmod N$

RSA Public-key encryption

Alice **Bob**

secret primes p, q, $\varphi(N) = (p-1)(q-1)$

public modulus $N = pq$, encryption exponent $e = 3$ or $2^{16}+1$

secret decryption exponent $d = 1/e \bmod (p-1)(q-1)$

so that $e \cdot d = 1 \bmod (p-1)(q-1)$

and $x^{ed} = x \bmod N$

N, e →

RSA Public-key encryption

Alice | **Bob**

secret primes p, q, $\varphi(N) = (p-1)(q-1)$
public modulus $N = pq$, encryption exponent $e = 3$ or $2^{16} + 1$
secret decryption exponent $d = 1/e \bmod (p-1)(q-1)$
so that $e \cdot d = 1 \bmod (p-1)(q-1)$
and $x^{ed} = x \bmod N$

$\xrightarrow{N, e}$

gets Alice's public key N, e
encrypts $\mathcal{M}$ as $C = m^e \bmod N$
sends C to Alice

RSA Public-key encryption

Alice | **Bob**

secret primes p, q, $\varphi(N) = (p-1)(q-1)$
public modulus $N = pq$, encryption exponent $e = 3$ or $2^{16}+1$
secret decryption exponent $d = 1/e \bmod (p-1)(q-1)$
so that $e \cdot d = 1 \bmod (p-1)(q-1)$
and $x^{ed} = x \bmod N$

N, e →

gets Alice's public key N, e
encrypts $\mathcal{M}$ as $C = m^e \bmod N$
sends C to Alice

← C

RSA Public-key encryption

Alice		**Bob**
secret primes p, q, $\varphi(N) = (p-1)(q-1)$ public modulus $N = pq$, encryption exponent $e = 3$ or $2^{16}+1$ secret decryption exponent $d = 1/e \bmod (p-1)(q-1)$ so that $e \cdot d = 1 \bmod (p-1)(q-1)$ and $x^{ed} = x \bmod N$		
	N, e →	
		gets Alice's public key N, e encrypts $\mathcal{M}$ as $C = m^e \bmod N$ sends C to Alice
	← C	
gets C from Bob computes $C^d \bmod N = \mathcal{M}$		

RSA Public-key encryption, toy example

Alice **Bob**

secret primes $p = 11$, $q = 17$

public key modulus $N = 11 \cdot 17 = 187$, exponent $e = 3$

$(11 - 1)(17 - 1) = 160$, $d = 1/3 \bmod 160 = 107$

so that $3 \cdot 107 = 321 = 1 \bmod 160$

and $x^{3 \cdot 107} = x \bmod N$

RSA Public-key encryption, toy example

Alice **Bob**

secret primes $p = 11, q = 17$

public key modulus $N = 11 \cdot 17 = 187$, exponent $e = 3$

$(11 - 1)(17 - 1) = 160$, $d = 1/3 \bmod 160 = 107$

so that $3 \cdot 107 = 321 = 1 \bmod 160$

and $x^{3 \cdot 107} = x \bmod N$

$\xrightarrow{187,\ 3}$

RSA Public-key encryption, toy example

Alice | **Bob**

secret primes $p = 11, q = 17$

public key modulus $N = 11 \cdot 17 = 187$, exponent $e = 3$

$(11 - 1)(17 - 1) = 160$, $d = 1/3 \bmod 160 = 107$

so that $3 \cdot 107 = 321 = 1 \bmod 160$

and $x^{3 \cdot 107} = x \bmod N$

$\xrightarrow{187,\ 3}$

gets Alice's public key $187,\ 3$

encrypts $\mathcal{M} = 38$ as $C = 38^3 \bmod 187 = 81$

sends $C = 81$ to Alice

RSA Public-key encryption, toy example

Alice		Bob
secret primes $p = 11, q = 17$		
public key modulus $N = 11 \cdot 17 = 187$, exponent $e = 3$		
$(11-1)(17-1) = 160$, $d = 1/3 \bmod 160 = 107$		
so that $3 \cdot 107 = 321 = 1 \bmod 160$		
and $x^{3 \cdot 107} = x \bmod N$		
	$187,\ 3$ →	
		gets Alice's public key $187,\ 3$
		encrypts $\mathcal{M} = 38$ as $C = 38^3 \bmod 187 = 81$
		sends $C = 81$ to Alice
	← $C = 81$	

RSA Public-key encryption, toy example

Alice | **Bob**

secret primes $p = 11, q = 17$
public key modulus $N = 11 \cdot 17 = 187$, exponent $e = 3$
$(11 - 1)(17 - 1) = 160$, $d = 1/3 \bmod 160 = 107$
so that $3 \cdot 107 = 321 = 1 \bmod 160$
and $x^{3 \cdot 107} = x \bmod N$

Alice → Bob: $187,\ 3$

Bob:
gets Alice's public key $187,\ 3$
encrypts $\mathcal{M} = 38$ as $C = 38^3 \bmod 187 = 81$
sends $C = 81$ to Alice

Bob → Alice: $C = 81$

Alice:
gets $C = 81$ from Bob
computes $81^{107} \bmod 187 = 38 = \mathcal{M}$

RSA, how does it work?

1977, Rivest, Shamir, Adleman

- modulus $N = p \times q$, p, q two distinct large primes
- arithmetic modulo N, in $\mathbb{Z}/N\mathbb{Z} = \{0, 1, \ldots, N-1\}$

The **multiplicative group** is the set of **invertible** integers in $\{1, 2, \ldots, N-1\}$.
invertible x means $gcd(x, N) = 1$, x coprime to N.
There are $\varphi(N) = (p-1)(q-1)$ invertible integers in $\{1, \ldots, N-1\}$

Hard tasks without knowing p, q if N is large enough:

- computing $(p-1)(q-1)$,
- computing a square root $\sqrt{x} = x^{1/2} \bmod N$,
- computing an e-th root $x^{1/e} \bmod N$.

RSA, how does it work?

The security relies on the hardness of computing d from N, e.
p, q are required to compute $\varphi(N)$
$\rightarrow$ security relies on the hardness of **integer factorization**.
Usecases:
`ssh-keygen` (linux), SSL-TLS, payment (chip) cards, PGP: Enigmails on Thunderbird, Protonmail.
Note that short keys are not allowed:

```
ssh-keygen -b 512 -t rsa
Invalid RSA key length: minimum is 1024 bits
```

Weakness on exponents

For faster encryption, one can choose a short public exponent e (coprime to N).
Two common choices of *prime* exponents:

- $e = 3$
- $e = 2^{16} + 1 = 65537$ (safer choice)

Old known facts/attacks:

- Knowing both the public and private exponents e, d gives a factorization of N
- Short private exponent is a bad idea
  - faster decryption (at the cost of larger e, slower encryption), but
  - Wiener attack
  - Idea: continued fraction technique.

Padding messages

$m \in \{0, 1, 2, \ldots, N-1\}$. Problems:

- $m = 0 \implies c = m^e = 0 \bmod N$
- $m = 1 \implies c = m^e = 1 \bmod N$
- $2 \leq m \leq \lfloor \sqrt[e]{N} \rfloor \implies c = m^e$ (no modular reduction) $\implies m = c^{1/e}$ as an integer.

Standards (PKCS) define ways to fill the zeros (the unused bytes) between m and N.

Padding messages

$m \in \{0, 1, 2, \ldots, N-1\}$. Problems:

- $m = 0 \implies c = m^e = 0 \bmod N$
- $m = 1 \implies c = m^e = 1 \bmod N$
- $2 \leq m \leq \lfloor \sqrt[e]{N} \rfloor \implies c = m^e$ (no modular reduction) $\implies m = c^{1/e}$ as an integer.

Standards (PKCS) define ways to fill the zeros (the unused bytes) between m and N.

Malleability

$$\begin{cases} c_1 = m_1^e \bmod N \\ c_2 = m_2^e \bmod N \end{cases} \implies c_1 \cdot c_2 \bmod N = (m_1 \cdot m_2)^e \bmod N$$

We don't want this property $\rightarrow$ padding

Knowing the public and private exponents e, d gives a factorization of N

- if x is a square mod N, it has 4 square roots y such that $y^2 = x \bmod N$
- $ed = 1 \bmod (p-1)(q-1) \iff ed - 1 = 0 \bmod (p-1)(q-1)$
- For all $x \in \{1, \ldots, N-1\}$ coprime to N, $x^{ed-1} \equiv 1 \bmod N$
- $ed - 1$ is even: $(ed-1)/2$ is an integer

If N, e and d are known:
Take at random $x \leftarrow \{2, \ldots, N-2\}$, compute $y = x^{(ed-1)/2} \bmod N$ a square root of 1.
If $y \neq \pm 1$, then

$$y^2 \equiv 1 \bmod N \iff y^2 - 1 = (y-1)(y+1) \equiv 0 \bmod N$$

$\rightarrow$ compute $\gcd(y-1, N)$ or $\gcd(y+1, N)$ to find p or q.

If y is 1, try with $(ed-1)/4, \ldots, (ed-1)/2^i$ as long as it is an integer.
Otherwise, try with another x. Success rate is high.

Example

$N = 43 \times 47 = 2021$, $e = 5$ coprime to $\varphi(N) = 42 \times 46 = 1932$,
$d = 1/e \bmod \varphi(N) = 773$

```
p = 43 ; q = 47 ; N = p * q
e = 5
phiN = (p-1) * (q-1)
g, d, v = xgcd(e, phiN) # d is the private exponent
y = 1; x = 2
while y == 1:
    expo = e*d - 1
    while y == 1 and (expo % 2) == 0:
        expo = expo // 2
        y = x**expo % N
    if y == 1:
        x = x+1
gcd(y-1, N); gcd(y+1, N)
```

We obtain: $2^{1932/4} = 988 \bmod N$, $\gcd(y-1, N) = 47 = q$, $\gcd(y+1, N) = 43 = p$.

Choosing key sizes

Symmetric ciphers (AES): key sizes are 128, 192 or 256 bits.
Perfect symmetric cipher: trying all keys of size n bits takes 2^n tests
$\rightarrow$ **brute-force search**

perfect symmetric cipher with secret key in $[0, 2^n - 1]$, of n bits $\leftrightarrow$ n bits of security

For RSA with N of length(N) bits:
n bits of security $\leftrightarrow$ the best (mathematical) attack should take at least 2^n steps

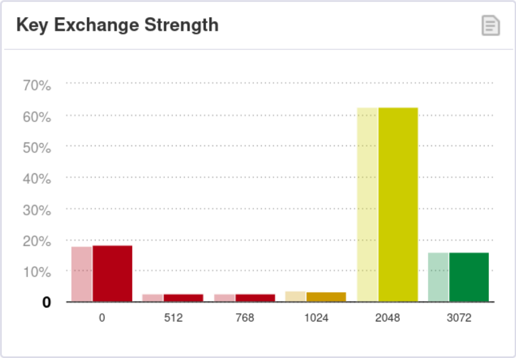
- what is the fastest attack?
- how much time does it take with respect to length(N)?

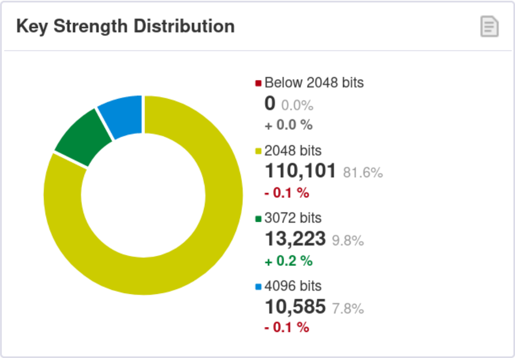
RSA keys are much larger.
Cipher suite: a pair of symmetric and asymmetric ciphers offering the same level of security.

Examples

https://www.lemonde.fr/, https, security information →
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, 128 bits, TLS 1.2

Qualys. SSL Labs https://www.ssllabs.com/ssl-pulse/





Particles

n	2^n	Examples
32	$2^{32} = 10^{9.6}$	number of humans on Earth
47	$2^{47} = 10^{14.2}$	distance Earth - Sun in millimeters ($149.6 \cdot 10^{12}$) number of operations in one day on a processor at 2 GHz
56	$2^{55.8} = 10^{16.8}$	number of operations in one year on a processor at 2 GHz
79	$2^{79} = 10^{23.8}$	Avogadro number: atoms of Carbon 12 in 1 mol
82	$2^{82.3} = 10^{24.8}$	mass of Earth in kilogrammes
100	$2^{100} = 10^{30}$	number of operations in $13.77 \cdot 10^9$ years (age of the universe) on a processor at 2 GHz
155	$2^{155} = 10^{46.7}$	number of molecules of water on Earth
256	$2^{256} = 10^{77.1}$	number of electrons in universe

Courtesy Marine Minier

Boiling water

Universal Security; From bits and mips to pools, lakes – and beyond
Arjen Lenstra, Thorsten Kleinjung, and Emmanuel Thomé
https://hal.inria.fr/hal-00925622

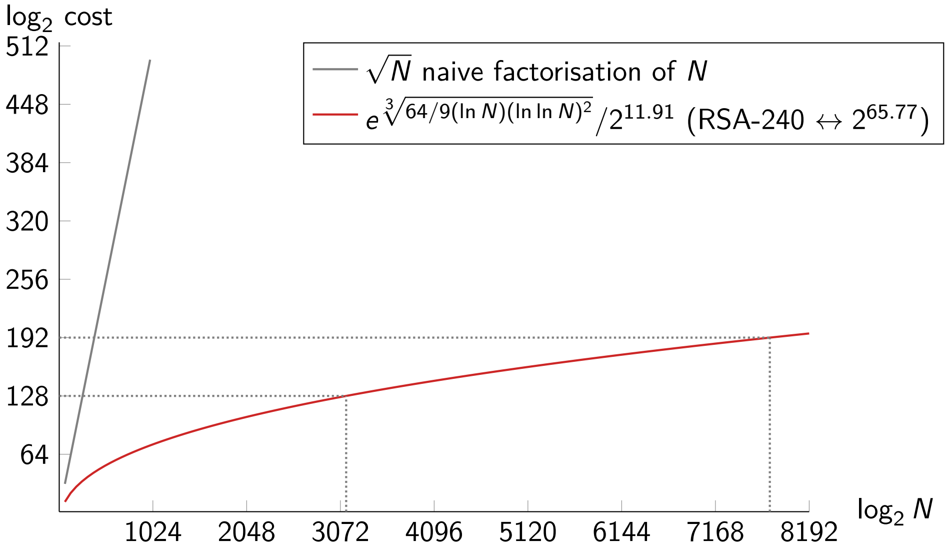
- 2^{90} operations require enough energy to boil the lake of Genève
- 2^{114} operations: boiling all the water on Earth
- 2^{128} operations: boiling 16,000 planets like the Earth

Choosing key sizes

For RSA with N of length(N) bits:
n bits of security $\leftrightarrow$ the best (mathematical) attack should take at least 2^n steps

- fastest factorization: with the Number Field Sieve algorithm
- Complexity: $\exp\left(\sqrt[3]{(64/9+o(1))(\ln N)(\ln\ln N)^2}\right)$
- $+o(1)$ not known
- $\exp\left(\sqrt[3]{(64/9+0)(\ln N_{\text{RSA-240}})(\ln\ln N_{\text{RSA-240}})^2}\right) = 2^{77.68}$
- RSA-240 in $2^{65.77}$ operations $\rightarrow 2^{65.77}/2^{77.68} = 2^{-11.91}$

Replace unknown $+o(1)$ in the exp() by a global scaling factor $2^{-11.91} \cdot \exp()$
(A. Lenstra, Verheul, Asiacrypt'01)



RSA-240: 953 core-years, Intel Xeon Gold 6130 CPUs as a reference (2.1GHz)
$\approx 953 \cdot 365.25 \cdot 24 \cdot 60 \cdot 60 \cdot 2.1 \cdot 10^9 \approx 2^{65.77}$

Outline

Naive way 1: Testing all primes up to square root of N

Trial division: testing all the primes up to $\sqrt{N}$
But if there are too many primes to test, it never ends

- $x/\ln x$ prime numbers between 1 and x (with $\ln\exp(1) = 1$)
- $\sqrt{N}/\ln\sqrt{N}$ prime numbers between 1 and $\sqrt{N}$

N (bits)	N (digits)	$\sqrt{N}/\ln\sqrt{N}$	
256	77	2^{122}	10^{37}
512	154	2^{249}	10^{75}
768	231	2^{376}	10^{114}
1024	308	2^{504}	10^{152}
1280	385	2^{632}	10^{191}
1536	462	2^{759}	10^{229}
1792	539	2^{887}	10^{267}
2048	617	2^{1015}	10^{306}

Naive way 2: testing all primes around square root of N

If p and q are of the same length (in bits), test all prime factors between $\lfloor\sqrt{N}/2\rfloor$ and $\lfloor\sqrt{N}\rfloor$.
How many primes in $[1, 2^n]$? approximately $2^n/\ln 2^n$
How many primes in $[2^{n-1}, 2^n]$? approximately $(1/2) \times 2^n/\ln 2^n$

Still completely impracticable.
(Trial division usually to detect prime factors up to 10^6 (78498 distinct prime factors, $10^6/\ln 10^6 = 72382.4$) or 10^7 (664579 distinct prime factors, $10^7/\ln 10^7 = 620420.7$))

Historical steps in integer factorization

- 1975, Morrison, Brillhard, continued fraction method CFRAC (factorization of $2^{2^7}+1 = 2^{128}+1$) (see the *Cunningham project* https://homes.cerias.purdue.edu/~ssw/cun/)
  $2^{128}+1 = 340282366920938463463374607431768211457 = 59649589127497217 \times 5704689200685129054721$
- 1981, Dixon, random squares method
- 70's, unpublished: Schroeppel, Linear Sieve
- 1982, Pomerance, Quadratic Sieve
- 1987, Lenstra, Elliptic Curve Method (ECM)
- 1993, Buhler, Lenstra, Pomerance, General Number Field Sieve

Strong joint work of researchers and manufacturers of computers in the US (*before* the Personal Computer)

Square roots modulo N

In $\mathbb{R}$ or $\mathbb{C}$, if x is a square, it has two square roots $\sqrt{x}$ and $-\sqrt{x}$.
But in $\mathbb{Z}/N\mathbb{Z}$ with $N = pq$ strange things happen: **four** square roots.

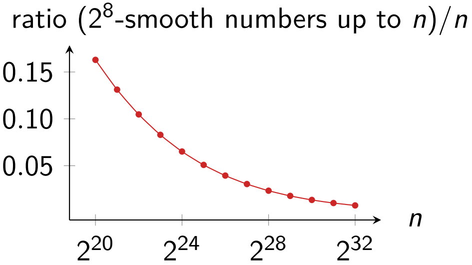
```
N = 2021
for i in range(-N//2, N//2):
    if (i**2 % N) == 1:
        print(i)
```

Two pairs of square roots of $x = 1$: $(1, -1)$ and $(-988, 988)$

$$
\begin{aligned}
& 988^2 = 1^2 \bmod 2021 \\
\iff & 988^2 - 1^2 = 0 \bmod 2021 \\
\iff & (988 - 1) \times (988 + 1) = 0 \bmod 2021
\end{aligned}
$$

Compute a gcd (greatest common divisor):
$\gcd(988 - 1, 2021) = 47$, $\gcd(988 + 1, 2021) = 43$.
$N = 43 \times 47$

Smooth numbers



B-smooth

A positive integer n is B-smooth $\iff$ n factors as a product of primes up to B
$n = 2^{e_1} 3^{e_2} 5^{e_3} \cdots p_i^{e_i}$ and $p_i \le B$.

B-smooth integers are quite common:
10% of 22-bit integers are 8-bit smooth
5% of 25-bit integers are 8-bit smooth
1% of 31-bit integers are 8-bit smooth

32-bit $a = 2654809430$
$= 2 \cdot 5 \cdot 7 \cdot 13 \cdot 59 \cdot 197 \cdot 251$
is 8-bit smooth ($B = 256$)

For very large integers:
Proba(n is B-smooth) = Dickman-$\rho(\log n / \log B)$

Factorization with the Quadratic Sieve

N to be factored

If $X^2 \equiv Y^2 \bmod N$ and $X \not\equiv \pm Y \bmod N$, then $\gcd(X \pm Y, N)$ gives a factor of N.
Find such X, Y.

Factorization with the Quadratic Sieve

N to be factored

If $X^2 \equiv Y^2 \bmod N$ and $X \neq \pm Y \bmod N$, then $\gcd(X \pm Y, N)$ gives a factor of N.

Find such X, Y.

Set $m = \lfloor \sqrt{N} \rfloor$, set bounds A, B

Factorization with the Quadratic Sieve

N to be factored

If $X^2 \equiv Y^2 \bmod N$ and $X \neq \pm Y \bmod N$, then $\gcd(X \pm Y, N)$ gives a factor of N.

Find such X, Y.

Set $m = \lfloor \sqrt{N} \rfloor$, set bounds A, B

For many small $a \leq A$, computes $n_a = (a + m)^2 - N$

if n_a is B-smooth, store the relation $n_a = p_1^{e_1} p_2^{e_2} \cdots p_j^{e_j}$ with all primes $p_i \leq B$

Factorization with the Quadratic Sieve

N to be factored

If $X^2 \equiv Y^2 \bmod N$ and $X \neq \pm Y \bmod N$, then $\gcd(X \pm Y, N)$ gives a factor of N.

Find such X, Y.

Set $m = \lfloor \sqrt{N} \rfloor$, set bounds A, B

For many small $a \leq A$, computes $n_a = (a+m)^2 - N$

if n_a is B-smooth, store the relation $n_a = p_1^{e_1} p_2^{e_2} \cdots p_j^{e_j}$ with all primes $p_i \leq B$ Find a combination s.t. $n_{a_1} n_{a_2} \cdots n_{a_i} = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ and all e_i even

$X = (a_1 + m)(a_2 + m) \cdots (a_i + m) \bmod N$, $Y = \sqrt{n_{a_1} n_{a_2} \cdots n_{a_i}} \bmod N$

If $X \neq \pm Y \bmod N$, computes $\gcd(X - Y, N)$.

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor \sqrt{N} \rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor \sqrt{N} \rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

is $n = (a + m)^2 - N$ smooth for small a?

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor \sqrt{N} \rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

is $n = (a+m)^2 - N$ smooth for small a?

$$\begin{array}{rl} (2+m)^2 - N = & 95 = 5 \cdot 19 \\ (5+m)^2 - N = & 380 = 2^2 \cdot 5 \cdot 19 \\ (17+m)^2 - N = & 1700 = 2^2 \cdot 5^2 \cdot 17 \end{array} \rightarrow \begin{array}{c} \begin{array}{cccc} 2 & 5 & 17 & 19 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 2 & 1 & 0 & 1 \\ 2 & 2 & 1 & 0 \end{bmatrix} \end{array} \text{ exponents}$$

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor \sqrt{N} \rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

is $n = (a+m)^2 - N$ smooth for small a?

$$\begin{array}{rcl} (2+m)^2 - N = & 95 & = 5 \cdot 19 \\ (5+m)^2 - N = & 380 & = 2^2 \cdot 5 \cdot 19 \\ (17+m)^2 - N = & 1700 & = 2^2 \cdot 5^2 \cdot 17 \end{array} \rightarrow \begin{array}{c} \begin{matrix} 2 & 5 & 17 & 19 \end{matrix} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \text{ exponents mod 2}$$

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor\sqrt{N}\rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \le B$ primes

is $n = (a+m)^2 - N$ smooth for small a?

$$\begin{array}{rl} \to (2+m)^2 - N = & 95 = 5 \cdot 19 \\ \to (5+m)^2 - N = & 380 = 2^2 \cdot 5 \cdot 19 \\ (17+m)^2 - N = & 1700 = 2^2 \cdot 5^2 \cdot 17 \end{array} \to \begin{array}{c} \begin{matrix} 2 & 5 & 17 & 19 \end{matrix} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \text{exponents mod 2}$$

Left kernel: $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

$$(2+m)^2(5+m)^2 \equiv 2^2 \cdot 5^2 \cdot 19^2 \bmod N$$

Factorization with the Quadratic Sieve: example

$N = 2021$, $m = \lfloor \sqrt{N} \rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

is $n = (a + m)^2 - N$ smooth for small a?

$$\begin{aligned} (2+m)^2 - N &= 95 = 5 \cdot 19 \\ (5+m)^2 - N &= 380 = 2^2 \cdot 5 \cdot 19 \\ (17+m)^2 - N &= 1700 = 2^2 \cdot 5^2 \cdot 17 \end{aligned} \rightarrow \begin{array}{cccc} 2 & 5 & 17 & 19 \end{array} \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{ exponents mod 2}$$

Left kernel: $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

$$\begin{aligned} (2+m)^2(5+m)^2 &\equiv 2^2 \cdot 5^2 \cdot 19^2 \bmod N \\ (\underbrace{46 \cdot 49}_{X})^2 &\equiv (\underbrace{2 \cdot 5 \cdot 19}_{Y})^2 \bmod N \end{aligned}$$

Factorization with the Quadratic Sieve: example

$N = 2021,\ m = \lfloor\sqrt{N}\rfloor = 44$

Smoothness bound $B = 19$

$\mathcal{F} = \{2, 3, 5, 7, 11, 13, 17, 19\}$ small primes up to B, $i = \#\mathcal{F} = 8$

B-smooth integer: $n = p_1^{e_1} p_2^{e_2} \cdots p_i^{e_i}$, all $p_i \leq B$ primes

is $n = (a+m)^2 - N$ smooth for small a?

$$\begin{aligned}(2+m)^2 - N &= 95 = 5 \cdot 19\\ (5+m)^2 - N &= 380 = 2^2 \cdot 5 \cdot 19\\ (17+m)^2 - N &= 1700 = 2^2 \cdot 5^2 \cdot 17\end{aligned} \rightarrow \begin{array}{c} \begin{matrix} 2 & 5 & 17 & 19 \end{matrix} \\ \begin{bmatrix} 0 & 1 & 0 & 1\\ 0 & 1 & 0 & 1\\ 0 & 0 & 1 & 0 \end{bmatrix}\end{array} \text{ exponents mod 2}$$

Left kernel: $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

$$\begin{aligned}(2+m)^2(5+m)^2 &\equiv 2^2 \cdot 5^2 \cdot 19^2 \bmod N\\ (\underbrace{46 \cdot 49}_{X})^2 &\equiv (\underbrace{2 \cdot 5 \cdot 19}_{Y})^2 \bmod N\end{aligned}$$

$X = 2254 \equiv 233 \bmod N,\ Y = 190 \bmod N$

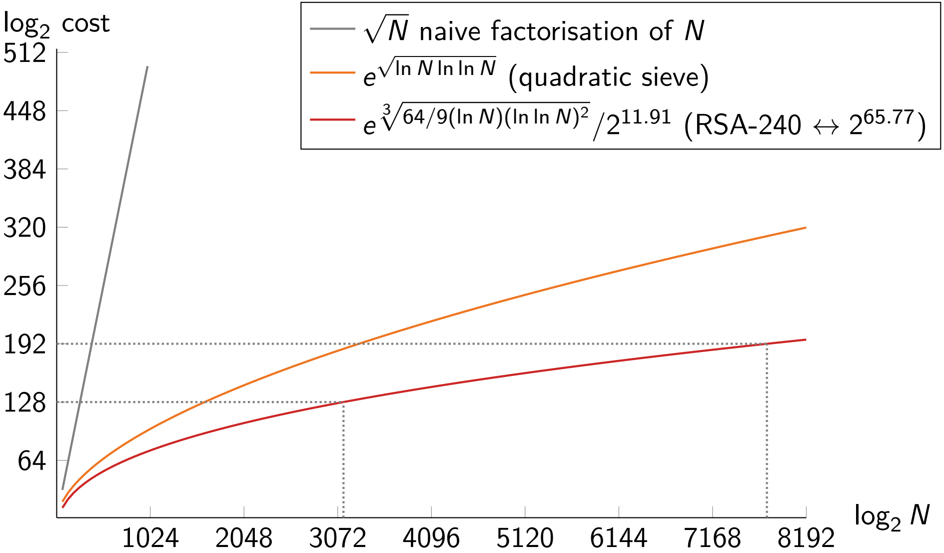
$\gcd(X - Y, N) = 43$, $\gcd(X + Y, N) = 47$

$N = 43 \cdot 47$

Quadratic Sieve: limitations for large numbers

Complexity: $e^{\sqrt{(1+o(1)) \ln N \ln \ln N}}$

- $n = (a+m)^2 - N \approx 2A\sqrt{N}$
  Factor integers of size $\approx 2A\sqrt{N}$
- $\#\mathcal{F} = \#\{$ primes up to $B\} \approx B/\ln B$
- Computes left kernel of huge linear system modulo 2



$\log_2$ cost
512
448
384
320
256
192
128
64
$\sqrt{N}$ naive factorisation of N
$e^{\sqrt{\ln N \ln \ln N}}$ (quadratic sieve)
$e^{\sqrt[3]{64/9(\ln N)(\ln \ln N)^2}}/2^{11.91}$ (RSA-240 $\leftrightarrow 2^{65.77}$)
1024
2048
3072
4096
5120
6144
7168
8192
$\log_2 N$

Outline

Sieving: Detect smooth numbers without factoring

Eratosthenes sieve

Array $T[1 \ldots n-1]$ of integers from 2 up to n

At iteration i, each non-marked integer in $T[1 \ldots i]$ is prime

For each non-marked $p_i = T[i]$ starting with $p_1 = T[1] = 2$:

  Mark as composite all multiples $T[i + kp_i]$, $1 \le k \le (n-i)/p_i$

$[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]$

Sieving: Detect smooth numbers without factoring

Eratosthenes sieve

Array $T[1 \ldots n-1]$ of integers from 2 up to n

At iteration i, each non-marked integer in $T[1 \ldots i]$ is prime

For each non-marked $p_i = T[i]$ starting with $p_1 = T[1] = 2$:

  Mark as composite all multiples $T[i + kp_i]$, $1 \leq k \leq (n-i)/p_i$

$[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]$

Sieving: Detect smooth numbers without factoring

Eratosthenes sieve

Array $T[1 \ldots n-1]$ of integers from 2 up to n
At iteration i, each non-marked integer in $T[1 \ldots i]$ is prime
For each non-marked $p_i = T[i]$ starting with $p_1 = T[1] = 2$:
Mark as composite all multiples $T[i + kp_i]$, $1 \leq k \leq (n-i)/p_i$

$[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]$

Sieving: Detect smooth numbers without factoring

Eratosthenes sieve

Array $T[1 \ldots n-1]$ of integers from 2 up to n
At iteration i, each non-marked integer in $T[1 \ldots i]$ is prime
For each non-marked $p_i = T[i]$ starting with $p_1 = T[1] = 2$:
    Mark as composite all multiples $T[i + kp_i]$, $1 \le k \le (n-i)/p_i$

$[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]$

Sieving: Detect smooth numbers without factoring

Quadratic sieve

1. Initialize array $T[1 \dots A]$ with $T[a] = (a+m)^2 - N$
2. For each prime p_i from 2 to B
   2.1 Solve $(x+m)^2 - N \equiv 0 \bmod p_i \rightarrow$ roots $x_0, x_1 \in [0, p_i - 1]$
   2.2 Update T: divide by p_i the cells $T[x_{0,1} + kp_i]$ for all $0 \leq k \leq (A - x_{0,1})/p_i$
   2.3 Consider higher powers $p_i^{e_i}$: solve $((x_{0,1} + m)^2 - N)/p_i \equiv 0 \bmod p_i$ ...
3. $T[a] = 1 \iff (a+m)^2 - N$ is B-smooth

Sieving: Detect smooth numbers without factoring

Quadratic sieve

1. Initialize array $T[1 \dots A]$ with $T[a] = (a+m)^2 - N$
2. For each prime p_i from 2 to B
   2.1 Solve $(x+m)^2 - N \equiv 0 \bmod p_i \rightarrow$ roots $x_0, x_1 \in [0, p_i - 1]$
   2.2 Update T: divide by p_i the cells $T[x_{0,1} + kp_i]$ for all $0 \leq k \leq (A - x_{0,1})/p_i$
   2.3 Consider higher powers $p_i^{e_i}$: solve $((x_{0,1}+m)^2 - N)/p_i \equiv 0 \bmod p_i$ ...
3. $T[a] = 1 \iff (a+m)^2 - N$ is B-smooth

$(a+m)^2 - N$ is larger than a machine-word:
store $\log_2((a+m)^2 - N)$ at step (1) and subtract $\log_2 p_i$ at step (2)
$T[a] = 0 \iff (a+m)^2 - N$ is B-smooth (up to rounding errors)
Recompute $(a+m)^2 - N$ and factor it
$\rightarrow$ factor only the smooth ones

Sieving: Detect smooth numbers without factoring

Quadratic sieve

1. Initialize array $T[1 \dots A]$ with $T[a] = \log_2((a+m)^2 - N)$
2. For each prime p_i from 2 to B
   2.1 Solve $(x+m)^2 - N \equiv 0 \bmod p_i \rightarrow$ roots $x_0, x_1 \in [0, p_i - 1]$
   2.2 Update T: subtract $\log_2 p_i$ to cells $T[x_{0,1} + kp_i]$ for all $0 \leq k \leq (A - x_{0,1})/p_i$
   2.3 Consider higher powers $p_i^{e_i}$: solve $((x_{0,1} + p_i x + m)^2 - N)/p_i \equiv 0 \bmod p_i$ ...
3. $T[a] = 0 \iff (a+m)^2 - N$ is B-smooth

Sieving: Detect smooth numbers without factoring

Quadratic sieve

1. Initialize array $T[1\dots A]$ with $T[a] = \log_2((a+m)^2 - N)$
2. For each prime p_i from 2 to B
   2.1 Solve $(x+m)^2 - N \equiv 0 \bmod p_i \to$ roots $x_0, x_1 \in [0, p_i - 1]$
   2.2 Update T: subtract $\log_2 p_i$ to cells $T[x_{0,1} + kp_i]$ for all $0 \le k \le (A - x_{0,1})/p_i$
   2.3 Consider higher powers $p_i^{e_i}$: solve $((x_{0,1} + p_i x + m)^2 - N)/p_i \equiv 0 \bmod p_i$ ...
3. $T[a] = 0 \iff (a+m)^2 - N$ is B-smooth

1987: ECM factoring
Do not sieve up to B, set a sieving bound $B_0 < B$
For all $T[a] \le$ ECM-bound,
recompute and run ECM on $(a+m)^2 - N$ with bound B
Store the B-smooth ones for the linear algebra step.

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$

$[-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$

$[-85, 1, 95, 47, 283, 95, 479, 145, 683, 197, 895,$
$251, 1115, 307, 1343, 365, 1579, 425, 1823, 487, 2075]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \dots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$

$[-85, 1, 95, 47, 283, 95, 479, 145, 683, 197, 895,$
$251, 1115, 307, 1343, 365, 1579, 425, 1823, 487, 2075]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$

$[-17, 1, 95, 47, 283, 19, 479, 145, 683, 197, 179,$
$251, 1115, 307, 1343, 73, 1579, 425, 1823, 487, 415]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$

$[-17, 1, 95, 47, 283, 19, 479, 145, 683, 197, 179,$
$251, 1115, 307, 1343, 73, 1579, 425, 1823, 487, 415]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$

$[-17, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 1343, 73, 1579, 85, 1823, 487, 415]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \dots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$

$[-17, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 1343, 73, 1579, 85, 1823, 487, 415]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$

$[-17, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 1343, 73, 1579, 17, 1823, 487, 83]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \dots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$
$p_i = 17, x_0 = 0, x_1 = 14$

$[-17, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 1343, 73, 1579, 17, 1823, 487, 83]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$
$p_i = 17, x_0 = 0, x_1 = 14$

$[-1, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 79, 73, 1579, 1, 1823, 487, 83]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \dots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$
$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$
$p_i = 17, x_0 = 0, x_1 = 14$
$p_i = 19, x_0 = 2, x_1 = 5$

$[-1, 1, 19, 47, 283, 19, 479, 29, 683, 197, 179,$
$251, 223, 307, 79, 73, 1579, 1, 1823, 487, 83]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$
$p_i = 17, x_0 = 0, x_1 = 14$
$p_i = 19, x_0 = 2, x_1 = 5$

$[-1, 1, 1, 47, 283, 1, 479, 29, 683, 197, 179,\ 251, 223, 307, 79, 73, 1579, 1, 1823, 487, 83]$

Sieving: Detect smooth numbers without factoring

$N = 2021,\ B = 19,\ A = 20,\ a \in \{0, \ldots, A\}$
$T = [-85, 4, 95, 188, 283, 380, 479, 580, 683, 788, 895,$
$1004, 1115, 1228, 1343, 1460, 1579, 1700, 1823, 1948, 2075]$

$p_i = 2, x_0 = 1,\ T[1 + 2k]/2^2$
$p_i = 5, x_0 = 0,\ T[0 + 5k]/5$
$p_i = 5, x_1 = 2,\ T[2 + 5k]/5$
$p_i = 5, x_0 = 0 + 4 \cdot 5,\ T[0 + 4 \cdot 5 + 5^2 k]/5,\ x_1 = 2 + 3 \cdot 5,\ T[2 + 3 \cdot 5 + 5^2 k]/5$
$p_i = 17, x_0 = 0, x_1 = 14$
$p_i = 19, x_0 = 2, x_1 = 5$

$[-1, 1, 1, 47, 283, 1, 479, 29, 683, 197, 179, 251, 223, 307, 79, 73, 1579, 1, 1823, 487, 83]$

Outline

Nowadays' method: the Number Field Sieve

- developed in the 80's and 90's
- reduce the size of the numbers to be factored from $A_1\sqrt{N}$ to $A_2^d\sqrt[d]{N}$ for a smaller $A_2 < A_1$ and $d \in \{3, 4, 5, 6\}$
- two huge steps: collecting relations, solving a large sparse system

Carl Pomerance.
A tale of two sieves.
Notices of the AMS, 43(12):1473–1485, Dec 1996.
http://www.ams.org/notices/199612/pomerance.pdf

The development of the NFS algorithm

1985 ElGamal: Discrete logarithms in $GF(p^2)$ with quadratic number fields

1986 Coppersmith, Odlyzko, Schroeppel:
factoring with a quadratic number field (Gaussian integers)

1988 J. M. Pollard, Factoring with cubic integers. Factorization of $F_7 = 2^{2^7} + 1$.
Special Number Field.

1993 Lenstra, Lenstra, Manasse, Pollard. The Number Field Sieve.

Arjen K. Lenstra and Hendrik W. Lenstra Jr., editors.
The development of the number field sieve, volume 1554 of *Lect. Note. Math.*
Springer, 1993.
http://doi.org/10.1007/BFb0091534

Factorization with NFS: recap

1. Polynomial selection: find two irreducible polynomials in $\mathbb{Z}[x]$ sharing a common root m modulo N
2. Relation collection: computes many smooth relations
3. Filtering: remove singletons, densify and shrink the matrix
4. Linear algebra: takes logarithms mod 2 of the relations: large sparse matrix over $\mathbb{F}_2$, computes left kernel
5. Characters: find a combination of the vectors of the kernel so that $X^2 \equiv Y^2 \bmod N$
6. Square root: computes X, Y
7. Factor N: computes $\gcd(X - Y, N)$

Factorization with NFS: key idea

Reduce further the size of the integers to factor

Choose integer $m \approx \sqrt[d]{N}$

Write N in basis m: $N = c_0 + c_1 m + \ldots + c_d m^d$

Set $f_1(x) = c_0 + c_1 x + \ldots + c_d x^d \implies f_1(m) = 0$, set $f_0 = x - m \implies f_0(m) = 0$

Polynomials f_0, f_1 share a common root m modulo N

If f_1 is irreducible, define $\alpha \in \mathbb{C}$ a root of f_1

Factorization with NFS: key idea

Reduce further the size of the integers to factor
Choose integer $m \approx \sqrt[d]{N}$
Write N in basis m: $N = c_0 + c_1 m + \ldots + c_d m^d$
Set $f_1(x) = c_0 + c_1 x + \ldots + c_d x^d \implies f_1(m) = 0$, set $f_0 = x - m \implies f_0(m) = 0$
Polynomials f_0, f_1 share a common root m modulo N
If f_1 is irreducible, define $\alpha \in \mathbb{C}$ a root of f_1

Define a map from $\mathbb{Z}[\alpha]$ to $\mathbb{Z}/N\mathbb{Z}$

$$\begin{array}{rcl} \phi\colon \mathbb{Z}[\alpha] & \to & \mathbb{Z}/N\mathbb{Z} \\ \alpha & \mapsto & m \bmod N \text{ where } f_1(m) = 0 \bmod N \end{array}$$

ring homomorphism $\phi(a + b\alpha) = a + bm$

$$\phi \underbrace{(a + b\alpha)}_{\text{factor in } \mathbb{Z}[\alpha]} = \underbrace{a + bm}_{\text{factor in } \mathbb{Z}} \mod N$$

Factorization with NFS: key idea

Reduce further the size of the integers to factor

Choose integer $m \approx \sqrt[d]{N}$

Write N in basis m: $N = c_0 + c_1 m + \ldots + c_d m^d$

Set $f_1(x) = c_0 + c_1 x + \ldots + c_d x^d \implies f_1(m) = 0$, set $f_0 = x - m \implies f_0(m) = 0$

Polynomials f_0, f_1 share a common root m modulo N

If f_1 is irreducible, define $\alpha \in \mathbb{C}$ a root of f_1

Define a map from $\mathbb{Z}[\alpha]$ to $\mathbb{Z}/N\mathbb{Z}$

$$\begin{array}{rcl} \phi\colon \mathbb{Z}[\alpha] & \to & \mathbb{Z}/N\mathbb{Z} \\ \alpha & \mapsto & m \bmod N \text{ where } f_1(m) = 0 \bmod N \end{array}$$

ring homomorphism $\phi(a + b\alpha) = a + bm$

$$\phi \underbrace{(a + b\alpha)}_{\substack{\text{factor in } \mathbb{Z}[\alpha] \\ \text{size } A^d N^{1/d}}} = \underbrace{a + bm}_{\substack{\text{factor in } \mathbb{Z} \\ \text{size } AN^{1/d}}} \mod N$$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$

$m = 38,\ 7 + 15m + m^2 = N,\ f_1(x) = x^2 + 15x + 7,\ f_0 = x - m$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$

$m = 38$, $7 + 15m + m^2 = N$, $f_1(x) = x^2 + 15x + 7$, $f_0 = x - m$

$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:

$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$

$m = 38$, $7 + 15m + m^2 = N$, $f_1(x) = x^2 + 15x + 7$, $f_0 = x - m$

$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:

$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$

Factorization in $\mathbb{Z}[i]$, $i \in \mathbb{C}$, $i^2 = -1$:

$(1 + i)(1 - i) = 2$, $(2 + i)(2 - i) = 5$, $(2 + 3i)(2 - 3i) = 13$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$

$m = 38$, $7 + 15m + m^2 = N$, $f_1(x) = x^2 + 15x + 7$, $f_0 = x - m$

$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:

$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$

Factorization in $\mathbb{Z}[i]$, $i \in \mathbb{C}$, $i^2 = -1$:

$(1 + i)(1 - i) = 2$, $(2 + i)(2 - i) = 5$, $(2 + 3i)(2 - 3i) = 13$

Fondamental Unit: $u = 2\alpha + 1$ and $\mathrm{Norm}(u) = 1$

Norm

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$

$m = 38$, $7 + 15m + m^2 = N$, $f_1(x) = x^2 + 15x + 7$, $f_0 = x - m$

$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:

$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$

Factorization in $\mathbb{Z}[i]$, $i \in \mathbb{C}$, $i^2 = -1$:

$(1+i)(1-i) = 2$, $(2+i)(2-i) = 5$, $(2+3i)(2-3i) = 13$

Fondamental Unit: $u = 2\alpha + 1$ and $\text{Norm}(u) = 1$

Norm

The norm of $a - bi$ in $\mathbb{Z}[i]$ is $\text{Norm}(a - bi) = a^2 + b^2$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$
$m = 38,\ 7 + 15m + m^2 = N,\ f_1(x) = x^2 + 15x + 7,\ f_0 = x - m$
$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:
$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$
Factorization in $\mathbb{Z}[i]$, $i \in \mathbb{C}$, $i^2 = -1$:
$(1+i)(1-i) = 2$, $(2+i)(2-i) = 5$, $(2+3i)(2-3i) = 13$
Fondamental Unit: $u = 2\alpha + 1$ and $\text{Norm}(u) = 1$

Norm

The norm of $a - bi$ in $\mathbb{Z}[i]$ is $\text{Norm}(a - bi) = a^2 + b^2$
The norm of $a - b\alpha$ in $\mathbb{Z}[\alpha]$ is

$$\text{Norm}(a - b\alpha) = b^2 f(a/b) = a^2 + 15ab + 7b^2$$

Factorization in $\mathbb{Z}[\alpha]$

Factor $N = 2021$
$m = 38$, $7 + 15m + m^2 = N$, $f_1(x) = x^2 + 15x + 7$, $f_0 = x - m$
$\alpha \in \mathbb{C}$ a root of f_1, factorization in $\mathbb{Z}[\alpha]$:
$7 = (7^+)(7^-)$,$19 = (19^+)(19^-)$,$23 = (23^+)(23^-)$
Factorization in $\mathbb{Z}[i]$, $i \in \mathbb{C}$, $i^2 = -1$:
$(1+i)(1-i) = 2$, $(2+i)(2-i) = 5$, $(2+3i)(2-3i) = 13$
Fondamental Unit: $u = 2\alpha + 1$ and $\text{Norm}(u) = 1$

Norm

The norm of $a - bi$ in $\mathbb{Z}[i]$ is $\text{Norm}(a - bi) = a^2 + b^2$
The norm of $a - b\alpha$ in $\mathbb{Z}[\alpha]$ is

$$\text{Norm}(a - b\alpha) = b^2 f(a/b) = a^2 + 15ab + 7b^2$$

To factor $a - b\alpha \in \mathbb{Z}[\alpha]$,
compute $\text{Norm}(a - b\alpha) \in \mathbb{Z}$ and factor in $\mathbb{Z}$
$\rightarrow$ To factor N, factor many smaller integers.

a,b	$a - bm =$ factor in $\mathbb{Z}$	$a^2 + 15ab + 7b^2$	factor in $\mathbb{Z}[\alpha]$
$-23,2$	$-99 = -3^2 \cdot 11$	$-133 = -7 \cdot 19$	$(7^+)(19^+)$
$-22,1$	$-60 = -2^2 \cdot 3 \cdot 5$	$161 = 7 \cdot 23$	$(7^+)(23^+)$
$-16,1$	$-54 = -2 \cdot 3^3$	$23 = 23$	(23^-)
$-14,1$	$-52 = -2^2 \cdot 13$	$-7 = -7$	(7^-)
$-13,1$	$-51 = -3 \cdot 17$	$-19 = -19$	(19^-)
$-9,2$	$-85 = -5 \cdot 17$	$-161 = -7 \cdot 23$	$(7^+)(23^-)$
$-8,5$	$-198 = -2 \cdot 3^2 \cdot 11$	$-361 = -19^2$	$(19^-)^2$
$-8,15$	$-578 = -2 \cdot 17^2$	$-161 = -7 \cdot 23$	$(7^+)(23^+)$
$-7,1$	$-45 = -3^2 \cdot 5$	$-49 = -7^2$	$(7^-)^2$
$-6,13$	$-500 = -2^2 \cdot 5^3$	$49 = 7^2$	$(7^+)^2$
$-2,1$	$-40 = -2^3 \cdot 5$	$-19 = -19$	(19^+)
$-1,1$	$-39 = -3 \cdot 13$	$-7 = -7$	(7^+)
$-1,2$	$-77 = -7 \cdot 11$	$-1 = -1$	
$5,4$	$-147 = -3 \cdot 7^2$	$437 = 19 \cdot 23$	$(19^-)(23^-)$
$6,1$	$-32 = -2^5$	$133 = 7 \cdot 19$	$(7^+)(19^-)$
$7,6$	$-221 = -13 \cdot 17$	$931 = 7^2 \cdot 19$	$(7^-)^2(19^+)$

Example in $\mathbb{Z}[\alpha]$: Matrix

Build the matrix of relations:

- one row per (a, b) pair s.t. both sides are smooth
- one column per prime $\{2, 3, 5, 7, 11, 13, 17\}$
- one column per prime ideal $(7^+), (7^-), (19^+), (19^-), (23^+), (23^-)$
- store the exponents mod 2

Example in $\mathbb{Z}[\alpha]$: Matrix

$M =$

2	3	5	7	11	13	17	(7^+)	(7^-)	(19^+)	(19^-)	(23^+)	(23^-)
0	2	0	0	1	0	0	1	0	1	0	0	0
2	1	1	0	0	0	0	1	0	0	0	1	0
1	3	0	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	1	0	0	1	0	0	0	0
0	1	0	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	1	0	0	0	0	1
1	2	0	0	1	0	0	0	0	0	2	0	0
1	0	0	0	0	0	2	1	0	0	0	1	0
0	2	1	0	0	0	0	0	2	0	0	0	0
2	0	3	0	0	0	0	2	0	0	0	0	0
3	0	1	0	0	0	0	0	0	1	0	0	0
0	1	0	0	0	1	0	1	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0
0	1	0	2	0	0	0	0	0	0	1	0	1
5	0	0	0	0	0	0	1	0	0	1	0	0
0	0	0	0	0	1	1	0	2	1	0	0	0

Example in $\mathbb{Z}[\alpha]$: Matrix

$$
\begin{array}{c}
\begin{matrix} 2 & 3 & 5 & 7 & 11 & 13 & 17 & (7^+) & (7^-) & (19^+) & (19^-) & (23^+) & (23^-) \end{matrix} \\
M = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix} \mod 2
\end{array}
$$

Example in $\mathbb{Z}[\alpha]$: Matrix

$$
\begin{array}{c}
\begin{array}{ccccccccccccc}
2 & 3 & 5 & 7 & 11 & 13 & 17 & (7^+) & (7^-) & (19^+) & (19^-) & (23^+) & (23^-)
\end{array}
\\
M = \begin{bmatrix}
 & & & & 1 & & & 1 & & 1 & & & \\
 & 1 & 1 & & & & & 1 & & & & 1 & \\
1 & 1 & & & & & & & & & & & 1 \\
 & & & & & 1 & & & 1 & & & & \\
 & 1 & & & & & 1 & & & & 1 & & \\
 & & 1 & & & & 1 & 1 & & & & & 1 \\
1 & & & & 1 & & & & & & & & \\
1 & & & & & & & 1 & & & & 1 & \\
 & & 1 & & & & & & & & & & \\
 & & 1 & & & & & & & & & & \\
1 & & 1 & & & & & & & 1 & & & \\
 & 1 & & & & 1 & & 1 & & & & & \\
 & & & 1 & 1 & & & & & & & & \\
 & 1 & & & & & & & & & 1 & & 1 \\
1 & & & & & & & 1 & & & 1 & & \\
 & & & & & 1 & 1 & & & 1 & & &
\end{bmatrix} \text{ sparse}
\end{array}
$$

Example: from left kernel in GF(2) to factorization

$$\ker(M) \quad \begin{matrix} (\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1\ ) \\ (\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1\ ) \\ (\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1\ ) \\ (\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\ ) \end{matrix}$$

Example: from left kernel in GF(2) to factorization

$$\ker(M) \quad \begin{array}{l} (\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ ) \\ (\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ ) \\ (\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ ) \\ (\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ ) \end{array}$$

Relations #9 and #10:

$$\left| \begin{array}{l} (-7-m) = -45 = -3^2 \cdot 5 \\ (-6-13m) = -500 = -2^2 \cdot 5^3 \end{array} \right| \quad \begin{array}{l} -7-\alpha = (7^-)^2 \\ -6-13\alpha = (7^+)^2 \end{array}$$

Example: from left kernel in GF(2) to factorization

$$\ker(M) \quad \begin{matrix} (1&0&0&0&0&1&1&0&0&1&0&1&0&1&1&1) \\ (0&0&1&0&0&1&0&0&0&0&1&1&0&0&0&1) \\ (0&0&0&0&1&0&0&0&0&1&1&1&0&0&1&1) \\ (0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0) \end{matrix}$$

Relations #9 and #10: $\left| \begin{array}{l} (-7-m) = -45 = -3^2 \cdot 5 \\ (-6-13m) = -500 = -2^2 \cdot 5^3 \end{array} \right| \begin{array}{l} -7-\alpha = (7^-)^2 \\ -6-13\alpha = (7^+)^2 \end{array}$

$(-7-m)(-6-13m) = 150^2$, but $(-7-\alpha)(-6-13\alpha) = -49-98\alpha$ not square

because of the units

Example: from left kernel in GF(2) to factorization

$$\ker(M) \quad \begin{matrix} (\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1\ ) \\ (\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1\ ) \\ (\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1\ ) \\ (\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\ ) \end{matrix}$$

Relations #9 and #10: $\left| \begin{array}{l} (-7-m) \quad = -45 = -3^2 \cdot 5 \\ (-6-13m) = -500 = -2^2 \cdot 5^3 \end{array} \right| \quad \begin{array}{l} -7-\alpha \quad = (7^-)^2 \\ -6-13\alpha = (7^+)^2 \end{array}$

$(-7-m)(-6-13m) = 150^2$, but $(-7-\alpha)(-6-13\alpha) = -49-98\alpha$ not square

because of the units

Relations # $\{5, 10, 11, 12, 15, 16\}$:

$(-13-m)(-6-13m)(-2-m)(-1-m)(6-m)(7-6m) = 530400^2$

$(-13-\alpha)(-6-13\alpha)(-2-\alpha)(-1-\alpha)(6-\alpha)(7-6\alpha) = -3113264-6456485\alpha$

not square

Example: from left kernel in GF(2) to factorization

$$\ker(M)\quad \begin{matrix} (1&0&0&0&0&1&1&0&0&1&0&1&0&1&1&1)\\ (0&0&1&0&0&1&0&0&0&0&1&1&0&0&0&1)\\ (0&0&0&0&1&0&0&0&0&1&1&1&0&0&1&1)\\ (0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0)\end{matrix}$$

Relations #9 and #10: $\left|\begin{matrix}(-7-m) = -45 = -3^2\cdot 5\\ (-6-13m) = -500 = -2^2\cdot 5^3\end{matrix}\right| \quad \begin{matrix}-7-\alpha = (7^-)^2\\ -6-13\alpha = (7^+)^2\end{matrix}$

$(-7-m)(-6-13m) = 150^2$, but $(-7-\alpha)(-6-13\alpha) = -49-98\alpha$ not square

because of the units

Relations # $\{5, 10, 11, 12, 15, 16\}$:

$(-13-m)(-6-13m)(-2-m)(-1-m)(6-m)(7-6m) = 530400^2$

$(-13-\alpha)(-6-13\alpha)(-2-\alpha)(-1-\alpha)(6-\alpha)(7-6\alpha) = -3113264-6456485\alpha$

not square $\rightarrow$ multiply both

$(-49-98\alpha)(-3113264-6456485\alpha) = (-12103-25137\alpha)^2$ square

$X = 150\cdot 530400 = 1314 \bmod N$ $\qquad$ $Y = (-12103-25137m) = 750 \bmod N$

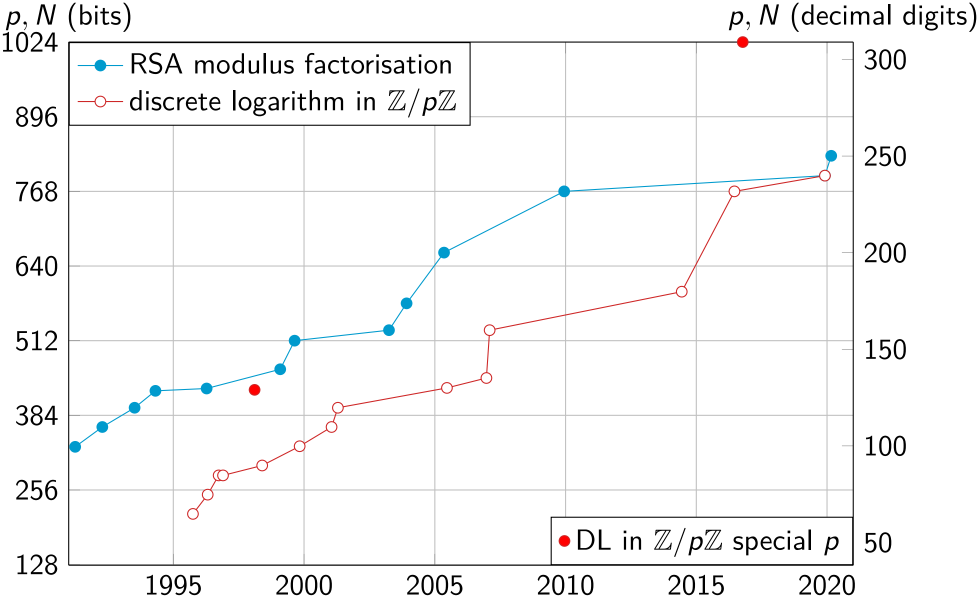
$\gcd(X-Y, N) = 47$, $\gcd(X+Y, N) = 43$ $\qquad$ $N = 43\cdot 47$

Factorization with NFS: recap

1. Polynomial selection: find two irreducible polynomials in $\mathbb{Z}[x]$ sharing a common root m modulo N
2. Relation collection: computes many smooth relations
3. Filtering: remove singletons, densify and shrink the matrix
4. Linear algebra: takes logarithms mod 2 of the relations: large sparse matrix over $\mathbb{F}_2$, computes left kernel
5. Characters: find a combination of the vectors of the kernel so that $X^2 \equiv Y^2 \bmod N$
6. Square root: computes X, Y
7. Factor N: computes $\gcd(X - Y, N)$

Outline

Record computations

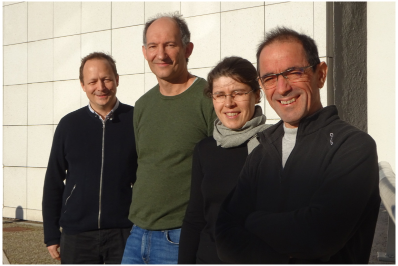


Latest record computations

Fabrice Boudot, Pierrick Gaudry, Aurore Guillevic, Nadia Heninger, Emmanuel Thomé, and Paul Zimmermann.
Comparing the difficulty of factorization and discrete logarithm: A 240-digit experiment.
In Daniele Micciancio and Thomas Ristenpart, eds., *CRYPTO 2020, Part II*, vol. 12171 of *LNCS*, pp. 62–91. Springer, August 2020.

Factorization of RSA-240 (795 bits) in December 2019 and RSA-250 (829 bits) in February 2020

Video at Crypto'2020: https://youtube.com/watch?v=Qk207A4H7kU



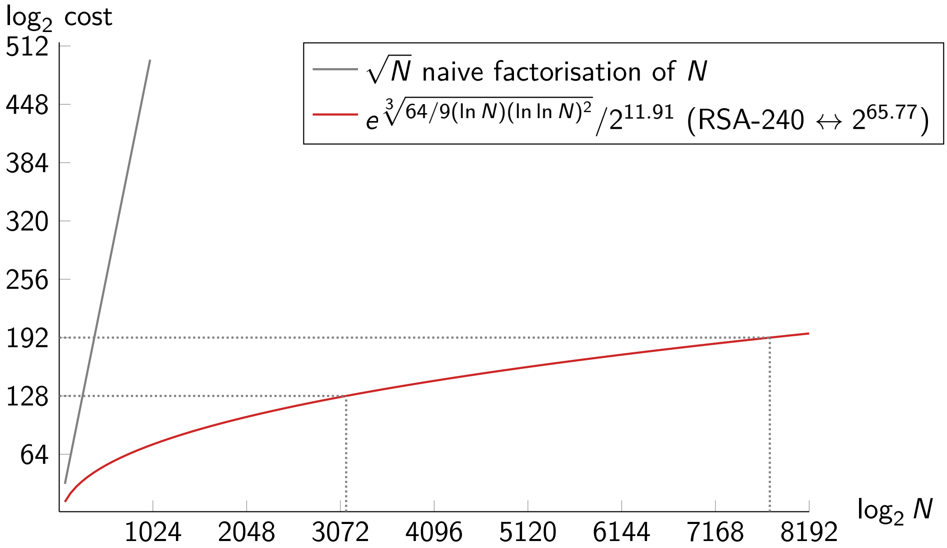
Emmanuel, Pierrick, Aurore, Paul in Nancy. Not on the picture: Fabrice, Nadia.

Latest record computations

$$
\begin{aligned}
\text{RSA-240} &= 124620366781718784065835044608106590434820374651678805754818\\
&\quad 788883289666801188210855036039570272508747509864768438458621\\
&\quad 054865537970253930571891217684318286362846948405301614416430\\
&\quad 468066875699415246993185704183030512549594371372159029236099,\\
p &= 509435952285839914555051023580843714132648382024111473186660\\
&\quad 296521821206469746700620316443478873837606252372049619334517,\\
q &= 244624208838318150567813139024002896653802092578931401452041\\
&\quad 221336558477095178155258218897735030590669041302045908071447.
\end{aligned}
$$

Latest record computations

$$
\begin{aligned}
\text{RSA-250} &= 2140324650240744961264423072839333563008614715144755017797754920 \\
&\ 8814180234471401366433455190958046796109928518724709145876873 9 \\
&\ 6261921557363047454770520805119056493106687691590019759405693 45 \\
&\ 7452230589325976697471681738069364894699871578494975937497937, \\
p &= 6413528947707158027879019017057738908482501474294344720811685 96 \\
&\ 32024532344630238623598752668347708737661925585694639798853367, \\
q &= 3337202759497815655622601060535511422794076034476755466678452 09 \\
&\ 87023841729210037080257448673296881877565718986258036932062711
\end{aligned}
$$



RSA-240: 953 core-years, Intel Xeon Gold 6130 CPUs as a reference (2.1GHz)
$\approx 953 \cdot 365.25 \cdot 24 \cdot 60 \cdot 60 \cdot 2.1 \cdot 10^9 \approx 2^{65.77}$

Breaking the previous record: Why?

- Record computations needed for key-size recommendations
- Open-source software Cado-NFS
- Motivation to improve all the steps
- Testing folklore ideas competitive only for huge sizes
- Exploits improvements of ECM (Bouvier–Imbert PKC'2020)
- Scaling the code for larger sizes improves the running-time on smaller sizes

The CADO-NFS software

Record computations with the CADO-NFS software.

- Important software development effort since 2007.
- 250k lines of C/C++ code, 60k for relation collection only.
- Significant improvements since 2016.
  - improved parallelism: strive to get rid of scheduling bubbles;
  - versatility: large freedom in parameter selection;
  - prediction of behaviour and yield: essential for tuning.
- Open source (LGPL), open development model (gitlab).
  Our results can be reproduced.

Factorization of N = RSA-240, 240 decimal digits

Polynomial selection

$m = m_1/m_2 = 105487753732969860223795041295860517380/17780390513045005995253$

$f_1 = 10853204947200x^6$
$-4763683724115259920x^5$
$-6381744461279867941961670x^4$
$+974448934853864807690675067037x^3$
$+179200573533665721310210640738061170x^2$
$+1595712553369335430496125795083146688523x$
$-221175588842299117590564542609977016567191860$

$f_0 = 17780390513045005995253x$
$-105487753732969860223795041295860517380$

$\mathrm{Res}(f_0, f_1) = 120N$

Integers $(am_2 - bm_1)$ much smaller than
$\mathrm{Norm}_{f_1}(a - b\alpha) = c_0 b^6 + c_1 ab^5 + c_2 a^2 b^4 + c_3 a^3 b^3 + c_4 a^4 b^2 + c_5 a^5 b + c_6 a^6$,
$f_1 = c_0 + c_1 x + \ldots + c_6 x^6$

Relation collection with lattice sieving

Most time-consuming part.
How to enumerate (a, b), and detect smooth $a - b\alpha$, $am_2 - bm_1$?

Special-$\mathfrak{q}$ (spq) Sieving

2-dimension array T of norm of $i - j\alpha$ all multiple of prime q_k

Allow Parallelization

Consider all primes $q_i \in$[0.8G, 7.4G] (G=10^9) s.t. $\exists$ $\mathfrak{q}$

- for $q_i \in$[0.8G, 2.1G]: **Lattice Sieve** on both sides
- for $q_i \in$[2.1G, 7.4G]: **Lattice Sieve** for f_1 (large norms) and **Factorization Tree** for f_0 (much smaller norms)

$\#$ spq $\approx 3.0e8 \approx 2^{28}$
Sieve area per spq: $\mathcal{A} = [-2^{15}, 2^{15}] \times [0, 2^{16}]$, $\#\mathcal{A} = 2^{32}$

Relations look like

small primes, special-q, large primes

✓	$5^2 \cdot 11 \cdot 23 \cdot 287093 \cdot 870953 \cdot 20179693 \cdot 28306698811 \cdot 47988583469$	$2^3 \cdot 5 \cdot 7 \cdot 13 \cdot 31 \cdot 61 \cdot 14407 \cdot 26563253 \cdot 86800081 \cdot 269845309 \cdot 802234039 \cdot 1041872869 \cdot 5552238917 \cdot 12144939971 \cdot 15856830239$
✓	$3 \cdot 1609 \cdot 77699 \cdot 235586599 \cdot 347727169 \cdot 369575231 \cdot 9087872491$	$2^3 \cdot 3 \cdot 5 \cdot 13 \cdot 19 \cdot 23 \cdot 31 \cdot 59 \cdot 239 \cdot 3989 \cdot 7951 \cdot 2829403 \cdot 31455623 \cdot 225623753 \cdot 811073867 \cdot 1304127157 \cdot 78955382651 \cdot 129320018741$
✓	$5 \cdot 1381 \cdot 877027 \cdot 15060047 \cdot 19042511 \cdot 11542780393 \cdot 13192388543$	$2^4 \cdot 5 \cdot 13 \cdot 31 \cdot 59 \cdot 823 \cdot 2801 \cdot 26539 \cdot 2944817 \cdot 3066253 \cdot 87271397 \cdot 108272617 \cdot 386616343 \cdot 815320151 \cdot 1361785079 \cdot 12322934353$
✓	$2^3 \cdot 5^2 \cdot 173 \cdot 971 \cdot 613909489 \cdot 929507779 \cdot 1319454803 \cdot 2101983503$	$2^7 \cdot 3^2 \cdot 5 \cdot 29 \cdot 1021 \cdot 42589 \cdot 190507 \cdot 473287 \cdot 31555663 \cdot 654820381 \cdot 802234039 \cdot 19147596953 \cdot 23912934131 \cdot 52023180217$
✗	$2^2 \cdot 15193 \cdot 232891 \cdot 19514983 \cdot 139295419 \cdot 540260173 \cdot 606335449$	$2^2 \cdot 3^4 \cdot 13 \cdot 19 \cdot 74897 \cdot 1377667 \cdot 55828453 \cdot 282012013 \cdot 802234039 \cdot 3350122463 \cdot 35787642311 \cdot 37023373909 \cdot 128377293101$
✗	$2^2 \cdot 5^4 \cdot 439 \cdot 1483 \cdot 13121 \cdot 21383 \cdot 67751 \cdot 452059523 \cdot 33099515051$	$2^2 \cdot 3^3 \cdot 11 \cdot 13 \cdot 19 \cdot 5023 \cdot 3683209 \cdot 98660459 \cdot 802234039 \cdot 1506372871 \cdot 4564625921 \cdot 27735876911 \cdot 32612130959 \cdot 45729461779$

small primes: abundant → dense column in the matrix

large primes: rare → sparse colum, limit to 2 or 3 on each side.

Relations look like

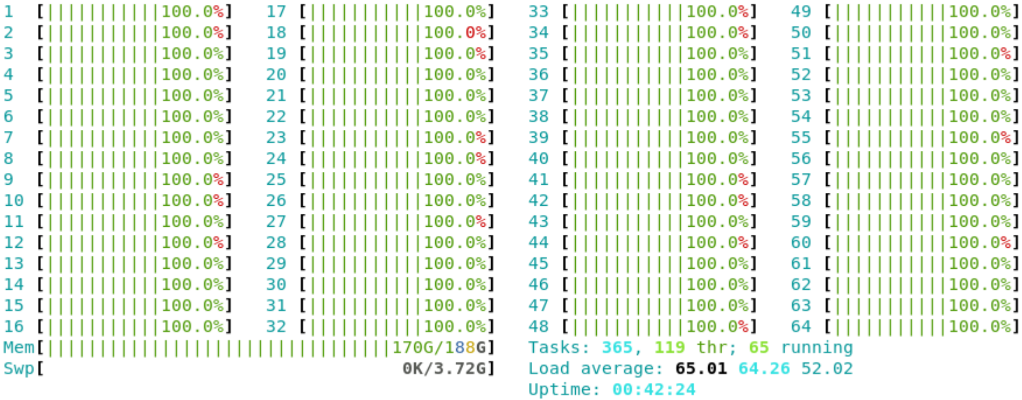
small primes, special-q, large primes

✓	$5^2 \cdot 11 \cdot 23 \cdot 287093 \cdot 870953 \cdot 20179693 \cdot 28306698811 \cdot 47988583469$	$2^3 \cdot 5 \cdot 7 \cdot 13 \cdot 31 \cdot 61 \cdot 14407 \cdot 26563253 \cdot 86800081 \cdot 269845309 \cdot 802234039 \cdot 1041872869 \cdot 5552238917 \cdot 12144939971 \cdot 15856830239$
✓	$3 \cdot 1609 \cdot 77699 \cdot 235586599 \cdot 347727169 \cdot 369575231 \cdot 9087872491$	$2^3 \cdot 3 \cdot 5 \cdot 13 \cdot 19 \cdot 23 \cdot 31 \cdot 59 \cdot 239 \cdot 3989 \cdot 7951 \cdot 2829403 \cdot 31455623 \cdot 225623753 \cdot 811073867 \cdot 1304127157 \cdot 78955382651 \cdot 129320018741$
✓	$5 \cdot 1381 \cdot 877027 \cdot 15060047 \cdot 19042511 \cdot 11542780393 \cdot 13192388543$	$2^4 \cdot 5 \cdot 13 \cdot 31 \cdot 59 \cdot 823 \cdot 2801 \cdot 26539 \cdot 2944817 \cdot 3066253 \cdot 87271397 \cdot 108272617 \cdot 386616343 \cdot 815320151 \cdot 1361785079 \cdot 12322934353$
✓	$2^3 \cdot 5^2 \cdot 173 \cdot 971 \cdot 613909489 \cdot 929507779 \cdot 1319454803 \cdot 2101983503$	$2^7 \cdot 3^2 \cdot 5 \cdot 29 \cdot 1021 \cdot 42589 \cdot 190507 \cdot 473287 \cdot 31555663 \cdot 654820381 \cdot 802234039 \cdot 19147596953 \cdot 23912934131 \cdot 52023180217$

small primes: abundant → dense column in the matrix
large primes: rare → sparse colum, limit to 2 or 3 on each side.

Before linear algebra: filtering step
as many cheap combinations as possible → smaller matrix

Relation collection looks like



Discrete logarithm problem

$\mathbf{G}$ multiplicative group of order ℓ
g generator, $\mathbf{G} = \{1, g, g^2, g^3, \ldots, g^{\ell-2}, g^{\ell-1}\}$

Given $h \in \mathbf{G}$, find integer $x \in \{0, 1, \ldots, \ell - 1\}$ such that $h = g^x$.
Exponentiation easy: $(g, x) \mapsto g^x$
Discrete logarithm hard in well-chosen groups $\mathbf{G}$

Choice of group

Prime finite field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ where p is a prime integer
Multiplicative group: $\mathbb{F}_p^* = \{1, 2, \ldots, p-1\}$
Multiplication *modulo p*

Finite field $\mathbb{F}_{2^n} = \mathrm{GF}(2^n)$, $\mathbb{F}_{3^m} = \mathrm{GF}(3^m)$ for efficient arithmetic, now broken

Elliptic curves $E\colon y^2 = x^3 + ax + b/\mathbb{F}_p$, $E_a\colon y^2 + xy = x^3 + ax^2 + 1/\mathbb{F}_{2^n}$

Discrete Logarithm 240 dd

$p = N + 49204$, $\ell = (p-1)/2$ prime

$f_1 = 39x^4 + 126x^3 + x^2 + 62x + 120$

$$\begin{aligned} f_0 = &\ 286512172700675411986966846394359924874576536408786368056\, x^3 \\ &+ 249088203007157661364751159824397355165818886038172555398 90\, x^2 \\ &- 187636975600130165644039539283271210355804094599448546527 37\, x \\ &- 236610408827000256250190838220824122997878994595785432202599 \end{aligned}$$

$\mathrm{Res}(f_0, f_1) = -540p$

More balanced integers

Smaller matrix but kernel modulo large prime ℓ

Relations, matrix size, core-years timings

	RSA-240	DLP-240
polynomial selection	76 core-years	152 core-years
deg f_0, deg f_1	1, 6	3, 4
relation collection	794 core-years	2400 core-years
raw relations	8 936 812 502	3 824 340 698
unique relations	6 011 911 051	2 380 725 637
filtering	days	days
after singleton removal	$2\,603\,459\,110 \times 2\,383\,461\,671$	$1\,304\,822\,186 \times 1\,000\,258\,769$
after clique removal	$1\,175\,353\,278 \times 1\,175\,353\,118$	$149\,898\,095 \times 149\,898\,092$
after merge	282M rows, density 200	36M rows, density 253
linear algebra	83 core-years	625 core-years
characters, sqrt, ind log	days	days
total	953 core-years $\approx 2^{65.77}$ op.	3177 core-years $\approx 2^{67.51}$ op.

Intel Xeon Gold 6130 CPUs as a reference (2.1GHz)

RSA-240 record computation

- Parameterization strategies
- Extensive simulation framework for parameter choices
- Implementation scales well

RSA-240 record computation

- Parameterization strategies
- Extensive simulation framework for parameter choices
- Implementation scales well

Comparisons:

- Comparing RSA-240 to 10 years old previous record not meaningful
- Comparing DL-240 to previous record (DLP-768, 232 digits, 2016): On identical hardware, our DLP-240 computation would have taken 25% less time than the 232-digits computation.
- Finite field DLP is not much harder than integer factoring.

choosing RSA modulus keysizes

- 512 bits: factorization in 7.5 h at cost \$100 on Amazon EC2
  `RSA_EXPORT` ciphersuite in SSL/TLS $\rightarrow$ FREAK attack (2015)
- 768 bits (232 dd): 2009
- 795 bits (240 dd): 2019
- 829 bits (250 dd): 2020
- 1024 bits: $\sim 2^{75}$ op. to factor, to be avoided
- 2048 bits: $\sim 2^{105}$, was standard until 2020 (ANSSI)
- 3072 bits: $\sim 2^{128}$, standard size $\iff$ 256-bit elliptic curves
- 4096 bits: $\sim 2^{145}$, high security

Outline

Attacks on the RSA cryptosystem

Survey paper by Dan Boneh in 1999:

Dan Boneh.
Twenty years of attacks on the RSA cryptosystem.
Notices of the AMS, 46(2):203–213, February 1999.

Too short keys: Humpich episode (1997 in France)

`http://www.bibmath.net/crypto/index.php?action=affiche&quoi=moderne/cb`

In 1997, the keys in payment cards were 320-bit long (96 decimal digits)
Serge Humpich: reverse-engineering, yescard, factorization of a 320-bit key
Showed that possible to pay with a non-legitimate card (RATP tickets)
Possible to factor such keys with the *quadratic sieve*
March 4, 2000: the keys of *GIE carte bancaire* and their factors were released on Internet

Nowadays 1152-bit keys (in 2020)

Wrong key sizes: Bitcrypt ransomware (2014)

https://airbus-cyber-security.com/fr/bitcrypt-broken/

Fabien Perigaud and Cédric Pernet, Airbus Cybersecurity (formerly Cassidian)

ransomware: encrypt the files of target computers

Asks to pay in bitcoins

Encryption: with AES

AES keys encrypted with RSA

But not RSA-1024 (bits)

$$\begin{aligned} N &= 3129884719662540063950693863716193016278901146429595260054414582 \\ & 9335849533528834917800088971765784757175491347320005860302574523 \end{aligned}$$

1024 bits = 128 bytes but the key was 128 decimal digit long (424 bits)!

Factorization with `cado-nfs`

$p = 4627583475399516037897017387039865329961620697520288948716924853$

$q = 6763540271723193027434512605129229364869394443946560226417693\,91.$

Gcd attack (2012, 2013)

N 2048 bits: p, q of 1024 bits, $\approx 2^{1014}$ prime numbers of 1024 bits
Good randomness is very important to be sure that no one will share a factor
Attack:

- scan the internet: collect certificates with RSA keys
- compute the gcd of each possible pair of keys
- optimise the search: *batch gcd*, product-tree
- non-trivial gcd were found!

$N_1 = p_1 q$, $N_2 = p_2 q$, then $\gcd(N_1, N_2) = q$ and the factorisation of N_1 and N_2 is found.

Coppersmith attack (2013), 1/2 Gcd and Patterns

Taiwan system of digital ID (tax payment, car registration...)

- More than 2 million of 1024-bit RSA public keys (2 086 177)
- Batch gcd over the keys: 103 public keys factor into 119 different primes
  206 distinct primes required for 103 independent RSA keys
- Pattern found in the primes, no entropy source, no random number generator
- Testing all primes following the expected pattern (164 primes) $\rightarrow$ 18 more factorizations

The most common prime factor (found in 46 distinct RSA moduli) was
$p = 2^{511} + 2^{510} + 761$ next prime after $2^{511} + 2^{510}$

Coppersmith attack (2013), 2/2

p and q follow a pattern except for the low bits because of `next_prime`
$a =$ `0xc9242492249292499249492449242492249292499249492449242492249292499249492449242492249292499249492449242492`
Coppersmith attack: if the high bits of p are known, can recover the low bits and the factor p

```
p = next_prime(2**511 + 2**510)
q = 0xc9242492249292499249492449242492249292499249492449242492249292499249 4
N = p * q
X = 2**168
a = 0xc9242492249292499249492449242492249292499249492449242492249292499249 4
M = Matrix(3, 3, [X**2, X*a, 0, 0, X, a, 0, 0, N])
R = M.LLL()
g0 = R[0][2]
g1 = R[0][1] // X
g2 = R[0][0] // X**2
c = gcd([g0,g1,g2]) # gcd of coefficients
ZZx.<x> = ZZ[]
g = (g0 + g1*x + g2*x**2) // c
g.factor()
# (x - 83) * (30064312327*x - 23972510637500)
g(83) == 0
q == a + 83
```

RSA and the quantum computer

1994: Peter Shor, algorithm for integer factorization with a quantum computer

Factorization of a n-bit integer requires a perfect quantum computer with $2n$ qbits (quantum bits)

Quantum computer extremely hard to build

Record computation in 2018: $4\ 088\ 459 = 2017 \times 2027$

RSA-1024 (bits) will be factored before a quantum compter become competitive.

Summary of RSA best practices

Use elliptic curve cryptography.
If that's not an option:

- Choose RSA modulus N at least 2048 bits, preferably 3072 bits.
- Use a good random number generator to generate primes.
- Use a secure, randomized padding scheme.

Conclusion

Slides at https://members.loria.fr/AGuillevic/teaching/

Future Milestones in the forthcoming decades: RSA-896, RSA-1024?