RSA, integer factorization, Diffie–Hellman, discrete logarithm computation

Aurore Guillevic

Inria Nancy, France

November 12, 2020

Aurore Guillevic aurore.guillevic@inria.fr

- L1-L2 at Université de Bretagne Sud, Lorient (2005–2007)
- L3 Vannes (2007–2008)
- M1-M2 maths and cryptography at Université de Rennes 1 (2008–2010)
- internship and PhD at Thales Communication, Gennevilliers (92)
- post-doc at Inria Saclay (2 years) and Calgary (Canada, 1 year)
- researcher in cryptography at Inria Nancy since November 2016
- adjunct assistant professor at Polytechnique (2017–2020)

Outline

Introduction: public-key cryptography

Introduced in 1976 (Diffie–Hellman, DH) and 1977 (Rivert–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)

Textbooks


Alfred Menezes, Paul C. van Oorschot, and Scott A. Vanstone.
Handbook of Applied Cryptography.
CRC Press, 1996.


Christof Paar and Jan Pelzl.
Understanding Cryptography, a Textbook for Students and Practitioners.
Springer, 2010.

(Two texbooks [PP10] (Paar and Pelzl) are available at the library at Lorient, code `005.8 PAA`).

Relvant chapters: 6, 7, 8, and 10.

During lockdown, the library is opened: see *La BU sur rendez-vous*
`https://www-actus.univ-ubs.fr/fr/index/actualites/scd/covid-19-la-bu-sur-rdv.html`

Lecture notes:

`https://gitlab.inria.fr/guillevi/enseignement/`

(`Lorient` → `Master-CSSE.md`)

Textbooks

The *Handbook* is available in PDF for free. Relevant chapters:

- chapter 3 `http://cacr.uwaterloo.ca/hac/about/chap3.pdf`
  - section 3.3 on the RSA problem,
  - section 3.6 on the discrete logarithm problem,
  - section 3.7 on the Diffie-Hellman problem,
- chapter 8 `http://cacr.uwaterloo.ca/hac/about/chap8.pdf`
  - section 8.2 on public-key cryptography and RSA,
  - section 8.4 on ElGamal encryption.
- chapter 11 `http://cacr.uwaterloo.ca/hac/about/chap11.pdf`
  - section 11.5.2 on ElGamal signature scheme,
  - section 11.5.1 on Digital Signature Algorithm.

Outline

Public-key encryption

Alice **Bob**

Public-key encryption

Alice

public parameters

Bob

public parameters

Public-key encryption

Alice

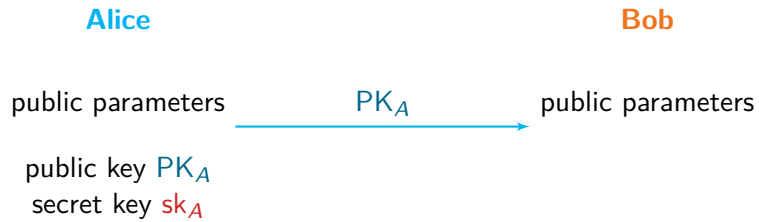
public parameters

public key PK_A
secret key sk_A

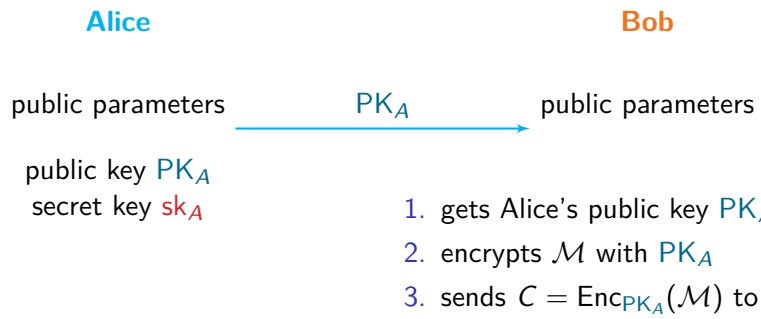
Bob

public parameters

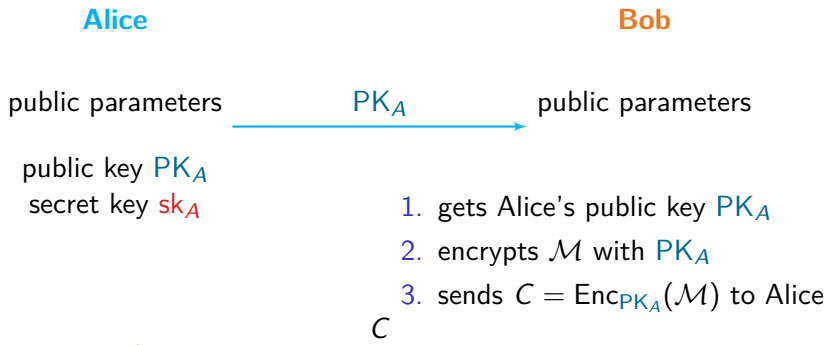
Public-key encryption



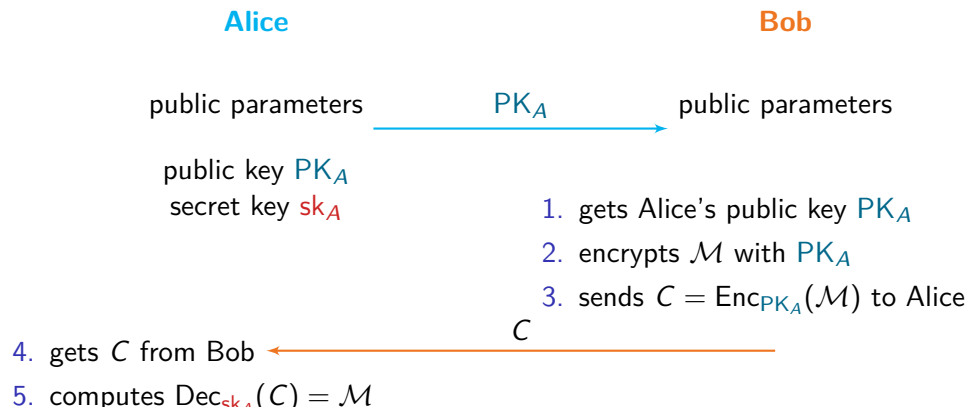
Public-key encryption



Public-key encryption



Public-key encryption



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?

Alice chooses two distinct primes p and q and set $N = pq$
Alice chooses a public exponent e coprime to N and $(p-1)(q-1)$
Alice computes $\varphi(N) = (p-1)(q-1)$ and $d = 1/e \bmod \varphi(N)$

public key: (N, e), e for *encryption*
private key: (p, q, d), d for *decryption*

Encryption: To send a secret message m to Alice: Bob

- obtains Alice's public key (N, e)
- encodes m as an integer between 0 and $N-1$,
- ciphertext $c = m^e \bmod N$, sends c to Alice.

Decryption: Alices computes $m = c^d \bmod N$.
It works because $m^{ed} = m \bmod N$, because $ed = 1 \bmod \varphi(N)$.

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), 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
```

Knowing the public and private exponents gives a factorization of N

Facts:

- 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, $x^{ed-1} \equiv x \bmod N$
- $ed - 1$ is even: $(ed-1)/2$ is integer

If N, e and d are known:
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: $2^{1932/4} = 988 \bmod N$, $\gcd(y-1, N) = 47 = q$, $\gcd(y+1, N) = 43 = p$.

Short private exponent is a bad idea

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

- $e = 3$
- $e = 2^{16} + 1 = 65537$

For a faster decryption, one could set a short private exponent d (and a large public exponent e instead).
It is wrong: Wiener attack.
Idea: continued fraction technique.

Padding

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

Choosing key sizes

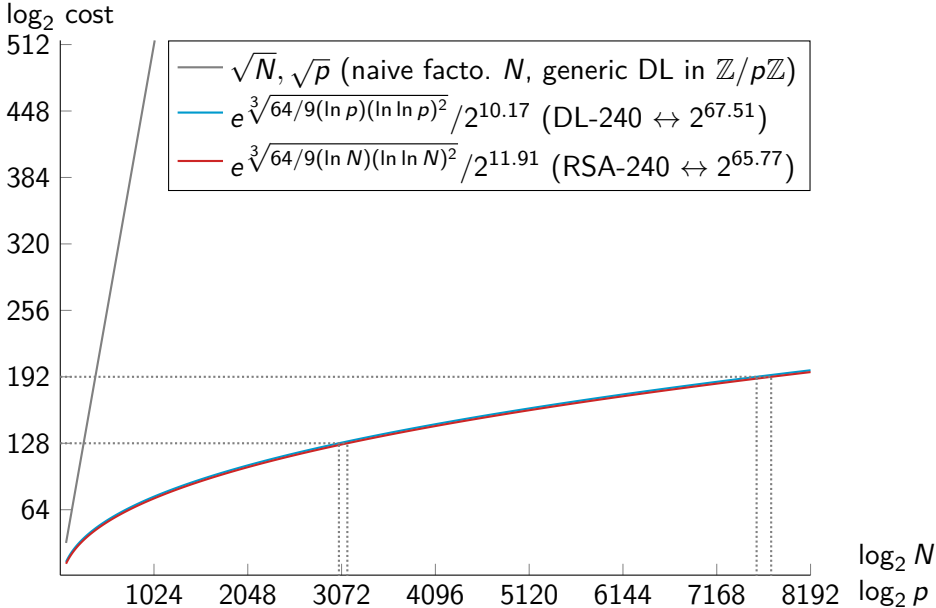
Symmetric ciphers (AES): key sizes are 128, 192 or 256 bits.
Perfect symmetric cipher: trying all keys of size n bits takes 2^n tests
$\rightarrow$ **brute-force search**

perfect symmetric cipher with secret key of 2^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.


log2 cost
512
448
384
320
256
192
128
64
√N, √p (naive facto. N, generic DL in ℤ/pℤ)
e∛(64/9(ln p)(ln ln p)²)/2^10.17 (DL-240 ↔ 2^67.51)
e∛(64/9(ln N)(ln ln N)²)/2^11.91 (RSA-240 ↔ 2^65.77)
1024
2048
3072
4096
5120
6144
7168
8192
log2 N
log2 p

Particles

n	2^n	Examples
32	$2^{32} = 10^{9.6}$	number of humans on Earth
46	$2^{46} = 10^{13.8}$	distance Earth - Sun in millimeters number of operations in one day on a processor at 1 GHz
55	$2^{55} = 10^{16.6}$	number of operations in one year on a processor at 1 GHz
82	$2^{82} = 10^{24.7}$	mass of Earth in kilogramms
90	$2^{90} = 10^{27.1}$	numer of operations in $15 \cdot 10^9$ years (age of the universe) on a processor at 1 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 16000 planets like the Earth

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):
$\gcd(988 - 1, 2021) = 47$, $\gcd(988 + 1, 2021) = 43$.
$N = 43 \times 47$

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.

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 $n = (a + m)^2 - N$

if n is B-smooth, store the relation $n = \prod_{p_j \text{ prime} \leq B} p_j^{e_j}$

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 $n = (a + m)^2 - N$

if n is B-smooth, store the relation $n = \prod_{p_j \text{ prime} \leq B} p_j^{e_j}$
Find a combination of n_i s.t.

$\prod_{i \in I} n_i = \prod_{p_k \text{ prime} \leq B} p_k^{e_k}$ and e_k even
$X = \prod_{i \in I} (a_i + m) \bmod N$, $Y = \sqrt{\prod_{i \in I} n_i} \bmod N$

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

$$
\begin{aligned}
(2 + m)^2 - N &= 95 = 5 \cdot 19 \\
(5 + m)^2 - N &= 380 = 2^2 \cdot 5 \cdot 19 \\
(17 + m)^2 - N &= 1700 = 2^2 \cdot 5^2 \cdot 17
\end{aligned}
$$

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

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

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

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

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

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

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

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

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

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

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

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

$$(\underbrace{46 \cdot 49}_{X})^2 \equiv (\underbrace{2 \cdot 5 \cdot 19}_{Y})^2 \bmod N$$

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: $n = \prod_{p_i \leq B} p_i^{e_i}$, p_i prime

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

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

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

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

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

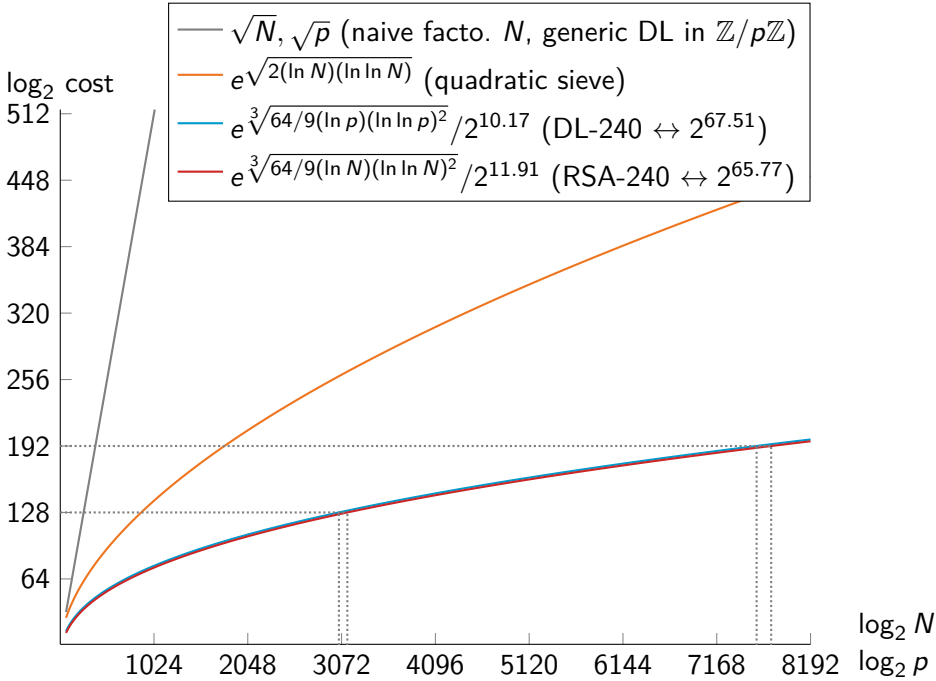
$\gcd(X - Y, N) = 43$, $\gcd(X + Y, N) = 47$

$N = 43 \cdot 47$

Quadratic Sieve: limitations for large numbers

Complexity: $e^{\sqrt{(2+o(1))\ln N \ln\ln N}}$

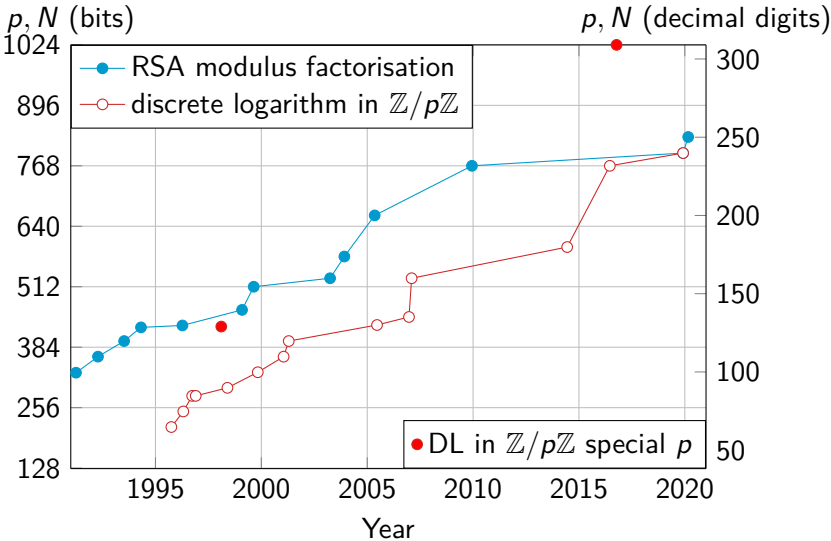
- $n = (a+m)^2 - N \approx 2A\sqrt{N}$
  Factor integers of size $\approx 2A\sqrt{N}$
- $\#\mathcal{F} = \#\{\text{ primes up to } B\} \approx B/\ln B$
- Computes left kernel of huge linear system modulo 2


$\sqrt{N}$, $\sqrt{p}$ (naive facto. N, generic DL in $\mathbb{Z}/p\mathbb{Z}$)
$e^{\sqrt{2(\ln N)(\ln\ln N)}}$ (quadratic sieve)
$e^{\sqrt[3]{64/9(\ln p)(\ln\ln p)^2}}/2^{10.17}$ (DL-240 $\leftrightarrow 2^{67.51}$)
$e^{\sqrt[3]{64/9(\ln N)(\ln\ln N)^2}}/2^{11.91}$ (RSA-240 $\leftrightarrow 2^{65.77}$)
$\log_2$ cost
512
448
384
320
256
192
128
64
1024
2048
3072
4096
5120
6144
7168
8192
$\log_2 N$
$\log_2 p$

Nowadays' method: the Number Field Sieve

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

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, editors, *CRYPTO 2020, Part II*, volume 12171 of *LNCS*, pages 62–91. Springer, Heidelberg, August 2020.

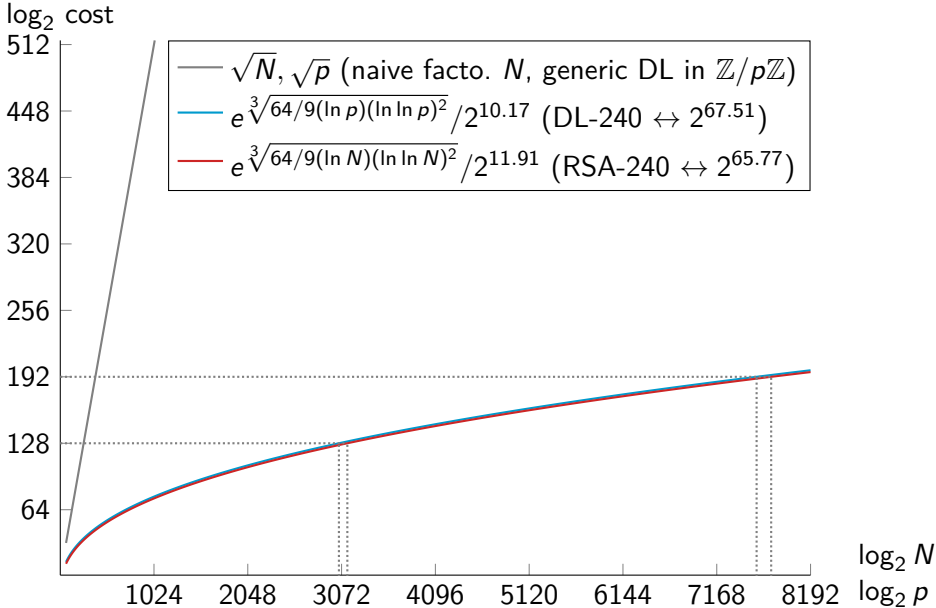
Factorization of RSA-240 (795 bits) in December 2019 and RSA-250 (829 bits) in February 2020

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

$$
\begin{aligned}
\text{RSA-250} &= 2140324650240744961264423072839333563008614715144755017797754920881418023447140136643345519095804679610992851872470914587687396261921557363047454770520805119056493106687691590019759405693457452230589325976697471681738069364894699871578494975937497937,\\
p &= 6413528947707158027879019017057738908482501474294344720811685963202453234463023862359875266834770873766192558569463979885336 7,\\
q &= 3337202759497815655622601060535511422794076034476755466678452098702384172921003708025744867329688187756571898625803693206271 1
\end{aligned}
$$


log2 cost
512
448
384
320
256
192
128
64
√N, √p (naive facto. N, generic DL in ℤ/pℤ)
e∛(64/9(ln p)(ln ln p)²)/2^10.17 (DL-240 ↔ 2^67.51)
e∛(64/9(ln N)(ln ln N)²)/2^11.91 (RSA-240 ↔ 2^65.77)
1024
2048
3072
4096
5120
6144
7168
8192
log2 N
log2 p

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)

$$N = 3129884719662540063950693863716193016278901146429595260054414582$$
$$93358495335288349178000889717657847571754913473200058603025745 23$$

1024 bits = 128 bytes but the key was 128 decimal digit long!
Factorization with `cado-nfs`

$$p = 462758347539951603789701738703986532996162069752028894871692 4853$$
$$q = 676354027172319302743451260512922936486939444439465602264176 9391.$$

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 modules) 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 = 0xc924249224929249924949244924242492249292499249492449242492249292499249
N = p * q
X = 2**168
a = 0xc924249224929249924949244924242492249292499249492449242492249292499249
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.

Outline

Discrete logarithm problem

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

Given $h \in \mathbf{G}$, find integer $x \in \{0, 1, \ldots, r-1\}$ such that $h = g^x$.
Exponentiation easy: $(g, x) \mapsto g^x$
Discrete logarithm hard in well-chosen groups **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 $E\colon y^2 = x^3 + ax + b/\mathbb{F}_p$

Diffie-Hellman key exchange

Alice **Bob**

Diffie-Hellman key exchange

Alice

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

public parameters

Bob

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

Diffie-Hellman key exchange

Alice

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

secret key $\mathsf{sk}_A = a \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$

public value $\mathsf{PK}_A = g^a$

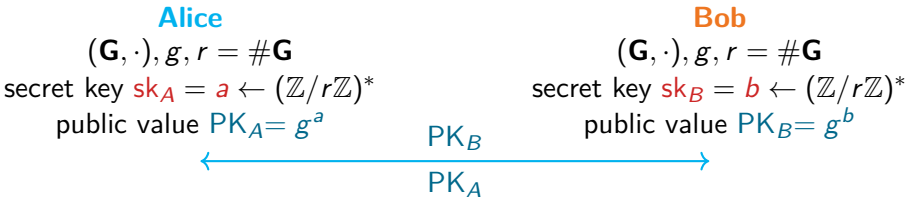
Bob

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

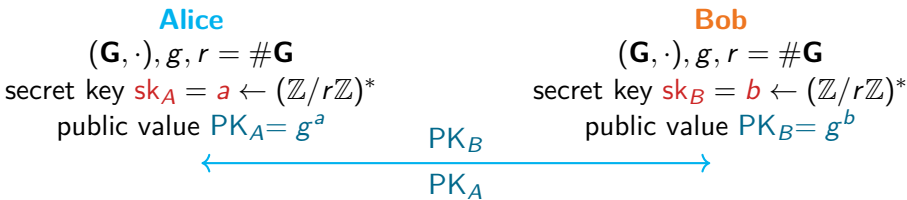
secret key $\mathsf{sk}_B = b \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$

public value $\mathsf{PK}_B = g^b$

Diffie-Hellman key exchange



Diffie-Hellman key exchange



ElGamal, Schnorr signature, DSA

ElGamal encryption

Alice **Bob**

ElGamal encryption

Alice		Bob
$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$	public parameters	$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

ElGamal encryption

Alice

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

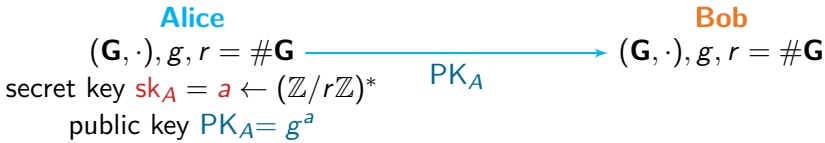
secret key $\mathsf{sk}_A = a \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$

public key $\mathsf{PK}_A = g^a$

Bob

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

ElGamal encryption



ElGamal encryption

1. gets Alice's public key PK_A
2. $\mathcal{M} \in \mathbf{G}$
3. $k_e \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$ at random
4. $\gamma = g^{k_e}$
5. $\mathsf{Enc}_{\mathsf{PK}_A}(\mathcal{M}) = \mathcal{M} \cdot \mathsf{PK}_A{}^{k_e} = \delta$
6. sends $C = (\gamma, \delta)$ to Alice

ElGamal encryption

Alice

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

secret key $\mathsf{sk}_A = a \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$

public key $\mathsf{PK}_A = g^a$

PK_A

Bob

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

Encryption

1. gets Alice's public key PK_A
2. $\mathcal{M} \in \mathbf{G}$
3. $k_e \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$ at random
4. $\gamma = g^{k_e}$
5. $\mathrm{Enc}_{\mathsf{PK}_A}(\mathcal{M}) = \mathcal{M} \cdot \mathsf{PK}_A{}^{k_e} = \delta$
6. sends $C = (\gamma, \delta)$ to Alice

C

ElGamal encryption

Alice

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

secret key $\mathsf{sk}_A = a \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$

public key $\mathsf{PK}_A = g^a$

$\xrightarrow{\mathsf{PK}_A}$

Bob

$(\mathbf{G}, \cdot), g, r = \#\mathbf{G}$

1. gets Alice's public key PK_A
2. $\mathcal{M} \in \mathbf{G}$
3. $k_e \leftarrow (\mathbb{Z}/r\mathbb{Z})^*$ at random
4. $\gamma = g^{k_e}$
5. $\mathsf{Enc}_{\mathsf{PK}_A}(\mathcal{M}) = \mathcal{M} \cdot {\mathsf{PK}_A}^{k_e} = \delta$
6. sends $C = (\gamma, \delta)$ to Alice

Decryption

$\xleftarrow{C}$

7. get $C = (\gamma, \delta)$ from Bob
8. $\mathsf{Dec}_{\mathsf{sk}_A}(C) = (\gamma^{-a}) \cdot \delta = \mathcal{M}$

Asymmetric cryptography

Factorization (RSA cryptosystem)

Discrete logarithm problem (use in Diffie-Hellman, etc)

Given a finite cyclic group $(\mathbf{G}, \cdot)$, a generator g and $h \in \mathbf{G}$, compute x s.t. $h = g^x$.
$\rightarrow$ can we invert the exponentiation function $(g, x) \mapsto g^x$?

Common choice of $\mathbf{G}$:

- prime finite field $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ (1976)
- characteristic 2 field $\mathbb{F}_{2^n}$ ($\approx$ 1979)
- elliptic curve $E(\mathbb{F}_p)$ (1985)

Discrete log problem

How fast can we invert the exponentiation function $(g, x) \mapsto g^x$?

- $g \in G$ generator, $\exists$ always a preimage $x \in \{1, \ldots, \#G\}$
- naive search, try them all: $\#G$ tests
- $O(\sqrt{\#G})$ generic algorithms

Discrete log problem

How fast can we invert the exponentiation function $(g, x) \mapsto g^x$?

- $g \in G$ generator, $\exists$ always a preimage $x \in \{1, \ldots, \#G\}$
- naive search, try them all: $\#G$ tests
- $O(\sqrt{\#G})$ generic algorithms
  - Shanks baby-step-giant-step (BSGS): $O(\sqrt{\#G})$, deterministic
  - random walk in G, cycle path finding algorithm in a connected graph (Floyd) $\rightarrow$ Pollard: $O(\sqrt{\#G})$, probabilistic
    (the cycle path encodes the answer)
  - parallel search (parallel Pollard, Kangarous)

Discrete log problem

How fast can we invert the exponentiation function $(g, x) \mapsto g^x$?

- $g \in G$ generator, $\exists$ always a preimage $x \in \{1, \ldots, \#G\}$
- naive search, try them all: $\#G$ tests
- $O(\sqrt{\#G})$ generic algorithms
  - Shanks baby-step-giant-step (BSGS): $O(\sqrt{\#G})$, deterministic
  - random walk in G, cycle path finding algorithm in a connected graph (Floyd) $\rightarrow$ Pollard: $O(\sqrt{\#G})$, probabilistic
    (the cycle path encodes the answer)
  - parallel search (parallel Pollard, Kangarous)
- independent search in each distinct subgroup
  + Chinese remainder theorem (Pohlig-Hellman)

Discrete log problem

How fast can we invert the exponentiation function $(g, x) \mapsto g^x$?

- $\rightarrow$ choose G of large prime order (no subgroup)
- $\rightarrow$ complexity of inverting exponentiation in $O(\sqrt{\#G})$
- $\rightarrow$ security level 128 bits means $\sqrt{\#G} \geq 2^{128}$
  take $\#G = 2^{256}$
  analogy with symmetric crypto, keylength 128 bits (16 bytes)

Discrete log problem

How fast can we invert the exponentiation function $(g, x) \mapsto g^x$?

- $\rightarrow$ choose G of large prime order (no subgroup)
- $\rightarrow$ complexity of inverting exponentiation in $O(\sqrt{\#G})$
- $\rightarrow$ security level 128 bits means $\sqrt{\#G} \geq 2^{128}$
  take $\#G = 2^{256}$
  analogy with symmetric crypto, keylength 128 bits (16 bytes)

Use additional structure of G if any.

Discrete log problem when $\mathbf{G} = (\mathbb{Z}/p\mathbb{Z})^*$

Index calculus algorithm [Western–Miller 68, Adleman 79],
prequel of the Number Field Sieve algorithm (NFS)

- p prime, $(p-1)/2$ prime, $\mathbf{G} = (\mathbb{Z}/p\mathbb{Z})^*$, gen. g, target h
- get many multiplicative relations in $\mathbf{G}$
  $g^t = g_1^{e_1} g_2^{e_2} \cdots g_i^{e_i} \pmod{p}$, $g, g_1, g_2, \ldots, g_i \in \mathbf{G}$
- find a relation $h \cdot g^s = g_1^{e'_1} g_2^{e'_2} \cdots g_i^{e'_i} \pmod{p}$
- take logarithm: linear relations

$$\begin{aligned} t &= e_1 \log g_1 + e_2 \log g_2 + \ldots + e_i \log g_i \pmod{p-1} \\ &\vdots \\ \log h &= -s + e'_1 \log g_1 + e'_2 \log g_2 + \ldots + e'_i \log g_i \pmod{p-1} \end{aligned}$$

- solve a linear system
- get $x = \log h$

Index calculus in $(\mathbb{Z}/p\mathbb{Z})^*$: example

$p = 1109$, $r = (p-1)/4 = 277$ prime
Smoothness bound $B = 13$
$\mathcal{F}_{13} = \{2, 3, 5, 7, 11, 13\}$ small primes up to B, $i = \#\mathcal{F}$
B-smooth integer: $n = \prod_{p_i \le B} p_i^{e_i}$, p_i prime
is g^s smooth? $1 \le s \le 72$ is enough

$$
\begin{array}{ll}
g^{1} = & 2 = 2 \\
g^{13} = & 429 = 3 \cdot 11 \cdot 13 \\
g^{16} = & 105 = 3 \cdot 5 \cdot 7 \\
g^{21} = & 33 = 3 \cdot 11 \\
g^{44} = & 1029 = 3 \cdot 7^3 \\
g^{72} = & 325 = 5^2 \cdot 13
\end{array}
\quad \rightarrow \quad
\begin{array}{c}
\begin{matrix} 2 & 3 & 5 & 7 & 11 & 13 \end{matrix} \\
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 3 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 & 1
\end{bmatrix}
\end{array}
\cdot \boldsymbol{x} =
\begin{bmatrix} 1 \\ 13 \\ 16 \\ 21 \\ 44 \\ 72 \end{bmatrix}
$$

$\boldsymbol{x} = [1, 219, 40, 34, 79, 269] \bmod 277$

$\rightarrow \log_g 7 = 34 \bmod 277$, that is, $(g^{34})^4 = 7^4$
$g^{34} = 7u$ and $u^4 = 1$

Index calculus in $(\mathbb{Z}/p\mathbb{Z})^*$: example

$\boldsymbol{x} = [1, 219, 40, 34, 79, 269] \bmod 277$
subgroup of order 4: $g_4 = g^{(p-1)/4}$
$\{1, g_4, g_4^2, g_4^3\} = \{1, 354, 1108, 755\}$

Pohlig-Hellman:

$$
\begin{array}{rrrrrrrrr}
3/g^{219} = & 1 = & 1 \Rightarrow \log_g 3 & = & & = & 219 \\
5/g^{40} = & 1108 = & -1 \Rightarrow \log_g 5 & = & 40 + (p-1)/2 = & & 594 \\
7/g^{34} = & 354 = & g_4 \Rightarrow \log_g 7 & = & 34 + (p-1)/4 = & & 311 \\
11/g^{79} = & 755 = & g_4^3 \Rightarrow \log_g 11 = & & 79 + 3(p-1)/4 = & & 910 \\
13/g^{269} = & 755 = & g_4^3 \Rightarrow \log_g 13 = & & 269 + 3(p-1)/4 = & & 1100
\end{array}
$$

$\boldsymbol{v} = [1, 219, 594, 311, 910, 1100] \bmod p - 1$

Target $h = 777$
$g^{10} \cdot 777 = 495 = 3^2 \cdot 5 \cdot 11 \bmod p$
$\log_2 777 = -10 + 2\log_g 3 + \log_g 5 + \log_g 11 = 824 \bmod p - 1$
$g^{824} = 777$

Index calculus in $(\mathbb{Z}/p\mathbb{Z})^*$: example

Trick

Multiplicative relations over the **integers**

$g_1, g_2, \ldots, g_i \longleftrightarrow$ small prime integers

Smooth integers $n = \prod_{p_i \leq B} p_i^{e_i}$ are quite common $\rightarrow$ it works Complexity $e^{\sqrt{(2+o(1))(\log p)(\log\log p)}}$ (Pomerance 87)

Index calculus in $(\mathbb{Z}/p\mathbb{Z})^*$: example

Trick

Multiplicative relations over the **integers**

$g_1, g_2, \ldots, g_i \longleftrightarrow$ small prime integers

Smooth integers $n = \prod_{p_i \leq B} p_i^{e_i}$ are quite common $\rightarrow$ it works Complexity $e^{\sqrt{(2+o(1))(\log p)(\log\log p)}}$ (Pomerance 87)

Improvements in the 80's, 90's:

- Sieve (faster relation collection)
- Smaller integers to factor
- Multiplicative relations in **number fields**
- Better **sparse linear algebra**
- Independent targets h

Number Field: Toy example with $\mathbb{Z}[i]$

1986: Coppersmith–Odlyzko–Schroeppel, DL in GF(p)

If $p = 1 \bmod 4$, $\exists\ U, V$ s.t. $p = U^2 + V^2$
and $|U|, |V| < \sqrt{p}$
$U/V \equiv m \mod p$ and $m^2 + 1 = 0 \mod p$

Define a map from $\mathbb{Z}[i]$ to $\mathbb{Z}/p\mathbb{Z}$

$$\begin{array}{rcl} \phi\colon \mathbb{Z}[i] & \to & \mathbb{Z}/p\mathbb{Z} \\ i & \mapsto & m \bmod p \text{ where } m = U/V,\ m^2 + 1 = 0 \bmod p \end{array}$$

ring homomorphism $\phi(a + bi) = a + bm$

$$\phi(\underbrace{a + bi}_{\text{factor in } \mathbb{Z}[i]}) = a + bm = (a + b\ \underbrace{U/V}_{=m}) = (\underbrace{aV + bU}_{\text{factor in } \mathbb{Z}})V^{-1} \mod p$$

Example in $\mathbb{Z}[i]$

$p = 1109 = 1 \bmod 4$, $r = (p-1)/4 = 277$ prime
$p = 22^2 + 25^2$
$\max(|a|, |b|) = A = 20$, $B = 13$ smoothness bound

Example in $\mathbb{Z}[i]$

$p = 1109 = 1 \bmod 4$, $r = (p-1)/4 = 277$ prime
$p = 22^2 + 25^2$
$\max(|a|, |b|) = A = 20$, $B = 13$ smoothness bound

Rational side

$\mathcal{F}_{\text{rat}} = \{2, 3, 5, 7, 11, 13\}$ primes up to B
$g(x) = Vx - U$

Example in $\mathbb{Z}[i]$

$p = 1109 = 1 \bmod 4$, $r = (p-1)/4 = 277$ prime
$p = 22^2 + 25^2$
$\max(|a|, |b|) = A = 20$, $B = 13$ smoothness bound

Rational side

$\mathcal{F}_{\text{rat}} = \{2, 3, 5, 7, 11, 13\}$ primes up to B
$g(x) = Vx - U$

Algebraic side: think about the complex number in $\mathbb{C}$

$-i(1+i)^2 = 2$, $(2+i)(2-i) = 5$, $(2+3i)(2-3i) = 13$
$\mathcal{F}_{\text{alg}} = \{1+i, 2+i, 2-i, 2+3i, 2-3i\}$
"primes" of norm up to B
$f(x) = x^2 + 1$

Example in $\mathbb{Z}[i]$

$p = 1109 = 1 \bmod 4$, $r = (p-1)/4 = 277$ prime
$p = 22^2 + 25^2$
$\max(|a|, |b|) = A = 20$, $B = 13$ smoothness bound

Rational side

$\mathcal{F}_{\text{rat}} = \{2, 3, 5, 7, 11, 13\}$ primes up to B
$g(x) = Vx - U$

Algebraic side: think about the complex number in $\mathbb{C}$

$-i(1+i)^2 = 2$, $(2+i)(2-i) = 5$, $(2+3i)(2-3i) = 13$
$\mathcal{F}_{\text{alg}} = \{1+i, 2+i, 2-i, 2+3i, 2-3i\}$
"primes" of norm up to B
$f(x) = x^2 + 1$

Units

$\mathcal{U}_{\text{alg}} = \{-1, i, -i\}$

Example in $\mathbb{Z}[i]$

$p = 1109$

$(a, b) = (-4, 7)$,
$\text{Norm}(-4 + 7i) = (-4)^2 + 7^2 = 65 = 5 \cdot 13$

In $\mathbb{Z}[i]$,

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

Example in $\mathbb{Z}[i]$

$p = 1109$

$(a, b) = (-4, 7)$,
$\text{Norm}(-4 + 7i) = (-4)^2 + 7^2 = 65 = 5 \cdot 13$

In $\mathbb{Z}[i]$,

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

$\rightarrow$ $(2 \pm i)(2 \pm 3i)$ has norm 65
$\rightarrow$ $\pm i(2 \pm i)(2 \pm 3i) = (-4 + 7i)$

We obtain $i(2 - i)(2 + 3i) = -4 + 7i$

Example in $\mathbb{Z}[i]$

$p = 1109$

$(a, b) = (-4, 7)$,
$\text{Norm}(-4 + 7i) = (-4)^2 + 7^2 = 65 = 5 \cdot 13$

In $\mathbb{Z}[i]$,

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

$\rightarrow$ $(2 \pm i)(2 \pm 3i)$ has norm 65
$\rightarrow$ $\pm i(2 \pm i)(2 \pm 3i) = (-4 + 7i)$

We obtain $i(2 - i)(2 + 3i) = -4 + 7i$

$i \leftrightarrow m = 22/25 = 755 \bmod p$
$m(2 - m)(2 + 3m) = 845 \bmod p$
$-4 + 7m = 845 \bmod p$
$(-4 \cdot 25 + 7 \cdot 22)/25 = 845 \bmod p$

Example in $\mathbb{Z}[i]$

$a+bi$	$aV+bU=$ factor in $\mathbb{Z}$	a^2+b^2	factor in $\mathbb{Z}[i]$
$-17+19i$	$-7=-7$	$650=2\cdot 5^2\cdot 13$	$i(1+i)(2+i)^2(2-3i)$
$-11+2i$	$-231=-3\cdot 7\cdot 11$	$125=5^3$	$i(2+i)^3$
$-6+17i$	$224=2^5\cdot 7$	$325=5^2\cdot 13$	$(2+i)^2(2+3i)$
$-4+7i$	$54=2\cdot 3^3$	$65=5\cdot 13$	$i(2-i)(2+3i)$
$-3+4i$	$13=13$	$25=5^2$	$-(2-i)^2$
$-2+i$	$-28=-2^2\cdot 7$	$5=5$	$-(2-i)$
$-2+3i$	$16=2^4$	$13=13$	$-(2-3i)$
$-2+11i$	$192=2^6\cdot 3$	$125=5^3$	$-(2-i)^3$
$-1+i$	$-3=-3$	$2=2$	$i(1+i)$
i	$22=2\cdot 11$	$1=1$	i
$1+3i$	$91=7\cdot 13$	$10=2\cdot 5$	$(1+i)(2+i)$
$1+5i$	$135=3^3\cdot 5$	$26=2\cdot 13$	$i(1+i)(2-3i)$
$2+i$	$72=2^3\cdot 3^2$	$5=5$	$(2+i)$
$5+i$	$147=3\cdot 7^2$	$26=2\cdot 13$	$-i(1+i)(2+3i)$

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

Build the matrix of relations:

- one row per (a, b) pair s.t. both norms are smooth
- one column per prime of $\mathcal{F}_{\text{rat}}$
- one column for $1/V$
- one column per prime ideal of $\mathcal{F}_{\text{alg}}$
- one column per unit $(-1, i)$
- store the exponents

$$
M = \begin{array}{c}
\begin{array}{cccccccccccccc}
2 & 3 & 5 & 7 & 11 & 13 & \frac{1}{V} & -1 & i & 1+i & 2+i & 2-i & 2+3i & 2-3i
\end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 3 & 0 & 0 & 0 \\
5 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 1 & 0 \\
1 & 3 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 2 & 0 & 0 \\
2 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
4 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
6 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 3 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 3 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
3 & 2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 2 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

$$
M = \begin{array}{c}
\begin{array}{ccccccccccccccc}
2 & 3 & 5 & 7 & 11 & 13 & \frac{1}{V} & -1 & i & 1+i & 2+i & 2-i & 2+3i & 2-3i
\end{array} \\
\left[\begin{array}{ccccccccccccccc}
 & & & & & & & 1 & 2 & & & & & \\
 & & & 1 & & & 1 & 1 & 1 & 1 & 2 & & & 1 \\
 & 1 & & 1 & 1 & & 1 & 1 & 1 & & 3 & & & \\
5 & & & 1 & & & 1 & & & & 2 & & 1 & \\
1 & 3 & & & & & 1 & & 1 & & & 1 & 1 & \\
 & & & & & 1 & 1 & 1 & & & & 2 & & \\
2 & & & 1 & & & 1 & & & & & 1 & & \\
4 & & & & & & 1 & 1 & & & & & & 1 \\
6 & 1 & & & & & 1 & 1 & & & & 3 & & \\
 & 1 & & & & & 1 & 1 & 1 & 1 & & & & \\
1 & & & & 1 & & 1 & & 1 & & & & & \\
 & & & 1 & & 1 & 1 & & & 1 & 1 & & & \\
 & 3 & 1 & & & & 1 & & 1 & 1 & & & & 1 \\
3 & 2 & & & & & 1 & & & & 1 & & & \\
 & 1 & & 2 & & & 1 & 1 & 1 & 1 & & & 1 &
\end{array}\right]
\end{array}
$$

$$
M = \begin{array}{c}
\begin{array}{cccccccccccccc}
2 & 3 & 5 & 7 & 11 & 13 & \frac{1}{V} & -1 & i & 1+i & 2+i & 2-i & 2+3i & 2-3i
\end{array} \\
\left[\begin{array}{cccccccccccccc}
 & & & & & & & -1 & -2 & & & & & \\
 & & & 1 & & & 1 & -1 & -1 & -1 & -2 & & & -1 \\
 & 1 & & 1 & 1 & & 1 & -1 & -1 & & -3 & & & \\
5 & & & 1 & & & 1 & & & & -2 & & -1 & \\
1 & 3 & & & & & 1 & & -1 & & & -1 & -1 & \\
 & & & & & 1 & 1 & -1 & & & & -2 & & \\
2 & & & 1 & & & 1 & & & & & -1 & & \\
4 & & & & & & 1 & -1 & & & & & & -1 \\
6 & 1 & & & & & 1 & -1 & & & & -3 & & \\
 & 1 & & & & & 1 & -1 & -1 & -1 & & & & \\
1 & & & & 1 & & 1 & & -1 & & & & & \\
 & & & 1 & & 1 & 1 & & & -1 & -1 & & & \\
 & 3 & 1 & & & & 1 & & -1 & -1 & & & & -1 \\
3 & 2 & & & & & 1 & & & & -1 & & & \\
 & 1 & & 2 & & & 1 & -1 & -1 & -1 & & & -1 &
\end{array}\right]
\end{array}
$$

Example in $\mathbb{Z}[i]$

Right kernel $M \cdot \boldsymbol{x} = 0 \bmod (p-1)/4 = 277$:

$$\boldsymbol{x} = (\underbrace{1, 219, 40, 34, 79, 269}_{\text{rational side}}, \underbrace{197}_{1/V}, \underbrace{0, 0}_{\text{units}}, \underbrace{139, 84, 233, 68, 201}_{\text{algebraic side}})$$

Logarithms (in some basis)

Example in $\mathbb{Z}[i]$

Right kernel $M \cdot \boldsymbol{x} = 0 \bmod (p-1)/4 = 277$:

$$\boldsymbol{x} = (\underbrace{1, 219, 40, 34, 79, 269}_{\text{rational side}}, \underbrace{197}_{1/V}, \underbrace{0, 0}_{\text{units}}, \underbrace{139, 84, 233, 68, 201}_{\text{algebraic side}})$$

Logarithms (in some basis)

Rational side: logarithms of $\{2, 3, 5, 7, 11, 13\}$ in basis 2

$\boldsymbol{x} = [1, 219, 40, 34, 79, 269] \bmod 277$

$\rightarrow$ order 4 subgroup

$\boldsymbol{v} = [1, 219, 594, 311, 910, 1100] \bmod p - 1$

Example in $\mathbb{Z}[i]$

Right kernel $M \cdot \boldsymbol{x} = 0 \bmod (p-1)/4 = 277$:

$$\boldsymbol{x} = (\underbrace{1, 219, 40, 34, 79, 269}_{\text{rational side}}, \underbrace{197}_{1/V}, \underbrace{0, 0}_{\text{units}}, \underbrace{139, 84, 233, 68, 201}_{\text{algebraic side}})$$

Logarithms (in some basis)

Rational side: logarithms of $\{2, 3, 5, 7, 11, 13\}$ in basis 2

$\boldsymbol{x} = [1, 219, 40, 34, 79, 269] \bmod 277$

$\rightarrow$ order 4 subgroup

$\boldsymbol{v} = [1, 219, 594, 311, 910, 1100] \bmod p - 1$

Target 314, generator $g = 2$

$314 = -20/7 \bmod p = -2^2 \cdot 5/7$

$$\begin{aligned} \log_g 314 &= \log_g -1 + 2\log_g 2 + \log_g 5 - \log_g 7 \\ &= (p-1)/2 + 2 + 594 - 311 = 839 \bmod p - 1 \end{aligned}$$

$2^{839} = 314 \bmod p$

Number Field Sieve

Since 1993 (Gordon, Schirokauer):

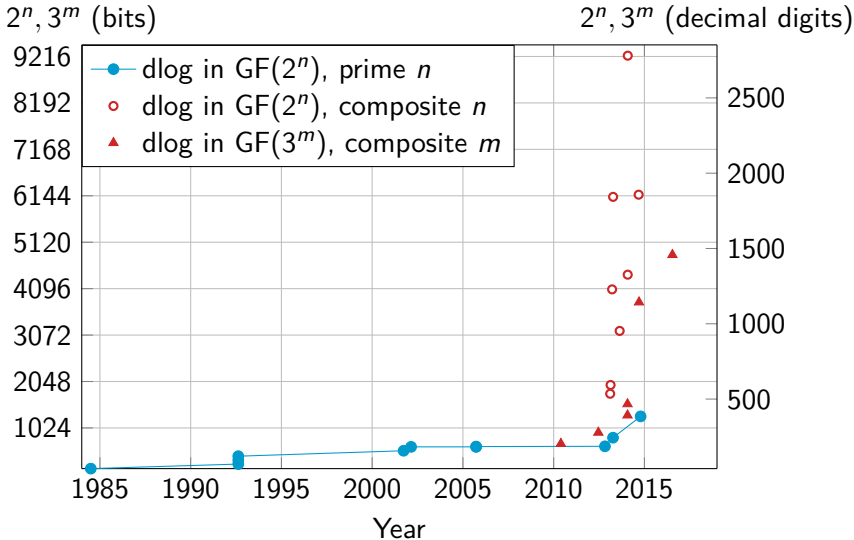
$$L_p(1/3, c) = e^{(c+o(1))(\log p)^{1/3}(\log\log p)^{2/3}}$$

- polynomial selection
- **relation collection** $L_p(1/3, 1.923)$
  sieve to enumerate efficiently (a, b) pairs
- **sparse linear algebra** $L_p(1/3, 1.923)$
  compute right kernel mod prime ℓ, block-Wiedemann alg.
- individual discrete logarithm

Attacks on discrete-logarithm based cryptosystems

1. Sony Play-Station 3 (PS3) hacking
   1.1 ECDSA signature
   1.2 PS3 problem
2. Weak DH attack
3. Weak keys in the Moscow internet voting system

Discrete logarithm computation in finite fields $\mathbb{F}_{2^n}$ and $\mathbb{F}_{3^m}$



Outline

What is a pairing?

$(\mathbf{G}_1, +), (\mathbf{G}_2, +), (\mathbf{G}_3, \cdot)$ three cyclic groups of order r
Pairing: map $e : \mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_3$

1. bilinear: $e(P_1 + P_2, Q) = e(P_1, Q) \cdot e(P_2, Q)$, $e(P, Q_1 + Q_2) = e(P, Q_1) \cdot e(P, Q_2)$
2. non-degenerate: $e(G_1, G_2) \neq 1$ for $\langle G_1 \rangle = \mathbf{G}_1$, $\langle G_2 \rangle = \mathbf{G}_2$
3. efficiently computable.

In practice we use mostly

$$e([a]P, [b]Q) = e([b]P, [a]Q) = e(P, Q)^{ab} \ .$$

$\rightsquigarrow$ Many applications in asymmetric cryptography.

Pairings in cryptography: 1993 and 2001

1993

Menezes–Okamoto–Vanstone attack

2001

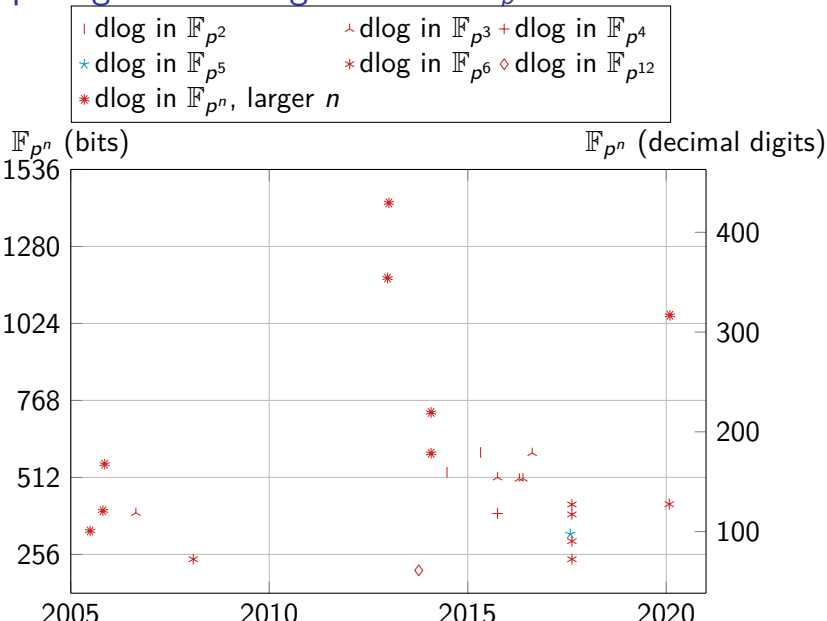
- Joux' tri-partite key exchange
- Boneh Frankin Identity based encryption
- Boneh Lynn Shacham short signature

Pairings with curves over fields $\mathbb{F}_{2^n}$ and $\mathbb{F}_{3^m}$, rise and fall

Pairings with curves over fields $\mathbb{F}_p$

`https://members.loria.fr/AGuillevic/pairing-friendly-curves/`

Computing Discrete logarithms in $\mathbb{F}_{p^n}$



Choosing key-sizes

`https://members.loria.fr/AGuillevic/pairing-friendly-curves/`