elliptikus gorbek

stf

<2020-09-12 Sat>

csoport (group)

halmaz , amin van 1 muvelet ertelmezve ami teljesiti ezeket az axiomakat:

- zart: a + b = c, ahol c szinten resze a halmaznak.
- asszociativ: (a + b) + c = a + (b + c).
- egysegelem: 0 + a = a + 0 = a
- inverzelem: a + b = b + a = 0 ahol b = -a

Pl egesz szamok halmaza es az osszeadas.

abelian csoport

a muvelet kommutativ: $a + b = b + a$

Weierstrass

$$y^2 = x^3 + ax + b$$

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

- point at infinity:

$$O = [0 : 1 : 0]$$

$$O = -O$$

$$P + O = P$$

$$P - P = O$$

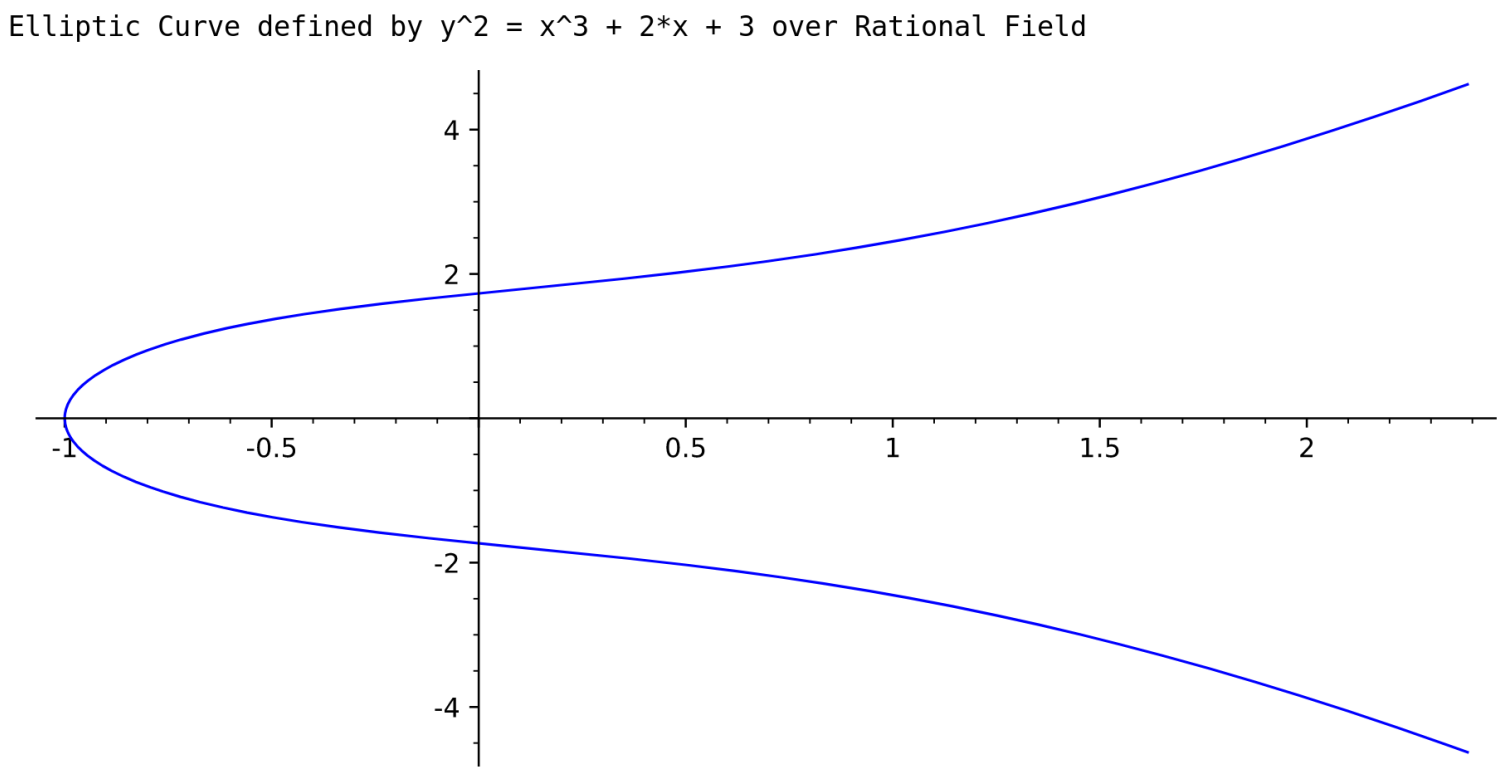
- van olyan Weierstrass amit Montgomeryva lehet konvertalni, pl secp256k1 **nem**

sage:Weierstrass

```
E0 = EllipticCurve([2,3])
latex(E0)
```

$$y^2 = x^3 + 2x + 3$$

```
E0; plot(E0)
```



Szimmetria

a gorbek szimmetrikusak "az" x tengelyre

veges test (finite field)

veges elemu halmaz, amin +-*/ muveletek ertelmezve vannak.
legegyszerubb vegesse tevo muvelet a mod p

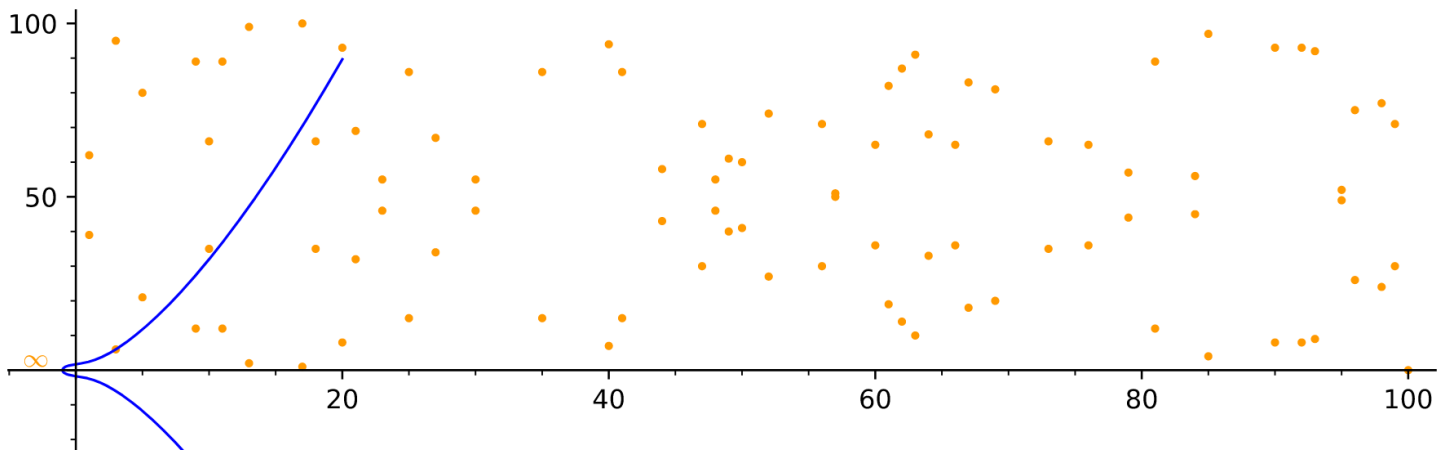
sage: Curve over Finite Field

```
E1 = EllipticCurve(GF(101),[5,1])
latex(E1)
```

$$y^2 = x^3 + 5x + 1$$

```
g0 = E1.gens()[0]
pts = [ g0 * x for x in range(g0.order()) ]
plots = [ plot(p, hue = 0.1) for p in pts ]
plots.append(plot(E0,xmax=20))
E1; (g0, g0.order()); plot(sum(plots))
```

```
Elliptic Curve defined by y^2 = x^3 + 2*x + 3 over Finite Field of size 101
((17 : 100 : 1), 96)
```



NIST P-256

$$y^2 = x^3 + ax + b$$

```
a=-3
b=41058363725152142129326129780047268409114441015993725554835256314039467401291
p = 0xffffffff00000001000000000000000000000000ffffffffffffffffffffffff
 = 2^256 - 2^224 + 2^192 + 2^96 - 1
```

25519-es Weierstrass gorbe

```
F2 = GF((2^255)-19)
latex(F2)
```

$$F_{36185027886661311069865932815214971204146870208012676262330495002472853012 39}$$

```
c25519 = EllipticCurve(F2, [0, 486662, 0, 1, 0])
P25519 = c25519.point([9,F2(9^3 + 486662*9^2 + 9).sqrt()])
print(c25519)
```

Elliptic Curve defined by

$$y^2 = x^3 + 486662 * x^2 + x$$

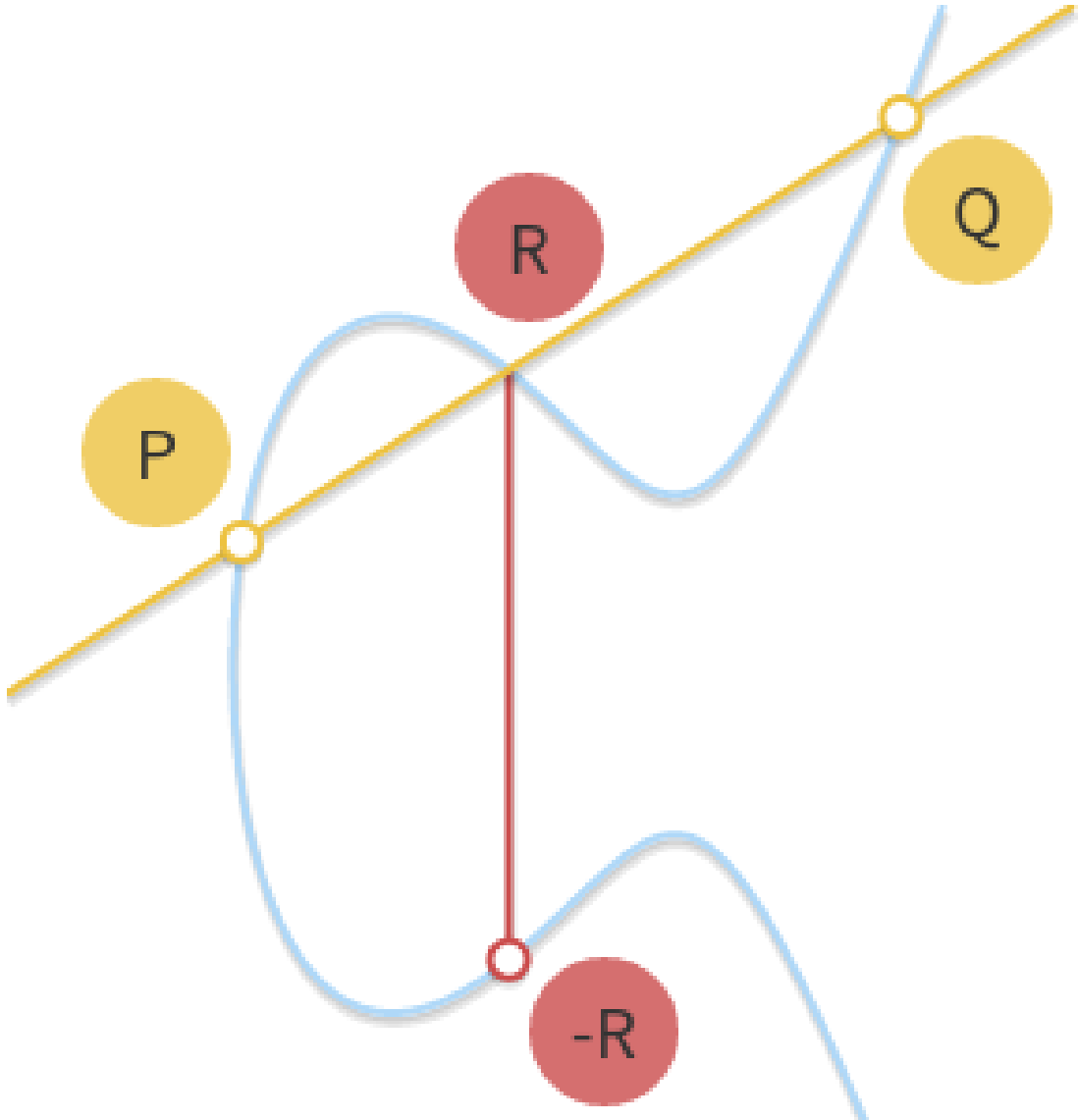
over Finite Field of size

57896044618658097711785492504343953926634992332820282019728792003956564819949

```
print("basepoint", P25519)
```

basepoint (9 : 14781619447589544791020593568409986887264606134616475288964881837755586237401 : 1)

P + Q = R (gfx)



P + Q = R (math)

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}$$

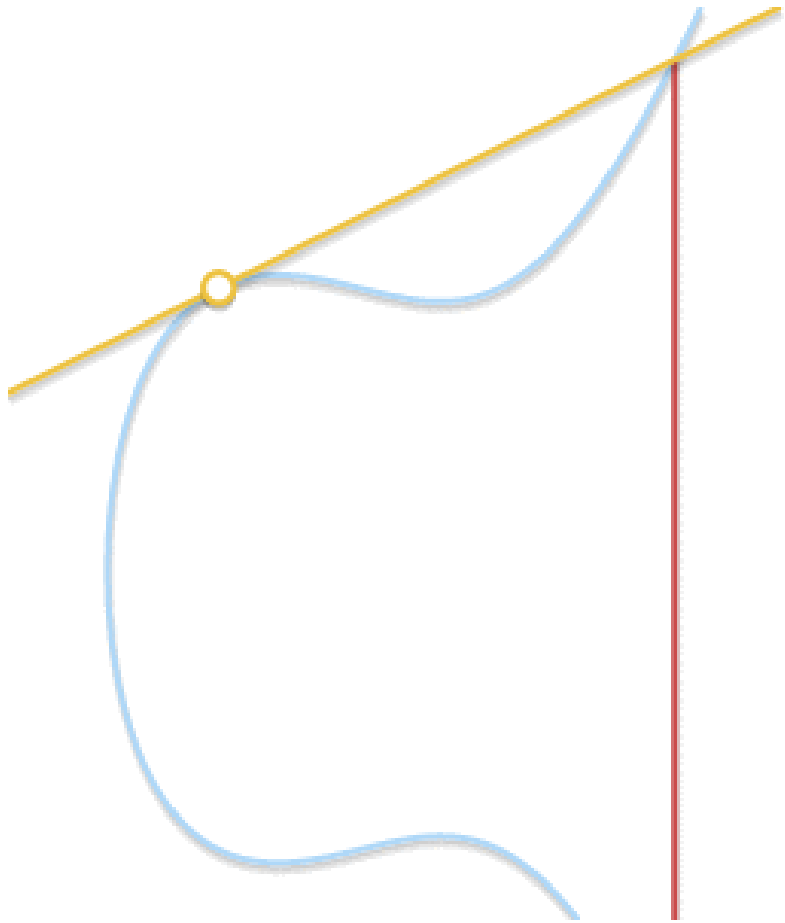
$$x_R = \lambda^2 - x_P - x_Q$$

$$y_R = \lambda(x_P - x_R) - y_P$$

P = Q?

$$P == Q$$

$$P + Q = 2P = R$$



2*P = R (math)

$$\lambda = \frac{3x_P^2 + a}{2y_P}$$

$$x_R = \lambda^2 - x_P - x_Q$$

$$y_R = \lambda * (x_P - x_R) - y_P$$

skalaris szorzas

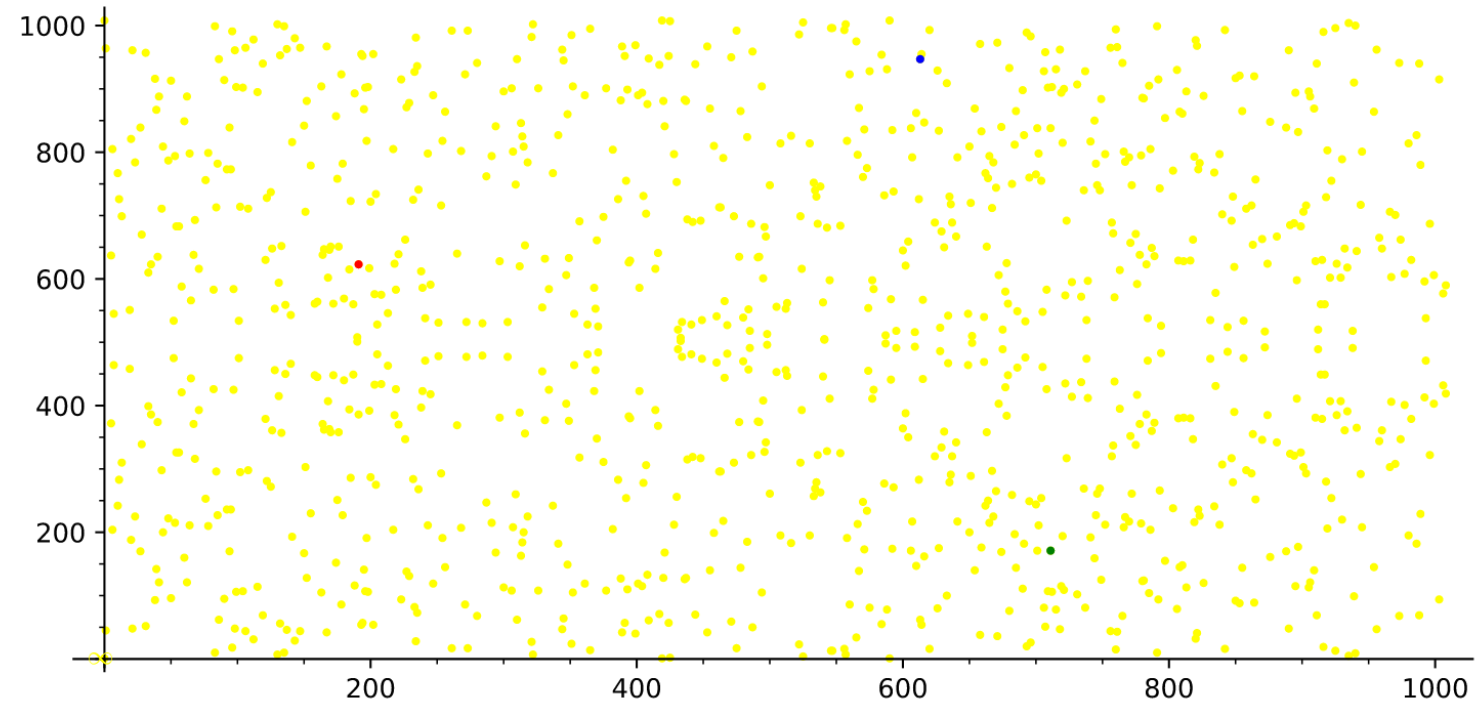
$$n * P$$

ECDH - toy curve

```
E = EllipticCurve(GF(1009),[5,1]); E
Elliptic Curve defined by y^2 = x^3 + 5*x + 1
  over Finite Field of size 1009

g = E.gens()[0]; (g, g.order())
((141 : 193 : 1), 1039)

rA = randrange(g.order()); RA = rA*g; (rA, RA)
(613, (692 : 533 : 1))

rB = randrange(g.order()); RB = rB*g; (rB, RB)
(562, (674 : 840 : 1))

kA = rA*RB; kA
(838 : 797 : 1)

kB = rB*RA; kB
(838 : 797 : 1)
```

ECDH - toy curve (gfx)

```
Elliptic Curve defined by y^2 = x^3 + 5*x + 1 over Finite Field of size 1009
generator: (553 : 684 : 1), order: 1039
rA = random(g.order()) = 242; RA = rA*g = (613 : 947 : 1)
rB = random(g.order()) = 299; RB = rB*g = (191 : 623 : 1)
kA = rA*RB = %s, kB = rb*RA = %s (711 : 171 : 1) (711 : 171 : 1)
```



ECDSA - toy curve

```
h = sha256("message")

# ephemeral key
k = randrange(g.order())
K = k*g
# t = the x coordinate of K
t = ZZ(K[0])
# rA = privkey
s = (h + rA*t)/k % g.order()
```

ECDSA verify(h, t, s, RA)

```
w = (1/s) % g.order()
u = (w*h) % g.order()
v = (w*t) % g.order()
F = u*g
H = v*RA
Q = F + H

return Q == K
```

ECDH - x25519

```
secret = randrange(P25519.order())
print('secret', secret)
public = P25519 * secret
print('public', public)
```

```
secret 24745173897881478126527009404969330899694166002010714916584519598331897911111
public (33847521402654535275984173222656606429154647117417863807654818999207209807879 :
   18850501614000529740270573117745299636309336266624443809366841812006263033815 : 1)
```

Montgomery

$$By^2 = x^3 + Ax^2 + x$$

$$B(A^2 - 4) \neq 0$$

curve25519:

$$A = 486662, B = 1 \quad \text{over the finite field} \quad \mathbb{F}_{2^{255}-19}$$

- konvertalhato Weierstrassza
- gyors az ismeretlen bazisu szorzas (sk*pk), es const-time

Twisted Edwards

Twisted edwards

$$ax^2 + y^2 = 1 + dx^2y^2$$

$$a \neq 1$$

$$a \neq d \neq 0$$

Ed25519:

$$a = -1, \quad d = \frac{-121665}{121666} \quad \text{over the finite field} \quad F_{2^{255}-19}$$

- konvertalhato Montgomeryva
- gyors fix (ez const time is) es dupla-bazisu skalar szorzas.

performance

Curve shape, representation	DBL	ADD	mADD	mDBL
Short Weierstrass projective	11	14	11	8
Short Weierstrass projective a4=-3	10	14	11	8
Twisted Edwards projective	7	11	10	6
Twisted Edwards Inverted	7	10	9	6
Twisted Edwards Extended	8	9	8	7
Edwards projective	7	11	9	6
Edwards curve inverted	7	10	9	6
Montgomery curve	4			3

elliptic curves zoo

- http://cr.yp.to/talks/2008.06.20/slides.pdf
- https://safecurves.cr.yp.to/equation.html
- https://hyperelliptic.org/EFD/g1p/index.html

EdDSA

non-interactive Schnorr zero-knowledge proof on the twisted edwards curve ed25519

EdDSA kulcsgeneralas

from hashlib import sha512

$$l = 2^{252} + 27742317777372353535851937790883648493$$

$$b = 256$$

$$k = randrange(2^b)$$

$$H = sha512(k)$$

$$a = int(H[:32])$$

$$A = aB$$

- privat kulcs: k v. H
- pub kulcs: A

EdDSA alairas

$$M = \text{"}uzenet\text{"}$$

$$r = sha512(H[32:]|M)$$

$$R = rB$$

$$S = (r + sha512(R, A, M)a) \ mod \ l$$

EdDSA ellenorzes

$$valid = 8SB == 8R + 8sha512(R, A, M)A$$

most akkor mi micsoda?

- curve25519: egy montgomery gorbe
- x25519: ECDH a curve25519 gorben
- ed25519: egy twisted edwards gorbe
- EdDSA: egy alairasi protokol az ed25519 gorben
- Ristretto: prime-order group over curve25519

ristretto

- sok kriptoprotokol prime-order groupot igenyel
- sajnos a modern gorbek 4 .v 8-as kofaktorral rendelkeznek
- aki ezt nem veszi figyelembe, ugy jarhat mint a monero, ahol ez octospending problemahoz vezetett
- a ristretto az edwards gorbeket "fixeli" meg
- 1/x included!

ki akarom probalni!

- sage a kriptografusok replje: `https://www.sagemath.org/`
- cocalc pedig az etherpadjuk: `https://cocalc.com/`

wikipedia

- https://en.wikipedia.org/wiki/Elliptic_curve
- https://en.wikipedia.org/wiki/Homogeneous_coordinates
- https://en.wikipedia.org/wiki/Montgomery_curve
- https://en.wikipedia.org/wiki/Edwards_curve
- https://en.wikipedia.org/wiki/Twisted_Edwards_curve
- https://en.wikipedia.org/wiki/Table_of_costs_of_operations_in_elliptic_curves
- https://en.wikipedia.org/wiki/Twists_of_curves
- https://en.wikipedia.org/wiki/Elliptic_curve_point_multiplication
- https://en.wikipedia.org/wiki/EdDSA

links

- https://safecurves.cr.yp.to/
- https://hackernoon.com/what-is-the-math-behind-elliptic-curve-cryptography-f61
- https://lukas-prokop.at/proj/eddsa/
- https://crypto.stackexchange.com/questions/27866
- https://blog.mozilla.org/warner/2011/11/29/ed25519-keys/
- https://github.com/mulllhausen/visual-secp256k1
- https://andrea.corbellini.name/2015/05/17/elliptic-curve-cryptography-a-gentle-introduction/
- https://www.cryptologie.net/article/193/schnorrs-signature-and-non-interactive-protocols/
- https://ristretto.group

kerdesek?